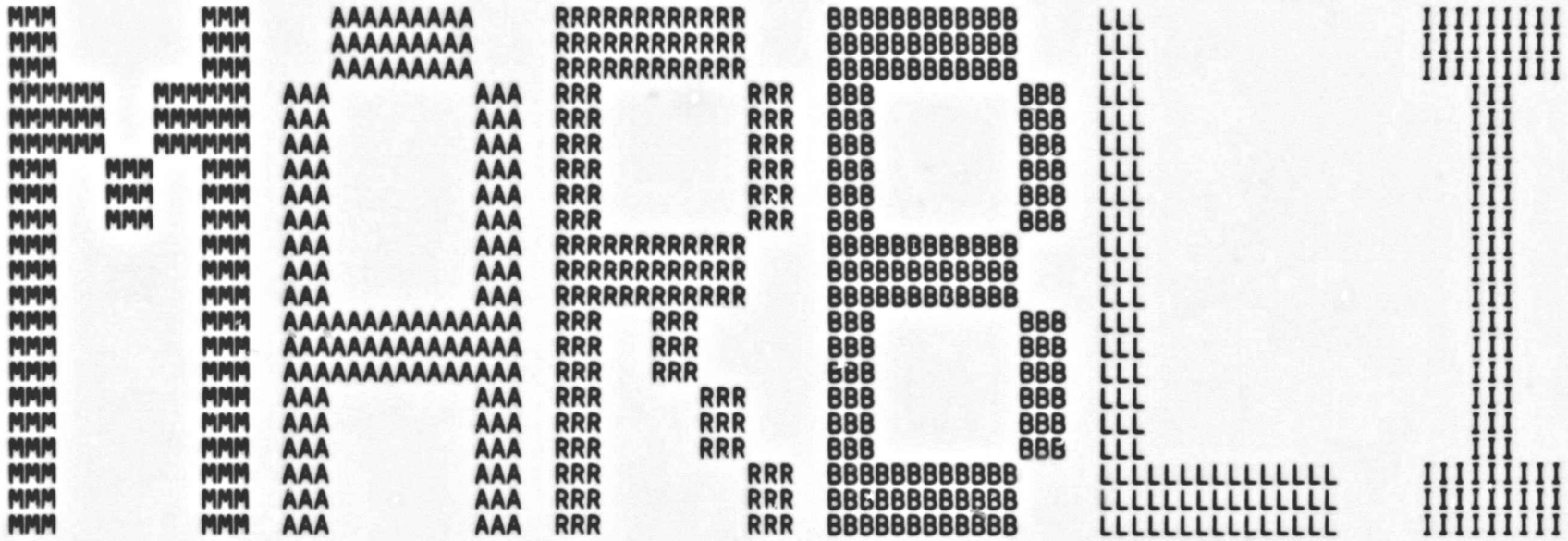
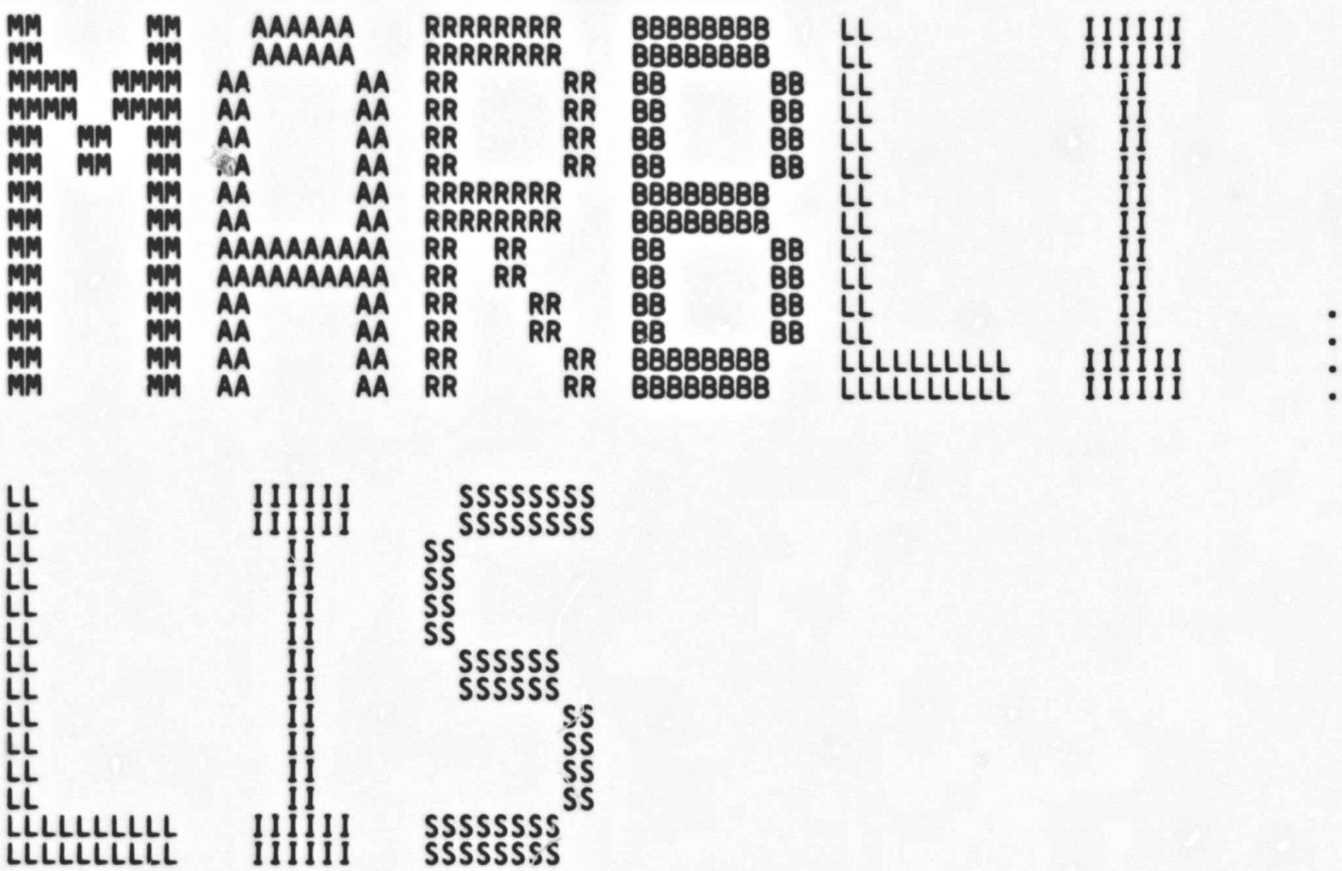

MMM MMM AAAAAAAAA RRRRRRRRRRR BBBBBBBBBBB LLL IIIIIIIII
MMM MMM AAAAAAAAA RRRRRRRRRRR BBBBBBBBBBB LLL IIIIIIIII
MMM MMM AAAAAAAAA RRRRRRRRRRR BBBBBBBBBBB LLL IIIIIIIII
MMMMMM MMMMMM AAA AAA RRR RRR BBB BBB LLL III
MMMMMM MMMMMM AAA AAA RRR RRR BBB BBB LLL III
MMMMMM MMMMMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM AAA AAA RRRRRRRRRRR BBBBBBBBBBB LLL III
MMM MMM AAA AAA RRRRRRRRRRR BBBBBBBBBBB LLL III
MMM MMM AAA AAA RRRRRRRRRRR BBBBBBBBBBB LLL III
MMM MMM AAAAAAAAAAAAAAA RRR RRR BBB BBB LLL III
MMM MMM AAAAAAAAAAAAAAA RRR RRR BBB BBB LLL III
MMM MMM AAAAAAAAAAAAAAA RRR RRR BBB BBB LLL III
MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM AAA AAA RRR RRR BBB BBB LLL III
MMM MMM AAA AAA RRR RRR BBBBBBBBBBB LLLLLLLLLLLLLLL IIIIIIIII
MMM MMM AAA AAA RRR RRR BBBBBBBBBBB LLLLLLLLLLLLLLL IIIIIIIII
MMM MMM AAA AAA RRR RRR BBBBBBBBBBB LLLLLLLLLLLLLLL IIIIIIIII

FILEID**MARBLI



MAR
V04

```
;     1          0001  0 MODULE MARBLI(IDENT='V04-000',MAIN=CONVERT %TITLE'MARS to BLISS Macro Converter')=
;     2          0002  1 BEGIN
;     3          0003  1
;     4          0004  1 !*****************************************************************************
;     5          0005  1 !*                                                                           *
;     6          0006  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     7          0007  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;     8          0008  1 !*  ALL RIGHTS RESERVED.                                                      *
;     9          0009  1 !*                                                                           *
;    10          0010  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    11          0011  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    12          0012  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    13          0013  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    14          0014  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    15          0015  1 !*  TRANSFERRED.                                                              *
;    16          0016  1 !*                                                                           *
;    17          0017  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    18          0018  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    19          0019  1 !*  CORPORATION.                                                              *
;    20          0020  1 !*                                                                           *
;    21          0021  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    22          0022  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    23          0023  1 !*                                                                           *
;    24          0024  1 !*                                                                           *
;    25          0025  1 !*****************************************************************************
;    26          0026  1
;    27          0027  1 !++
;    28          0028  1 !
;    29          0029  1 ! ABSTRACT:
;    30          0030  1 !       The routines in this module, along with XPORT
;    31          0031  1 !       convert a specifically formatted MARS macro file into a
;    32          0032  1 !       BLISS macro file.
;    33          0033  1 !
;    34          0034  1 !       The logic in this module is somewhat ad hoc and
;    35          0035  1 !       any perturbation of the format of the MARS file may result
;    36          0036  1 !       in these routines not working.
;    37          0037  1 !
;    38          0038  1 ! AUTHOR:
;    39          0039  1 !       P.C. Marks, CREATION DATE: 3 FEB 77
;    40          0040  1 !
;    41          0041  1 ! MODIFIED BY:
;    42          0042  1 !
;    43          0043  1 !       V03-001 MLJ0093         Martin L. Jack, 14-Jul-1982  11:38
;    44          0044  1 !               If the ".MACRO" line contains the string "CJF$", change the
;    45          0045  1 !               routine prefix from SYS$ to CJF$.  Do the same for RUF.
;    46          0046  1 !
;    47          0047  1 !       V02-003         BLS0073         Benn Schreiber          26-Aug-1981
;    48          0048  1 !               Disable per-macro output
;    49          0049  1 !
;    50          0050  1 !       V02-002         APL0001         Al Lehotsky             4-Jun-1981
;    51          0051  1 !               Change for 31-character names which was missed earlier.
;    52          0052  1 !
;    53          0053  1 !       V02-001         BLS0055         Benn Schreiber          3-Jun-1981
;    54          0054  1 !               Use BLISS linkage and GENERAL addressing mode
;    55          0055  1 !
;    56          0056  1 !       A.P. Lehotsky   31-Oct-79       Transport to VAX. Suppress "()" in
;    57          0057  1 !                                       macros without arguments, e.g. $HIBER
```

```
;    58          0058  1 !
;    59          0059  1 !--
;    60          0060  1
;    61          0061  1
;    62          0062  1 FORWARD ROUTINE
;    63          0063  1     CONVERT,                        ! Main routine
;    64          0064  1     CONVMACRO:          NOVALUE,    ! Convert a .MACRO definition
;    65          0065  1     OUTLIST,                        ! Output a parameter list
;    66          0066  1     CONVARGLIST:        NOVALUE;    ! Convert argument list of a .MACRO
;    67          0067  1
;    68          0068  1 MACRO
```

```
;    69          0069  1      VERSTR=     'MARBLI V03-001' %;
;    70          0070  1  LIBRARY 'SYS$LIBRARY:XPORT';
;    71          0071  1
;    72          0072  1  LITERAL
;    73          0073  1          MAX_NO_ARGS=    63,                 ! Max number of macro arguments
;    74          0074  1          MAX_ARG_LIST=   65*MAX_NO_ARGS,     ! Max characters in formal arg list
;    75          0075  1          MAX_REC_SIZE=   130;                ! Max input and output record length
;    76          0076  1
;    77          0077  1  MACRO
;    78        M 0078  1          REPEAT=
;    79          0079  1              WHILE 1 DO %,
;    80          0080  1
;    81          0081  1          ! Construct length and pointer parameters
;    82          0082  1          !
;    83        M 0083  1          CH$LEN_PTR[]=
;    84          0084  1              %CHARCOUNT(%REMAINING), CH$PTR(UPLIT(%REMAINING)) %,
;    85          0085  1
;    86          0086  1
;    87          0087  1          ! Return the index of string S in context C
;    88          0088  1          !
;    89        M 0089  1          CH$INDEX(CL,C,S) =
;    90          0090  1              CH$FIND_SUB(CL, C, %CHARCOUNT(S), CH$PTR(UPLIT(S))) %,
;    91          0091  1
;    92          0092  1          ! Return the size of the zero-truncated output buffer
;    93          0093  1          !
;    94        M 0094  1          TRUNCATED_OUTPUT =
;    95        M 0095  1                  CH$DIFF(CH$FIND_CH( MAX_REC_SIZE+1, CH$PTR(OUTPUT_RECORD),0),
;    96          0096  1                          CH$PTR(OUTPUT_RECORD)) %;
;    97          0097  1
;    98          0098  1
;    99          0099  1  OWN
;   100          0100  1      !+
;   101          0101  1      ! Declarations for processing a text line for macro declaration and
;   102          0102  1      ! the associated argument list.
;   103          0103  1      !-
;   104          0104  1      INPUT_RECORD:       VECTOR[CH$ALLOCATION(MAX_REC_SIZE)],
;   105          0105  1      INPUT_PTR,
;   106          0106  1      INPUT_LENGTH,
;   107          0107  1
;   108          0108  1                          ! Guarantee at least 1 zero byte at end of output
;   109          0109  1                          !  record s.t. TRUNCATED_OUTPUT macro cannot fail.
;   110          0110  1                          !
;   111          0111  1      OUTPUT_RECORD:      VECTOR[CH$ALLOCATION(MAX_REC_SIZE+1)],
;   112          0112  1      OUTPUT_PTR,
;   113          0113  1
;   114          0114  1      ARG_PTR,
;   115          0115  1      ARG_LENGTH,
;   116          0116  1      ARG_LIST:   VECTOR[CH$ALLOCATION(MAX_ARG_LIST)],
;   117          0117  1
;   118          0118  1      CALL_PTR,
;   119          0119  1      CALL_LENGTH,
;   120          0120  1      CALL_LIST:  VECTOR[CH$ALLOCATION(MAX_ARG_LIST)],
;   121          0121  1
;   122          0122  1      VAR_ARGS;
;   123          0123  1
;   124          0124  1  OWN
;   125          0125  1      !+
```

```
;  126          0126  1     ! I/O related declarations.
;  127          0127  1     !-
;  128          0128  1
;  129          0129  1     TERMINAL:   $XPO_IOB(),
;  130          0130  1     INPUT:      $XPO_IOB(),
;  131          0131  1     OUTPUT:     $XPO_IOB();
```

```
;  133          0132  1 ROUTINE CONVERT =
;  134          0133  1 !++
;  135          0134  1 ! FUNCTIONAL DESCRIPTION:
;  136          0135  1 !       This is the main routine of this module.
;  137          0136  1 !       The chief function is to examine an input line and determine
;  138          0137  1 !       whether it should be ignored or processed by other routines.
;  139          0138  1 !
;  140          0139  1 ! IMPLICIT INPUTS:
;  141          0140  1 !
;  142          0141  1 !       OWN storage
;  143          0142  1 !
;  144          0143  1 ! IMPLICIT OUTPUTS:
;  145          0144  1 !
;  146          0145  1 !       OWN storage
;  147          0146  1 !
;  148          0147  1 ! ROUTINE VALUE:
;  149          0148  1 !
;  150          0149  1 !       Success or an XPORT completion code
;  151          0150  1 !--
;  152          0151  2     BEGIN
;  153          0152  2     LOCAL
;  154          0153  2         PRINT_COMMENTS; ! Flag to pass comments and blank lines to output
;  155          0154  2
;  156          0155  2     $XPO_IOB_INIT(IOB=TERMINAL);
;  157          0156  2     $XPO_IOB_INIT(IOB=INPUT);
;  158          0157  2     $XPO_IOB_INIT(IOB=OUTPUT);
;  159          0158  2
;  160          0159  2     $XPO_OPEN( IOB=TERMINAL, FILE_SPEC=$XPO_INPUT);
;  161          0160  2 !    $XPO_PUT( IOB=TERMINAL,
;  162          0161  2 !       STRING=(%STRING('System-Service Macro Translator ',%EXPAND VERSTR)) );
;  163          0162  2
;  164          0163  2     REPEAT
;  165          0164  3         BEGIN
;  166          0165  4         IF NOT $XPO_GET(IOB=TERMINAL, PROMPT=('Input file (STARLET.MAR): ') )
;  167          0166  3         THEN
;  168          0167  3             RETURN XPO$_NORMAL;
;  169          0168  3
;  170        P 0169  3         IF $XPO_OPEN(IOB=INPUT,
;  171        P 0170  3                 FILE_SPEC=TERMINAL[IOB$T_STRING],       ! User supplied name
;  172        P 0171  3                 DEFAULT=('STARLET.MAR'),                ! The standard input
;  173          0172  4                 FAILURE=XPO$IO_FAILURE)
;  174          0173  3         THEN
;  175          0174  3             EXITLOOP                                    ! Got good input.
;  176          0175  2         END;
;  177          0176  2
;  178          0177  3     IF NOT $XPO_GET(IOB=TERMINAL, PROMPT=('Output file (*.B32): ') )
;  179          0178  2     THEN
;  180          0179  2         RETURN XPO$_NORMAL;
;  181          0180  2
;  182        P 0181  2     $XPO_OPEN(IOB=OUTPUT,
;  183        P 0182  2                 FILE_SPEC=TERMINAL[IOB$T_STRING],
;  184        P 0183  2                 DEFAULT=('*.B32'),
;  185        P 0184  2                 RELATED=INPUT[IOB$T_RESULTANT],
;  186          0185  2                 OPTION=OUTPUT);
;  187          0186  2
;  188        P 0187  2     $XPO_PUT(IOB=OUTPUT,
```

```
;  189     0188  2          STRING=(%STRING('! Translated from MACRO-32 by ',%EXPAND VERSTR)) );
;  190     0189  2      ! Set CSP pointer for the output record.
;  191     0190  2      !
;  192     0191  2      OUTPUT_PTR = CH$PTR(OUTPUT_RECORD);
;  193     0192  2      PRINT_COMMENTS = 0;
;  194     0193  2
;  195     0194  2
;  196     0195  2      ! Main loop:
;  197     0196  2      !   Exit from this loop when an end of file is read
;  198     0197  2      !   or an I/O error has occurred.
;  199     0198  2      !
;  200     0199  2      WHILE $XPO_GET(IOB=INPUT) DO
;  201     0200  3          BEGIN
;  202     0201  3          INPUT_LENGTH = .INPUT[IOB$H_STRING];
;  203     0202  3          INPUT_PTR = .INPUT[IOB$A_STRING];
;  204     0203  3
;  205     0204  3          IF .INPUT_LENGTH LSS 0 THEN EXITLOOP;
;  206     0205  3
;  207     0206  3          IF .INPUT_LENGTH EQL 0
;  208     0207  3          THEN
;  209     0208  4              BEGIN
;  210     0209  5              IF .PRINT_COMMENTS THEN $XPO_PUT(IOB=OUTPUT, STRING=('') )
;  211     0210  4              END
;  212     0211  3          ELSE
;  213     0212  4              BEGIN
;  214     0213  4              CH$COPY(.INPUT_LENGTH, .INPUT_PTR, 0, MAX_REC_SIZE, .OUTPUT_PTR);
;  215     0214  4
;  216     0215  4              INPUT_LENGTH = TRUNCATED_OUTPUT;
;  217     0216  4
;  218     0217  4              IF CH$RCHAR(.OUTPUT_PTR) EQL %C';'
;  219     0218  4              THEN
;  220     0219  5                  BEGIN
;  221     0220  5                  !
;  222     0221  5                  ! Comment line
;  223     0222  5                  !       Note comments are printed only after line ";*" is read
;  224     0223  5                  !
;  225     0224  5                  CH$WCHAR(%C'!', .OUTPUT_PTR);
;  226     0225  5                  IF .PRINT_COMMENTS
;  227     0226  5                  THEN
;  228     0227  5                      $XPO_PUT(IOB=OUTPUT, STRING=(.INPUT_LENGTH, .OUTPUT_PTR));
;  229     0228  5
;  230     0229  5                  IF CH$RCHAR(CH$PLUS(.OUTPUT_PTR, 1)) EQL %C'*'
;  231     0230  5                  THEN
;  232     0231  5                      PRINT_COMMENTS = 1
;  233     0232  5                  END
;  234     0233  5
;  235     0234  4              ELSE IF (CH$INDEX(.INPUT_LENGTH, .OUTPUT_PTR, '.MACRO')) NEQ 0
;  236     0235  4              THEN
;  237     0236  4                  IF (CH$INDEX(.INPUT_LENGTH, .OUTPUT_PTR, '_S')) NEQ 0
;  238     0237  4                  THEN
;  239     0238  4                      !
;  240     0239  4                      ! Macro in "_S" form:  Output it
;  241     0240  4                      !
;  242     0241  4                      CONVMACRO()
;  243     0242  4              END
;  244     0243  2      END;
;  245     0244  2
```

```
;  246          0245  2
;  247          0246  2      $XPO_CLOSE(IOB=INPUT);
;  248          0247  2      $XPO_CLOSE(IOB=OUTPUT);
;  249          0248  2
;  250          0249  2 !     $XPO_PUT( IOB=TERMINAL, STRING=('End MARBLI') );
;  251          0250  2      $XPO_CLOSE( IOB=TERMINAL );
;  252          0251  2
;  253          0252  2      XPO$_NORMAL
;  254          0253  1      END;


                                                                .TITLE  MARBLI MARS to BLISS Macro Converter
                                                                .IDENT  \V04-000\

                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                        54 55 50 4E 49 24 53 59 53  00000 P.AAA:  .ASCII  \SYS$INPUT\                                    ;
41 54 53 28 20 65 6C 69 66 20 74 75 70 6E 49  00009 P.AAB:  .ASCII  \Input file (STARLET.MAR): \                        ;
           20 3A 29 52 41 4D 2E 54 45 4C 52  00018                                                                      ;
           52 41 4D 2E 54 45 4C 52 41 54 53  00023 P.AAC:  .ASCII  \STARLET.MAR\                                        ;
2E 2A 28 20 65 6C 69 66 20 74 75 70 74 75 4F  0002E P.AAD:  .ASCII  \Output file (*.B32): \                             ;
                    20 3A 29 32 33 42 00003D                                                                         ;
                       32 33 42 2E 2A 00043 P.AAE:  .ASCII  \*.B32\                                                  ;
72 66 20 64 65 74 61 6C 73 6E 61 72 54 20 21  00048 P.AAF:  .ASCII  \! Translated from MACRO-32 by MARBLI V03\          ;
20 79 62 20 32 33 2D 4F 52 43 41 4D 20 6D 6F  00057                                                                  ;
              33 30 56 20 49 4C 42 52 41 4D  00066                                                                   ;
                          31 30 30 2D  00070           .ASCII  \-001\                                                  ;
                                       00074 P.AAG:    .BLKB   0                                                       ;
                    00 00 4F 52 43 41 4D 2E  00074 P.AAH:  .ASCII  \.MACRO\<0><0>                                     ;
                          00 00 53 5F  0007C P.AAI:  .ASCII  \_S\<0><0>                                              ;

                                                                .PSECT  $OWN$,NOEXE,2

                                                        00000 INPUT_RECORD:
                                                                .BLKB   132
                                                        00084 INPUT_PTR:
                                                                .BLKB   4
                                                        00088 INPUT_LENGTH:
                                                                .BLKB   4
                                                        0008C OUTPUT_RECORD:
                                                                .BLKB   132
                                                        00110 OUTPUT_PTR:
                                                                .BLKB   4
                                                        00114 ARG_PTR:  .BLKB   4
                                                        00118 ARG_LENGTH:
                                                                .BLKB   4
                                                        0011C ARG_LIST:
                                                                .BLKB   4096
                                                        0111C CALL_PTR:
                                                                .BLKB   4
                                                        01120 CALL_LENGTH:
                                                                .BLKB   4
                                                        01124 CALL_LIST:
                                                                .BLKB   4096
                                                        02124 VAR_ARGS:
                                                                .BLKB   4
                                                        02128 TERMINAL:
```

```
                                .BLKB   244
                02221C INPUT:   .BLKB   244
                02310 OUTPUT:   .BLKB   244
    0009        02404 $IOB$FILE_SPEC:
                                .WORD   9
  01  0E        02406           .BYTE   14, 1
00000000'       02408           .ADDRESS P.AAA
    001A        0240C $IOB$PROMPT:
                                .WORD   26
  01  0E        0240E           .BYTE   14, 1
00000000'       02410           .ADDRESS P.AAB
    000B        02414 $IOB$DEFAULT:
                                .WORD   11
  01  0E        02416           .BYTE   14, 1
00000000'       02418           .ADDRESS P.AAC
    0015        0241C $IOB$PROMPT:
                                .WORD   21
  01  0E        0241E           .BYTE   14, 1
00000000'       02420           .ADDRESS P.AAD
    0005        02424 $IOB$DEFAULT:
                                .WORD   5
  01  0E        02426           .BYTE   14, 1
00000000'       02428           .ADDRESS P.AAE
    002C        0242C $IOB$OUTPUT:
                                .WORD   44
  01  0E        0242E           .BYTE   14, 1
00000000'       02430           .ADDRESS P.AAF
    0000        02434 $IOB$OUTPUT:
                                .WORD   0
  01  0E        02436           .BYTE   14, 1
00000C00'       02438           .ADDRESS P.AAG

                IOB$=                   TERMINAL
                IOB$RESULTANT=          TERMINAL+28
                IOB$=                   INPUT
                IOB$RESULTANT=          INPUT+28
                IOB$=                   OUTPUT
                IOB$RESULTANT=          OUTPUT+28
                IOB$=                   TERMINAL
                IOB$=                   TERMINAL
                IOB$=                   INPUT
                $IOB$FILE_SPEC=         TERMINAL+52
                IOB$=                   TERMINAL
                IOB$=                   OUTPUT
                $IOB$FILE_SPEC=         TERMINAL+52
                $IOB$RELATED=           INPUT+28
                IOB$=                   OUTPUT
                IOB$=                   INPUT
                IOB$=                   OUTPUT
                IOB$=                   OUTPUT
                IOB$=                   INPUT
                IOB$=                   OUTPUT
                IOB$=                   TERMINAL
                        .EXTRN  XPO$OPEN, XPO$FAILURE
                        .EXTRN  XST$FREE_TEMP, XPO$GET
                        .EXTRN  XPO$IO_FAILURE, XPO$PUT
                        .EXTRN  XPO$CLOSE
```

```
                                                        .PSECT  $CODE$,NOWRT,2

                                      OFFC 00000 CONVERT:.WORD  Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11      ; 0132
                  5B 00000000G  EF   9E 00002          MOVAB   XPO$GET, R11                               ;
                  5A 00000000G  EF   9E 00009          MOVAB   XPO$OPEN, R10                              ;
                  59 00000000G  EF   9E 00010          MOVAB   XPO$FAILURE, R9                            ;
                  58     0000'  CF   9E 00017          MOVAB   IOB$, R8                                   ;
                  5E            08   C2 0001C          SUBL2   #8, SP                                     ;
 00F4   8F      00   6E         00   2C 0001F          MOVC5   #0, (SP), #0, #244, IOB$                   ; 0155
                          FE18  C8      00026                                                             ;
            FE18  C8 0301003D   8F   D0 00029          MOVL    #50397245, IOB$                            ;
            FE36  C8     020E   8F   B0 00032          MOVW    #526, IOB$RESULTANT+2                      ;
 00F4   8F      00   6E         00   2C 00039          MOVC5   #0, (SP), #0, #244, IOB$                   ; 0156
                          FF0C  C8      00040                                                             ;
            FF0C  C8 0301003D   8F   D0 00043          MOVL    #50397245, IOB$                            ;
            FF2A  C8     020E   8F   B0 0004C          MOVW    #526, IOB$RESULTANT+2                      ;
 00F4   8F      00   6E         00   2C 00053          MOVC5   #0, (SP), #0, #244, IOB$                   ; 0157
                                68      0005A                                                             ;
                  68 0301003D   8F   D0 0005B          MOVL    #50397245, IOB$                            ;
              1E  A8     020E   8F   B0 00062          MOVW    #526, IOB$RESULTANT+2                      ;
            FE1C  C8     00F4   C8   9E 00068          MOVAB   $IOB$FILE_SPEC, IOB$+4                     ; 0159
            FE44  C8            01   90 0006F          MOVB    #1, IOB$+44                                ;
                                59   DD 00074          PUSHL   R9                                         ;
                                7E   D4 00076          CLRL    -(SP)                                      ;
                          FE18  C8   9F 00078          PUSHAB  IOB$                                       ;
                  6A            03   FB 0007C          CALLS   #3, XPO$OPEN                               ;
                  50      FE3C  C8   D0 0007F 1$:      MOVL    IOB$+36, R0                                ; 0165
                                09   13 00084          BEQL    2$                                         ;
                                50   DD 00086          PUSHL   R0                                         ;
         00000000G  EF          01   FB 00088          CALLS   #1, XST$FREE_TEMP                          ;
            FE3C  C8      00FC  C8   9E 0008F 2$:      MOVAB   $IOB$PROMPT, IOB$+36                       ;
            FE44  C8            06   90 00096          MOVB    #6, IOB$+44                                ;
                                59   DD 0009B          PUSHL   R9                                         ;
                                7E   D4 0009D          CLRL    -(SP)                                      ;
                          FE18  C8   9F 0009F          PUSHAB  IOB$                                       ;
                  6B            03   FB 000A3          CALLS   #3, XPO$GET                                ;
                  4C            50   E9 000A6          BLBC    R0, 4$                                     ;
            FF10  C8      FE4C  C8   9E 000A9          MOVAB   $IOB$FILE_SPEC, IOB$+4                     ; 0172
            FF14  C8      0104  C8   9E 000B0          MOVAB   $IOB$DEFAULT, IOB$+8                       ;
            FF38  C8            01   90 000B7          MOVB    #1, IOB$+44                                ;
                     00000000G  EF   9F 000BC          PUSHAB  XPO$IO_FAILURE                             ;
                                7E   D4 000C2          CLRL    -(SP)                                      ;
                          FF0C  C8   9F 000C4          PUSHAB  IOB$                                       ;
                  6A            03   FB 000C8          CALLS   #3, XPO$OPEN                               ;
                  B1            50   E9 000CB          BLBC    R0, 1$                                     ;
                  50      FE3C  C8   D0 000CE          MOVL    IOB$+36, R0                                ; 0177
                                09   13 000D3          BEQL    3$                                         ;
                                50   DD 000D5          PUSHL   R0                                         ;
         00000000G  EF          01   FB 000D7          CALLS   #1, XST$FREE_TEMP                          ;
            FE3C  C8      010C  C8   9E 000DE 3$:      MOVAB   $IOB$PROMPT, IOB$+36                       ;
            FE44  C8            06   90 000E5          MOVB    #6, IOB$+44                                ;
                                59   DD 000EA          PUSHL   R9                                         ;
                                7E   D4 000EC          CLRL    -(SP)                                      ;
                          FE18  C8   9F 000EE          PUSHAB  IOB$                                       ;
                  6B            03   FB 000F2          CALLS   #3, XPO$GET                                ;
                  03            50   E8 000F5 4$:      BLBS    R0, 5$                                     ;
```

```
                                   015A  31 000F8      BRW     18$                                          ;
                04    A8    FE4C    C8  9E 000FB 5$:   MOVAB   $IOB$FILE_SPEC, IOB$+4                       ; 0185
                08    A8    0114    C8  9E 00101       MOVAB   $IOB$DEFAULT, IOB$+8                         ;
                0C    A8    FF28    C8  9E 00107       MOVAB   $IOB$RELATED, IOB$+12                        ;
                2E    A8            02  88 0010D       BISB2   #2, IOB$+46                                  ;
                2C    A8            01  90 00111       MOVB    #1, IOB$+44                                  ;
                                    59  DD 00115       PUSHL   R9                                           ;
                                    7E  D4 00117       CLRL    -(SP)                                        ;
                                    58  DD 00119       PUSHL   R8                                           ;
                      6A            03  FB 0011B       CALLS   #3, XPO$OPEN                                 ;
                44    A8    011C    C8  9E 0011E       MOVAB   $IOB$OUTPUT, IOB$+68                         ; 0188
                2C    A8            07  90 00124       MOVB    #7, IOB$+44                                  ;
                                    59  DD 00128       PUSHL   R9                                           ;
                                    7E  D4 0012A       CLRL    -(SP)                                        ;
                                    58  DD 0012C       PUSHL   R8                                           ;
           00000000G  EF            03  FB 0012E       CALLS   #3, XPO$PUT                                  ;
                DE00  C8    DD7C    C8  9E 00135       MOVAB   OUTPUT_RECORD, OUTPUT_PTR                    ; 0191
                                    57  D4 0013C       CLRL    PRINT_COMMENTS                               ; 0192
                FF38  C8            06  90 0013E 6$:   MOVB    #6, IOB$+44                                  ; 0199
                                    59  DD 00143       PUSHL   R9                                           ;
                                    7E  D4 00145       CLRL    -(SP)                                        ;
                            FF0C    C8  9F 00147       PUSHAB  IOB$                                         ;
                      6B            03  FB 0014B       CALLS   #3, XPO$GET                                  ;
                      03            50  E8 0014E       BLBS    R0, 8$                                       ;
                                   00C8  31 00151 7$:  BRW     17$                                          ;
                DD78  C8    FF40    C8  3C 00154 8$:   MOVZWL  INPUT+52, INPUT_LENGTH                       ; 0201
                DD74  C8    FF44    C8  D0 0015B       MOVL    INPUT+56, INPUT_PTR                          ; 0202
                      50    DD78    C8  D0 00162       MOVL    INPUT_LENGTH, R0                             ; 0204
                                    E8  19 00167       BLSS    7$                                           ;
                                    1C  12 00169       BNEQ    10$                                          ; 0206
                      D0            57  E9 0016B       BLBC    PRINT_COMMENTS, 6$                           ; 0209
                44    A8    0124    C8  9E 0016E       MOVAB   $IOB$OUTPUT, IOB$+68                         ;
                2C    A8            07  90 00174       MOVB    #7, IOB$+44                                  ;
                                    59  DD 00178       PUSHL   R9                                           ;
                                    7E  D4 0017A       CLRL    -(SP)                                        ;
                                    58  DD 0017C       PUSHL   R8                                           ;
           00000000G  EF            03  FB 0017E       CALLS   #3, XPO$PUT                                  ;
                                    B7  11 00185 9$:   BRB     6$                                           ; 0208
                      56    DE00    C8  D0 00187 10$:  MOVL    OUTPUT_PTR, R6                               ; 0213
  0082   8F           00    DD74    D8  50  2C 0018C   MOVC5   R0, @INPUT_PTR, #0, #130, (R6)               ;
                                    66     00195                                                            ;
       DD7C   C8    0083    8F      00  3A 00196       LOCC    #0, #131, OUTPUT_RECORD                      ; 0215
                                    02  12 0019E       BNEQ    11$                                          ;
                                    51  D4 001A0       CLRL    R1                                           ;
                      50    DD7C    C8  9E 001A2 11$:  MOVAB   OUTPUT_RECORD, R0                            ;
       DD78   C8            51      50  C3 001A7       SUBL3   R0, R1, INPUT_LENGTH                         ;
                      3B            66  91 001AD       CMPB    (R6), #59                                    ; 0217
                                    3C  12 001B0       BNEQ    13$                                          ;
                      66            21  90 001B2       MOVB    #33, (R6)                                    ; 0224
                      26            57  E9 001B5       BLBC    PRINT_COMMENTS, 12$                          ; 0225
                      6E    DD78    C8  B0 001B8       MOVW    INPUT_LENGTH, $IOB$OUTPUT                    ; 0227
                02    AE            0E  90 001BD       MOVB    #14, $IOB$OUTPUT+2                           ;
                03    AE            01  90 001C1       MOVB    #1, $IOB$OUTPUT+3                            ;
                04    AE            56  D0 001C5       MOVL    R6, $IOB$OUTPUT+4                            ;
                44    A8            6E  9E 001C9       MOVAB   $IOB$OUTPUT, IOB$+68                         ;
                2C    A8            07  90 001CD       MOVB    #7, IOB$+44                                  ;
                                    59  DD 001D1       PUSHL   R9                                           ;
```

```
                                         7E  D4 001D3           CLRL     -(SP)                                       ;
                                         58  DD 001D5           PUSHL    R8                                          ;
                   00000000G  EF         03  FB 001D7           CALLS    #3, XPO$PUT                                 ;
                              50   DE00  C8  D0 001DE  12$:     MOVL     OUTPUT_PTR, R0                              ; 0229
                              2A     01  A0  91 001E3           CMPB     1(R0), #42                                  ;
                                         9C  12 001E7           BNEQ     9$                                          ;
                              57         01  D0 001E9           MOVL     #1, PRINT_COMMENTS                          ; 0231
                                         97  11 001EC           BRB      9$                                          ; 0229
      66      DD78     C8      0000'  CF         06  39 001EE  13$:     MATCHC   #6, P.AAH, INPUT_LENGTH, (R6)       ; 0234
                                         03  13 001F7           BEQL     14$                                         ;
                              53         06  D0 001F9           MOVL     #6, R3                                      ;
                              53         06  C2 001FC  14$:     SUBL2    #6, R3                                      ;
                                         84  13 001FF           BEQL     9$                                          ;
      66      DD78     C8      0000'  CF         02  39 00201           MATCHC   #2, P.AAI, INPUT_LENGTH, (R6)       ; 0236
                                         03  13 0020A           BEQL     15$                                         ;
                              53         02  D0 0020C           MOVL     #2, R3                                      ;
                              53         02  C2 0020F  15$:     SUBL2    #2, R3                                      ;
                                         05  13 00212           BEQL     16$                                         ;
                       0000V  CF         00  FB 00214           CALLS    #0, CONVMACRO                               ; 0241
                                       FF22  31 00219  16$:     BRW      6$                                          ; 0206
                        FF38  C8         02  90 0021C  17$:     MOVB     #2, IOB$+44                                 ; 0246
                                         59  DD 00221           PUSHL    R9                                          ;
                                         7E  D4 00223           CLRL     -(SP)                                       ;
                                   FF0C  C8  9F 00225           PUSHAB   IOB$                                        ;
                   00000000G  EF         03  FB 00229           CALLS    #3, XPO$CLOSE                               ;
                          2C  A8         02  90 00230           MOVB     #2, IOB$+44                                 ; 0247
                                         59  DD 00234           PUSHL    R9                                          ;
                                         7E  D4 00236           CLRL     -(SP)                                       ;
                                         58  DD 00238           PUSHL    R8                                          ;
                   00000000G  EF         03  FB 0023A           CALLS    #3, XPO$CLOSE                               ;
                        FE44  C8         02  90 00241           MOVB     #2, IOB$+44                                 ; 0250
                                         59  DD 00246           PUSHL    R9                                          ;
                                         7E  D4 00248           CLRL     -(SP)                                       ;
                                   FE18  C8  9F 0024A           PUSHAB   IOB$                                        ;
                   00000000G  EF         03  FB 0024E           CALLS    #3, XPO$CLOSE                               ;
                              50 00208001  8F  D0 00255  18$:     MOVL     #2129921, R0                                ; 0253
                                         04     0025C           RET                                                  ;

; Routine Size:  605 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  256     0254  1 ROUTINE CONVMACRO: NOVALUE=
;  257     0255  1
;  258     0256  1 !++
;  259     0257  1 ! FUNCTIONAL DESCRIPTION:
;  260     0258  1 !       This routine reads and processes a MARS system macro
;  261     0259  1 !       It calls the routine CONVARGLIST to convert the argument list
;  262     0260  1 !       of the macro definition.
;  263     0261  1 !
;  264     0262  1 ! PARAMETERS:
;  265     0263  1 !
;  266     0264  1 !       NONE
;  267     0265  1 !
;  268     0266  1 ! ROUTINE VALUE:
;  269     0267  1 !
;  270     0268  1 !       NONE
;  271     0269  1 !
;  272     0270  1 !--
;  273     0271  1
;  274     0272  2     BEGIN
;  275     0273  2     LOCAL
;  276     0274  2         PREFIX_PTR,
;  277     0275  2         PTR,                        ! Temp CH$ pointer for copying arglists
;  278     0276  2         TEMP_PTR1,
;  279     0277  2         TEMP_PTR2,
;  280     0278  2         TEMPSTRING:     VECTOR[CH$ALLOCATION(80)],
;  281     0279  2         MACRO_PTR,
;  282     0280  2         MACRO_LENGTH,
;  283     0281  2         MACRO_NAME:     VECTOR[CH$ALLOCATION(31)];
;  284     0282  2
;  285     0283  2
;  286     0284  2     MACRO_PTR = CH$PTR(MACRO_NAME);
;  287     0285  2
;  288     0286  2
;  289     0287  2     ! If the .MACRO line contains the string "CJF$", then change the routine
;  290     0288  2     ! name prefix from SYS$ to CJF$.  Do the same for RUF.
;  291     0289  2     !
;  292     0290  2     PREFIX_PTR = CH$PTR(UPLIT BYTE('SYS'));
;  293     0291  2     IF CH$FIND_SUB(MAX_REC_SIZE, .OUTPUT_PTR, CH$LEN_PTR('CJF$')) NEQ 0
;  294     0292  2         THEN PREFIX_PTR = CH$PTR(UPLIT BYTE('CJF'));
;  295     0293  2     IF CH$FIND_SUB(MAX_REC_SIZE, .OUTPUT_PTR, CH$LEN_PTR('RUF$')) NEQ 0
;  296     0294  2         THEN PREFIX_PTR = CH$PTR(UPLIT BYTE('RUF'));
;  297     0295  2
;  298     0296  2
;  299     0297  2     ! Search for the "$" which begins the macro name and
;  300     0298  2     ! the "_" which ends it, and initialize MACRO_NAME and MACRO_LENGTH
;  301     0299  2     !
;  302     0300  2     TEMP_PTR1 = CH$FIND_CH(MAX_REC_SIZE, .OUTPUT_PTR, %C'$');
;  303     0301  2     TEMP_PTR2 = CH$FIND_CH(MAX_REC_SIZE, .OUTPUT_PTR, %C'_');
;  304     0302  2     MACRO_LENGTH = CH$DIFF(.TEMP_PTR2, .TEMP_PTR1);
;  305     0303  2     CH$MOVE(.MACRO_LENGTH, .TEMP_PTR1, .MACRO_PTR);
;  306     0304  2
;  307     0305  2
;  308     0306  2     ! Inform user of progress
;  309     0307  2     !
;  310     0308  2 !    CH$COPY(CH$LEN_PTR('Macro: '),
;  311     0309  2 !            .MACRO_LENGTH, .MACRO_PTR,
;  312     0310  2 !            0,
```

```
;  313          0311  2 !            80, CH$PTR(TEMPSTRING) );
;  314          0312  2
;  315          0313  2 !    $XPO_PUT( IOB=TERMINAL, STRING=(80, CH$PTR(TEMPSTRING) ) );
;  316          0314  2
;  317          0315  2
;  318          0316  2     ! Gather the argument list
;  319          0317  2     !
;  320          0318  2     CONVARGLIST(.TEMP_PTR2);
;  321          0319  2
;  322          0320  2
;  323          0321  2     ! Put out one of the lines
;  324          0322  2     !   "KEYWORDMACRO $name(formal=default,...)="
;  325          0323  2     ! or
;  326          0324  2     !   "MACRO $name[]="        if VAR_ARGS is true
;  327          0325  2     !
;  328          0326  2     IF .VAR_ARGS
;  329          0327  2     THEN
;  330          0328  2         CH$COPY(
;  331          0329  2             CH$LEN_PTR('MACRO '),
;  332          0330  2             .MACRO_LENGTH,
;  333          0331  2             .MACRO_PTR,
;  334          0332  2             CH$LEN_PTR('[]='),
;  335          0333  2             0,
;  336          0334  2             MAX_REC_SIZE,
;  337          0335  2             .OUTPUT_PTR)
;  338          0336  2         ELSE
;  339          0337  3             BEGIN
;  340          0338  3             LOCAL
;  341          0339  3                 PTR;
;  342          0340  3
;  343          0341  3             PTR = CH$MOVE(CH$LEN_PTR('KEYWORDMACRO '), .OUTPUT_PTR);
;  344          0342  3             PTR = CH$MOVE(.MACRO_LENGTH, .MACRO_PTR, .PTR);
;  345          0343  3
;  346          0344  3             IF .ARG_LENGTH GTR 0            ! If there are formal parameters, then
;  347          0345  3             THEN                            !  put out the argument list.
;  348          0346  4                 BEGIN
;  349          0347  4                 PTR = CH$MOVE( CH$LEN_PTR('('), .PTR );
;  350          0348  4                 PTR = OUTLIST( .PTR, .ARG_LENGTH, .ARG_PTR);
;  351          0349  4                 PTR = CH$MOVE( CH$LEN_PTR(')='), .PTR)
;  352          0350  4                 END
;  353          0351  3             ELSE
;  354          0352  3                 CH$WCHAR_A(%C'=', PTR);     ! There were no formal parameters
;  355          0353  3
;  356          0354  3             CH$WCHAR(0, .PTR)               ! Mark end of string
;  357          0355  2             END;
;  358          0356  2
;  359          0357  2
;  360        P 0358  2     $XPO_PUT(IOB=OUTPUT,
;  361          0359  2             STRING=(TRUNCATED_OUTPUT, .OUTPUT_PTR) );
;  362          0360  2
;  363          0361  2
;  364          0362  2     ! Put out the line "   ("
;  365          0363  2     !
;  366          0364  2     $XPO_PUT(IOB=OUTPUT, STRING=('    (') );
;  367          0365  2
;  368          0366  2
;  369          0367  2     ! Put out the line "    EXTERNAL ROUTINE SYS$name: BLISS ADDRESSING_MODE(GENERAL);"
```

```
;  370     0368  2     !
;  371     0369  2     CH$COPY(
;  372     0370  2             CH$LEN_PTR('    EXTERNAL ROUTINE '),
;  373     0371  2             3, .PREFIX_PTR,
;  374     0372  2             .MACRO_LENGTH,
;  375     0373  2             .MACRO_PTR,
;  376     0374  2             CH$LEN_PTR(': BLISS ADDRESSING_MODE(GENERAL);'),
;  377     0375  2             0,
;  378     0376  2             MAX_REC_SIZE,
;  379     0377  2             .OUTPUT_PTR);
;  380     0378  2     $XPO_PUT(IOB=OUTPUT, STRING=(TRUNCATED_OUTPUT, .OUTPUT_PTR) );
;  381     0379  2
;  382     0380  2
;  383     0381  2     ! Put out the line "    SYS$name(formal,...)"
;  384     0382  2     !
;  385     0383  2     PTR = CH$COPY(
;  386     0384  2             CH$LEN_PTR('    '),
;  387     0385  2             3, .PREFIX_PTR,
;  388     0386  2             .MACRO_LENGTH,
;  389     0387  2             .MACRO_PTR,
;  390     0388  2             CH$LEN_PTR('('),
;  391     0389  2             0,
;  392     0390  2             .MACRO_LENGTH + 8,          ! Exact copying
;  393     0391  2             .OUTPUT_PTR);
;  394     0392  2
;  395     0393  2     PTR = OUTLIST( .PTR, .CALL_LENGTH, .CALL_PTR);
;  396     0394  2     PTR = CH$MOVE( CH$LEN_PTR(')', %CHAR(0)), .PTR );
;  397     0395  2
;  398     0396  2
;  399     0397  2     $XPO_PUT( IOB=OUTPUT, STRING=(TRUNCATED_OUTPUT, .OUTPUT_PTR) );
;  400     0398  2
;  401     0399  2
;  402     0400  2     ! Close up the macro declaration
;  403     0401  2     !
;  404     0402  3     $XPO_PUT(IOB=OUTPUT, STRING=('    ) %;') )
;  405     0403  1     END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                    53  59  53  00080 P.AAJ:  .ASCII  \SYS\                                ;
                                                                00083        .BLKB   1
                                                24  46  4A  43  00084 P.AAK:  .ASCII  \CJF$\                               ;
                                                    46  4A  43  00088 P.AAL:  .ASCII  \CJF\                                ;
                                                                0008B        .BLKB   1
                                                24  46  55  52  0008C P.AAM:  .ASCII  \RUF$\                               ;
                                                    46  55  52  00090 P.AAN:  .ASCII  \RUF\                                ;
                                                                00093        .BLKB   1
                                        00  00  20  4F  52  43  41  4D  00094 P.AAO:  .ASCII  \MACRO \<0><0>                    ;
                                                    00  3D  5D  5B  0009C P.AAP:  .ASCII  \[]=\<0>                         ;
00  00  20  4F  52  43  41  4D  44  52  4F  57  59  45  4B  000A0 P.AAQ:  .ASCII  \KEYWORDMACRO \<0><0><0>                  ;
                                                                00  000AF        ;
                                                    00  00  00  28  000B0 P.AAR:  .ASCII  \(\<0><0><0>                     ;
                                                    00  00  3D  29  000B4 P.AAS:  .ASCII  \)=\<0><0>                       ;
                                                28  20  20  20  20  000B8 P.AAT:  .ASCII  \    (\                        ;
                                                                000BD        .BLKB   3
```

```
4F 52 20 4C 41 4E 52 45 54 58 45 20 20 20 20 000C0 P.AAU:   .ASCII  \    EXTERNAL ROUTINE \<0><0><0>                 ;
                  00 00 00 20 45 4E 49 54 55 000CF                                                                  ;
53 53 45 52 44 44 41 20 53 53 49 4C 42 20 3A 000D8 P.AAV:   .ASCII  \: BLISS ADDRESSING_MODE(GENERAL);\<0><0>         ;
41 52 45 4E 45 47 28 45 44 4F 4D 5F 47 4E 49 000E7                                                                  ;
                              00 00 3B 29 4C 000F6                                                                  ;
                                          00 000FB          .ASCII  <0>                                             ;
                                 20 20 20 20 000FC P.AAW:   .ASCII  \    \                                          ;
                                 00 00 00 28 00100 P.AAX:   .ASCII  \(\<0><0><0>                                    ;
                                 00 00 00 29 00104 P.AAY:   .ASCII  \)\<0><0><0>                                    ;
                                 00 00 00 00 00108          .ASCII  <0><0><0><0>                                    ;
                     3B 25 20 29 20 20 20 20 0010C P.AAZ:   .ASCII  \    ) %;\                                      ;

                                                            .PSECT  $OWN$,NOEXE,2

                                   0005 0243C $IOB$OUTPUT:
                                                            .WORD   5                                               ;
                                  01 0E 0243E               .BYTE   14, 1                                           ;
                              00000000' 02440               .ADDRESS P.AAT                                          ;
                                   0008 02444 $IOB$OUTPUT:
                                                            .WORD   8                                               ;
                                  01 0E 02446               .BYTE   14, 1                                           ;
                              00000000' 02448               .ADDRESS P.AAZ                                          ;

                                           IOB$=                    OUTPUT
                                           IOB$=                    OUTPUT
                                           IOB$=                    OUTPUT
                                           IOB$=                    OUTPUT
                                           IOB$=                    OUTPUT


                                                            .PSECT  $CODE$,NOWRT,2

                                   0FFC 00000 CONVMACRO:
                                                            .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11            ; 0254
              5B    0000'  CF 9E 00002                      MOVAB   OUTPUT_PTR, R11                                 ;
              5E       88  AE 9E 00007                      MOVAB   -120(SP), SP                                    ;
              59       08  AE 9E 0000B                      MOVAB   MACRO_NAME, MACRO_PTR                           ; 0284
              5A    0000'  CF 9E 0000F                      MOVAB   P.AAJ, PREFIX_PTR                               ; 0290
              54           6B D0 00014                      MOVL    OUTPUT_PTR, R4                                  ; 0291
64   0082  8F    0000'  CF 04 39 00017                      MATCHC  #4, P.AAK, #130, (R4)                           ;
                           03 13 00020                      BEQL    1$                                              ;
              53           04 D0 00022                      MOVL    #4, R3                                          ;
              53           04 C2 00025 1$:                  SUBL2   #4, R3                                          ;
                           05 13 00028                      BEQL    2$                                              ;
              5A    0000'  CF 9E 0002A                      MOVAB   P.AAL, PREFIX_PTR                               ; 0292
64   0082  8F    0000'  CF 04 39 0002F 2$:                  MATCHC  #4, P.AAM, #130, (R4)                           ; 0293
                           03 13 00038                      BEQL    3$                                              ;
              53           04 D0 0003A                      MOVL    #4, R3                                          ;
              53           04 C2 0003D 3$:                  SUBL2   #4, R3                                          ;
                           05 13 00040                      BEQL    4$                                              ;
              5A    0000'  CF 9E 00042                      MOVAB   P.AAN, PREFIX_PTR                               ; 0294
        64   0082  8F      24 3A 00047 4$:                  LOCC    #36, #130, (R4)                                 ; 0300
                           02 12 0004D                      BNEQ    5$                                              ;
                           51 D4 0004F                      CLRL    R1                                              ;
              52           51 D0 00051 5$:                  MOVL    R1, TEMP_PTR1                                   ;
        64   0082  8F   5F 8F 3A 00054                      LOCC    #95, #130, (R4)                                 ; 0301
                           02 12 0005B                      BNEQ    6$                                              ;
```

```
                                    51  D4 0005D       CLRL    R1                                              ;
                        57          51  D0 0005F 6$:   MOVL    R1, TEMP_PTR2                                   ;
              56        57          52  C3 00062       SUBL3   TEMP_PTRT, TEMP_PTR2, MACRO_LENGTH              ; 0302
              69        62          56  28 00066       MOVC3   MACRO_LENGTH, (TEMP_PTR1), (MACRO_PTR)          ; 0303
                                    57  DD 0006A       PUSHL   TEMP_PTR2                                       ; 0318
                 0000V  CF          01  FB 0006C       CALLS   #1, CONVARGLIST                                 ;
                        2F    2014  CB  E9 00071       BLBC    VAR_ARGS, 7$                                    ; 0326
                        58      82  8F  9A 00076       MOVZBL  #130, R8                                        ; 0328
                        57          6B  D0 0007A       MOVL    OUTPUT_PTR, R7                                  ; 0335
      58      00 0000'  CF          06  2C 0007D       MOVC5   #6, P.AAO, #0, R8, (R7)                         ;
                                    67     00084                                                               ;
                                    4F  18 00085       BGEQ    10$                                             ;
                        57          06  C0 00087       ADDL2   #6, R7                                          ;
                        58          06  C2 0008A       SUBL2   #6, R8                                          ;
      58      00        69          56  2C 0008D       MOVC5   MACRO_LENGTH, (MACRO_PTR), #0, R8, (R7)         ;
                                    67     00092                                                               ;
                                    41  18 00093       BGEQ    10$                                             ;
                        57          56  C0 00095       ADDL2   MACRO_LENGTH, R7                                ;
                        58          56  C2 00098       SUBL2   MACRO_LENGTH, R8                                ;
      58      00 0000'  CF          03  2C 0009B       MOVC5   #3, P.AAP, #0, R8, (R7)                         ;
                                    67     000A2                                                               ;
                                    31  11 000A3       BRB     10$                                             ; 0328
          00  BB 0000'  CF          0D  28 000A5 7$:   MOVC3   #13, P.AAQ, @OUTPUT_PTR                         ; 0341
              63        69          56  28 000AC       MOVC3   MACRO_LENGTH, (MACRO_PTR), (PTR)                ; 0342
                        50      08  AB  D0 000B0       MOVL    ARG_LENGTH, R0                                  ; 0344
                                    1B  15 000B4       BLEQ    8$                                              ;
                        83   0000'  CF  90 000B6       MOVB    P.AAR, (PTR)+                                   ; 0347
                                04  AB  DD 000BB       PUSHL   ARG_PTR                                         ; 0348
                                    50  DD 000BE       PUSHL   R0                                              ;
                                    53  DD 000C0       PUSHL   PTR                                             ;
                 0000V  CF          03  FB 000C2       CALLS   #3, OUTLIST                                     ;
                        53          50  D0 000C7       MOVL    R0, PTR                                         ;
                        83   0000'  CF  B0 000CA       MOVW    P.AAS, (PTR)+                                   ; 0349
                                    03  11 000CF       BRB     9$                                              ; 0352
                        83          3D  90 000D1 8$:   MOVB    #61, (PTR)+                                     ;
                                    63  94 000D4 9$:   CLRB    (PTR)                                           ; 0354
        FF7C  CB  0083  8F          00  3A 000D6 10$:  LOCC    #0, #131, OUTPUT_RECORD                         ; 0359
                                    02  12 000DE       BNEQ    11$                                             ;
                                    51  D4 000E0       CLRL    R1                                              ;
                        50    FF7C  CB  9E 000E2 11$:  MOVAB   OUTPUT_RECORD, R0                               ;
              6E        51          50  A3 000E7       SUBW3   R0, R1, $IOB$OUTPUT                             ;
                   02   AE          0E  90 000EB       MOVB    #14, $IOB$OUTPUT+2                              ;
                   03   AE          01  90 000EF       MOVB    #1, $IOB$OUTPUT+3                               ;
                   04   AE          6B  D0 000F3       MOVL    OUTPUT_PTR, $IOB$OUTPUT+4                       ;
                 2244   CB          6E  9E 000F7       MOVAB   $IOB$OUTPUT, IOB$+68                            ;
                 222C   CB          07  90 000FC       MOVB    #7, IOB$+44                                     ;
                          00000000G EF  9F 00101       PUSHAB  XPO$FAILURE                                     ;
                                    7E  D4 00107       CLRL    -(SP)                                           ;
                              2200  CB  9F 00109       PUSHAB  IOB$                                            ;
             00000000G  EF          03  FB 0010D       CALLS   #3, XPO$PUT                                     ;
                 2244   CB    232C  CB  9E 00114       MOVAB   $IOB$OUTPUT, IOB$+68                            ; 0364
                 222C   CB          07  90 0011B       MOVB    #7, IOB$+44                                     ;
                          00000000G EF  9F 00120       PUSHAB  XPO$FAILURE                                     ;
                                    7E  D4 00126       CLRL    -(SP)                                           ;
                              2200  CB  9F 00128       PUSHAB  IOB$                                            ;
             00000000G  EF          03  FB 0012C       CALLS   #3, XPO$PUT                                     ;
                        58      82  8F  9A 00133       MOVZBL  #130, R8                                        ; 0369
```

```
                                  57    6B  D0 00137          MOVL    OUTPUT_PTR, R7                              ; 0377
   58          00      0000'      CF    15  2C 0013A          MOVC5   #21, P.AAU, #0, R8, (R7)                    ;
                                        67     00141                                                              ;
                                        2A  18 00142          BGEQ    12$                                         ;
                                  57    15  C0 00144          ADDL2   #21, R7                                     ;
                                  58    15  C2 00147          SUBL2   #21, R8                                     ;
   58          00                 6A    03  2C 0014A          MOVC5   #3, (PREFIX_PTR), #0, R8, (R7)              ;
                                        67     0014F                                                              ;
                                        1C  18 00150          BGEQ    12$                                         ;
                                  57    03  C0 00152          ADDL2   #3, R7                                      ;
                                  58    03  C2 00155          SUBL2   #3, R8                                      ;
   58          00                 69    56  2C 00158          MOVC5   MACRO_LENGTH, (MACRO_PTR), #0, R8, (R7)     ;
                                        67     0015D                                                              ;
                                        0E  18 0015E          BGEQ    12$                                         ;
                                  57    56  C0 00160          ADDL2   MACRO_LENGTH, R7                            ;
                                  58    56  C2 00163          SUBL2   MACRO_LENGTH, R8                            ;
   58          00      0000'      CF    21  2C 00166          MOVC5   #33, P.AAV, #0, R8, (R7)                    ;
                                        67     0016D                                                              ;
         FF7C  CB      0083       8F    00  3A 0016E  12$:    LOCC    #0, #131, OUTPUT_RECORD                     ; 0378
                                        02  12 00176          BNEQ    13$                                         ;
                                        51  D4 00178          CLRL    R1                                          ;
                                  50  FF7C  CB  9E 0017A  13$: MOVAB  OUTPUT_RECORD, R0                           ;
               6E                 51    50  A3 0017F          SUBW3   R0, R1, $IOB$OUTPUT                         ;
                         02       AE    0E  90 00183          MOVB    #14, $IOB$OUTPUT+2                          ;
                         03       AE    01  90 00187          MOVB    #1, $IOB$OUTPUT+3                           ;
                         04       AE    6B  D0 0018B          MOVL    OUTPUT_PTR, $IOB$OUTPUT+4                   ;
                       2244       CB    6E  9E 0018F          MOVAB   $IOB$OUTPUT, IOB$+68                        ;
                       222C       CB    07  90 00194          MOVB    #7, IOB$+44                                 ;
                             00000000G  EF  9F 00199          PUSHAB  XPO$FAILURE                                 ;
                                        7E  D4 0019F          CLRL    -(SP)                                       ;
                                  2200  CB  9F 001A1          PUSHAB  IOB$                                        ;
                  00000000G       EF    03  FB 001A5          CALLS   #3, XPO$PUT                                 ;
                                  58    08  A6  9E 001AC      MOVAB   8(R6), R8                                   ; 0390
                                  57    6B  D0 001B0          MOVL    OUTPUT_PTR, R7                              ; 0391
   58          00      0000'      CF    04  2C 001B3          MOVC5   #4, P.AAW, #0, R8, (R7)                     ;
                                        67     001BA                                                              ;
                                        2A  18 001BB          BGEQ    14$                                         ;
                                  57    04  C0 001BD          ADDL2   #4, R7                                      ;
                                  58    04  C2 001C0          SUBL2   #4, R8                                      ;
   58          00                 6A    03  2C 001C3          MOVC5   #3, (PREFIX_PTR), #0, R8, (R7)              ;
                                        67     001C8                                                              ;
                                        1C  18 001C9          BGEQ    14$                                         ;
                                  57    03  C0 001CB          ADDL2   #3, R7                                      ;
                                  58    03  C2 001CE          SUBL2   #3, R8                                      ;
   58          00                 69    56  2C 001D1          MOVC5   MACRO_LENGTH, (MACRO_PTR), #0, R8, (R7)     ;
                                        67     001D6                                                              ;
                                        0E  18 001D7          BGEQ    14$                                         ;
                                  57    56  C0 001D9          ADDL2   MACRO_LENGTH, R7                            ;
                                  58    56  C2 001DC          SUBL2   MACRO_LENGTH, R8                            ;
   58          00      0000'      CF    01  2C 001DF          MOVC5   #1, P.AAX, #0, R8, (R7)                     ;
                                        67     001E6                                                              ;
                                  100C  CB  DD 001E7  14$:    PUSHL   CALL_PTR                                    ; 0393
                                  1010  CB  DD 001EB          PUSHL   CALL_LENGTH                                 ;
                                        53  DD 001EF          PUSHL   PTR                                         ;
                       0000V      CF    03  FB 001F1          CALLS   #3, OUTLIST                                 ;
                                  53    50  D0 001F6          MOVL    R0, PTR                                     ;
                                  83  0000'  CF  B0 001F9     MOVW    F.AAY, (PTR)+                               ; 0394
```

```
        FF7C   CB      0083   8F              00  3A 001FE          LOCC    #0, #131, OUTPUT_RECORD          ; 0397
                                              02  12 00206          BNEQ    15$                              :
                                              51  D4 00208          CLRL    R1                               :
                              50     FF7C     CB  9E 0020A 15$:     MOVAB   OUTPUT_RECORD, R0                :
               6E             51              50  A3 0020F          SUBW3   R0, R1, $IOB$OUTPUT              :
                       02     AE              0E  90 00213          MOVB    #14, $IOB$OUTPUT+2               :
                       03     AE              01  90 00217          MOVB    #1, $IOB$OUTPUT+3                :
                       04     AE              6B  D0 0021B          MOVL    OUTPUT_PTR, $IOB$OUTPUT+4        :
                     2244     CB              6E  9E 0021F          MOVAB   $IOB$OUTPUT, IOB$+68             :
                     222C     CB              07  90 00224          MOVB    #7, IOB$+44                      :
                                  00000000G   EF  9F 00229          PUSHAB  XPO$FAILURE                      :
                                              7E  D4 0022F          CLRL    -(SP)                            :
                                       2200   CB  9F 00231          PUSHAB  IOB$                             :
                00000000G     EF              03  FB 00235          CALLS   #3, XPO$PUT                      :
                     2244     CB       2334   CB  9E 0023C          MOVAB   $IOB$OUTPUT, IOB$+68             : 0402
                     222C     CB              07  90 00243          MOVB    #7, IOB$+44                      :
                                  00000000G   EF  9F 00248          PUSHAB  XPO$FAILURE                      :
                                              7E  D4 0024E          CLRL    -(SP)                            :
                                       2200   CB  9F 00250          PUSHAB  IOB$                             :
                00000000G     EF              03  FB 00254          CALLS   #3, XPO$PUT                      :
                                                  04 0025B          RET                                      ; 0403

; Routine Size:  604 bytes,    Routine Base:  $CODE$ + 025D
```

```
ROUTINE OUTLIST(BUFPTR, LEN, LPTR)=
!+
! FUNCTION
!       Dump an argument list buffer (which may span several source
!       records..) to the output file.  Continuation lines are flagged
!       by an ASCII NUL (0) byte.
! INPUTS
!       BUFPTR  - byte-pointer into the OUTPUT_RECORD containing prefix
!                 information for the first record written.
!
!       LEN     - Length of the argument-list string.
!
!       LPTR    - Pointer to the argument-list string.
! OUTPUT
!       Updated "BUFPTR"
!-
    BEGIN
    LOCAL
        IPTR,               ! Input string pointer (arg-list)
        OPTR,               ! Copy of BUFPTR...
        CHAR;               ! Character temporary

    IPTR = .LPTR;
    OPTR = .BUFPTR;

    DECR I FROM .LEN-1 TO 0 DO
        BEGIN
        CHAR = CH$RCHAR_A( IPTR );          ! Get next character and copy to
        CH$WCHAR_A( .CHAR, OPTR );          !  output string

        IF .CHAR EQL 0
        THEN
            BEGIN
            $XPO_PUT( IOB=OUTPUT,
                    STRING=( TRUNCATED_OUTPUT, .OUTPUT_PTR)
                    );
            OPTR = .OUTPUT_PTR;                     ! Restore buffer ptr.
            CH$WCHAR_A( %CHAR(9), OPTR )            !  and TAB from left-margin
            END
        END;

    .OPTR
    END;


                                IOB$=                   OUTPUT


                003C 00000 OUTLIST:.WORD    Save R2,R3,R4,R5                ; 0404
  5E              08 C2 00002        SUBL2    #8, SP
  55        0C    AC D0 00005        MOVL     LPTR, IPTR                      ; 0426
  52        04    AC 7D 00009        MOVQ     BUFPTR, OPTR                    ; 0427
                  52 11 0000D        BRB      3$                              ; 0429
  54              85 9A 0000F 1$:    MOVZBL   (IPTR)+, CHAR                   ; 0431
  82              54 90 00012        MOVB     CHAR, (OPTR)+                   ; 0432
                  54 D5 00015        TSTL     CHAR                            ; 0434
```

```
                                      48  12 00017        BNEQ    3$                                  ;
         0000'  CF     0083   8F      00  3A 00019        LOCC    #0, #131, OUTPUT_RECORD             ; 0439
                                      02  12 00021        BNEQ    2$                                  ;
                                      51  D4 00023        CLRL    R1                                  ;
                        50    0000'   CF  9E 00025 2$:    MOVAB   OUTPUT_RECORD, R0                   ;
                 6E     51            50  A3 0002A        SUBW3   R0, R1, $IOB$OUTPUT                 ;
                   02   AE            0E  90 0002E        MOVB    #14, $IOB$OUTPUT+2                  ;
                   03   AE            01  90 00032        MOVB    #1, $IOB$OUTPUT+3                   ;
                   04   AE    0000'   CF  D0 00036        MOVL    OUTPUT_PTR, $IOB$OUTPUT+4           ;
                0000'   CF            6E  9E 0003C        MOVAB   $IOB$ODTPUT, IOB$+68                ;
                0000'   CF            07  90 00041        MOVB    #7, IOB$+44                         ;
                           00000000G  EF  9F 00046        PUSHAB  XPO$FAILURE                         ;
                                      7E  D4 0004C        CLRL    -(SP)                               ;
                              0000'   CF  9F 0004E        PUSHAB  IOB$                                ;
            00000000G   EF            03  FB 00052        CALLS   #3, XPO$PUT                         ;
                        52    0000'   CF  D0 00059        MOVL    OUTPUT_PTR, OPTR                    ; 0440
                        82            09  90 0005E        MOVB    #9, (OPTR)+                         ; 0441
                        AB            53  F4 00061 3$:    SOBGEQ  I, 1$                               ; 0434
                        50            52  D0 00064        MOVL    OPTR, R0                            ; 0446
                                          04 00067        RET                                         ;
```

; Routine Size: 104 bytes,   Routine Base: $CODE$ + 04B9

```
;  451     0447  1 ROUTINE CONVARGLIST(TEMP_PTR): NOVALUE=
;  452     0448  1
;  453     0449  1 !++
;  454     0450  1 ! FUNCTIONAL DESCRIPTION:
;  455     0451  1 !       This routine gathers the argument list of a macro definition.
;  456     0452  1 !       It processes continuation lines.  It eliminates unnecessary
;  457     0453  1 !       characters and places the arguments with default values in ARG_LIST
;  458     0454  1 !       and a list of the formal names in CALL_LIST
;  459     0455  1 !
;  460     0456  1 ! PARAMETERS:
;  461     0457  1 !
;  462     0458  1 !       TEMP_PTR        =  The value of a CH pointer in the macro
;  463     0459  1 !                          definition line.
;  464     0460  1 !
;  465     0461  1 ! IMPLICIT OUTPUTS:
;  466     0462  1 !       ARG_LIST        =  String of formal arguments with default values
;  467     0463  1 !       ARG_PTR         =  String pointer to ARG_LIST
;  468     0464  1 !       ARG_LENGTH      =  Length of string in ARG_LIST
;  469     0465  1 !       CALL_LIST       =  String of formal arguments
;  470     0466  1 !       CALL_PTR        =  String pointer to CALL_LIST
;  471     0467  1 !       CALL_LENGTH     =  Length of string in CALL_LIST
;  472     0468  1 !       VAR_ARGS        =  True if the last formal argument was END_VARNUM_ARGS
;  473     0469  1 !
;  474     0470  1 ! ROUTINE VALUE:
;  475     0471  1 !       NONE
;  476     0472  1 !
;  477     0473  1 !--
;  478     0474  1
;  479     0475  2     BEGIN
;  480     0476  2     LOCAL
;  481     0477  2         LIST_PTR,       ! Input character string pointer
;  482     0478  2         CHAR,           ! Character from string
;  483     0479  2         STATE,          ! State of parse
;  484     0480  2                         !       0 = No argument seen yet
;  485     0481  2                         !       1 = Scanning formal name
;  486     0482  2                         !       2 = Scanning default value
;  487     0483  2         BEGIN_ARG;      ! Character string pointer to beginning of argument
;  488     0484  2
;  489     0485  2     MACRO
;  490   M 0486  2         TRY_ADD_DEFAULT=
;  491     0487  2             IF .STATE EQL 1 THEN (CH$WCHAR_A(%C'=', ARG_PTR);  CH$WCHAR_A(%C'0', ARG_PTR)) %;
;  492     0488  2
;  493     0489  2
;  494     0490  2
;  495     0491  2     ! Initialize
;  496     0492  2     !
;  497     0493  2     ARG_PTR = CH$PTR(ARG_LIST);
;  498     0494  2     CH$FILL(0, MAX_ARG_LIST, .ARG_PTR);
;  499     0495  2     CALL_PTR = CH$PTR(CALL_LIST);
;  500     0496  2     CH$FILL(0, MAX_ARG_LIST, .CALL_PTR);
;  501     0497  2     LIST_PTR = CH$PLUS(.TEMP_PTR, 3);
;  502     0498  2     VAR_ARGS = 0;
;  503     0499  2     STATE = 0;
;  504     0500  2
;  505     0501  2
;  506     0502  2
;  507     0503  2     ! Scan the argument list
```

```
;  508     0504  2     ! Exit when an end of line (not preceded by line continuation mark) is read
;  509     0505  2     !
;  510     0506  2     REPEAT
;  511     0507  3         BEGIN
;  512     0508  3         CHAR = CH$RCHAR_A(LIST_PTR);
;  513     0509  3         SELECTONE .CHAR OF
;  514     0510  3             SET
;  515     0511  3
;  516     0512  3             [0]:
;  517     0513  4                 BEGIN
;  518     0514  4                 !
;  519     0515  4                 ! End of list
;  520     0516  4                 !       Add a default value to the last argument if it did not
;  521     0517  4                 !       have one.
;  522     0518  4                 !       Set VAR_ARGS if the name of the last argument is
;  523     0519  4                 !       END_VARNUM_ARGS
;  524     0520  4                 !
;  525     0521  4                 TRY_ADD_DEFAULT;
;  526     0522  4                 IF .STATE GEQ 1
;  527     0523  4                 THEN
;  528     0524  4                     IF
;  529     0525  4                         CH$EQL(
;  530     0526  4                             CH$DIFF(.CALL_PTR, .BEGIN_ARG),
;  531     0527  4                             .BEGIN_ARG,
;  532     0528  4                             CH$LEN_PTR('END_VARNUM_ARGS'),
;  533     0529  4                             0)
;  534     0530  4                     THEN
;  535     0531  4                         VAR_ARGS = 1;
;  536     0532  4
;  537     0533  4                 EXITLOOP
;  538     0534  3                 END;
;  539     0535  3
;  540     0536  3             [%C'A' TO %C'Z', %C'0' TO %C'9', %C'_', %C'$']:
;  541     0537  4                 BEGIN
;  542     0538  4                 !
;  543     0539  4                 ! Symbol constituent
;  544     0540  4                 !       If STATE is zero, it is the first character of a formal name
;  545     0541  4                 !       STATE distinguishes scanning formal name from scanning default value
;  546     0542  4                 !
;  547     0543  4                 IF .STATE EQL 0
;  548     0544  4                 THEN
;  549     0545  5                     BEGIN
;  550     0546  5                     STATE = 1;
;  551     0547  5                     BEGIN_ARG = .CALL_PTR
;  552     0548  4                     END;
;  553     0549  4
;  554     0550  4                 IF .STATE EQL 1 THEN CH$WCHAR_A(.CHAR, CALL_PTR);
;  555     0551  4                 CH$WCHAR_A(.CHAR, ARG_PTR)
;  556     0552  3                 END;
;  557     0553  3
;  558     0554  3             [%C'=']:
;  559     0555  4                 BEGIN
;  560     0556  4                 !
;  561     0557  4                 ! Beginning of default value
;  562     0558  4                 !       Set STATE to indicate scanning default and collect character
;  563     0559  4                 !
;  564     0560  4                 STATE = 2;
```

```
;   565     0561  4                 CH$WCHAR_A(.CHAR, ARG_PTR)
;   566     0562  3                 END;
;   567     0563  3
;   568     0564  3             [%C',']:
;   569     0565  4                 BEGIN
;   570     0566  4                 !
;   571     0567  4                 ! End of argument
;   572     0568  4                 !       Add a default value to the argument if it did not have one
;   573     0569  4                 !       Collect the character
;   574     0570  4                 !       Reset STATE to indicate not within an argument
;   575     0571  4                 !
;   576     0572  4                 TRY_ADD_DEFAULT;
;   577     0573  4                 CH$WCHAR_A(.CHAR, CALL_PTR);
;   578     0574  4                 CH$WCHAR_A(.CHAR, ARG_PTR);
;   579     0575  4                 STATE = 0
;   580     0576  3                 END;
;   581     0577  3
;   582     0578  3             [%C'-']:
;   583     0579  4                 BEGIN
;   584     0580  4                 !
;   585     0581  4                 ! Line continuation indicator
;   586     0582  4                 !       Get a new line and reset LIST_PTR
;   587     0583  4                 !
;   588     0584  4                 CH$WCHAR_A( 0, CALL_PTR );
;   589     0585  4                 CH$WCHAR_A( 0, ARG_PTR );
;   590     0586  4
;   591     0587  4                 $XPO_GET(IOB=INPUT);
;   592     0588  4                 INPUT_LENGTH = .INPUT[IOB$H_STRING];
;   593     0589  4                 INPUT_PTR = .INPUT[IOB$A_STRING];
;   594     0590  4                 CH$COPY(.INPUT_LENGTH, .INPUT_PTR,
;   595     0591  4                         0,
;   596     0592  4                         MAX_REC_SIZE, .OUTPUT_PTR);
;   597     0593  4                 LIST_PTR = .OUTPUT_PTR
;   598     0594  3                 END;
;   599     0595  3
;   600     0596  3             [%C';']:
;   601     0597  3                 !
;   602     0598  3                 ! MACRO comment character
;   603     0599  3                 !       The remainder of the line is "ignorable"
;   604     0600  3                 !
;   605     0601  3                 EXITLOOP
;   606     0602  3             TES
;   607     0603  2         END;
;   608     0604  2
;   609     0605  2
;   610     0606  2         ! Set up implicit outputs as noted
;   611     0607  2         ! If variable number of arguments, then replace CALL_LIST by "%REMAINING"
;   612     0608  2         !   since this is the call argument
;   613     0609  2         !
;   614     0610  2         ARG_LENGTH = CH$DIFF(.ARG_PTR, CH$PTR(ARG_LIST));
;   615     0611  2         ARG_PTR = CH$PTR(ARG_LIST);
;   616     0612  2
;   617     0613  2         IF .VAR_ARGS
;   618     0614  2         THEN
;   619     0615  3             BEGIN
;   620     0616  3             CH$COPY(
;   621     0617  3                 CH$LEN_PTR('%REMAINING'),
```

```
;  622        0618 3                    0,
;  623        0619 3                    MAX_ARG_LIST,
;  624        0620 3                    CH$PTR(CALL_LIST));
;  625        0621 3            CALL_LENGTH = %CHARCOUNT('%REMAINING');
;  626        0622 3            END
;  627        0623 2        ELSE
;  628        0624 2            CALL_LENGTH = CH$DIFF(.CALL_PTR, CH$PTR(CALL_LIST));
;  629        0625 2
;  630        0626 2        CALL_PTR = CH$PTR(CALL_LIST)
;  631        0627 1        END;
;  INFO#250              L1:0526
; Referenced LOCAL symbol BEGIN_ARG is probably not initialized


                                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

53 47 52 41 5F 4D 55 4E 52 41 56 5F 44 4E 45  00114 P.ABA:  .ASCII  \END_VARNUM_ARGS\<0>                    ;
                                          00  00123                                                         ;
         00 00 47 4E 49 4E 49 41 4D 45 52 25  00124 P.ABB:  .ASCII  \%REMAINING\<0><0>                      ;

                                                IOB$=                INPUT

                                                            .PSECT  $CODE$,NOWRT,2

                                     07FC 00000 CONVARGLIST:
                                                            .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10          ; 0447
                      5A    0000'  CF  9E 00002             MOVAB   ARG_PTR, R10
                      6A      08   AA  9E 00007             MOVAB   ARG_LIST, ARG_PTR                         ; 0493
  0FFF  8F      00    6E           00  2C 0000B             MOVC5   #0, (SP), #0, #4095, @ARG_PTR             ; 0494
                              00   BA     00012                                                               ;
                1008  CA    1010   CA  9E 00014             MOVAB   CALL_LIST, CALL_PTR                       ; 0495
  0FFF  8F      00    6E           00  2C 0001B             MOVC5   #0, (SP), #0, #4095, @CALL_PTR            ; 0496
                            1008   DA     00022                                                               ;
                59      04    AC   03  C1 00025             ADDL3   #3, TEMP_PTR, LIST_PTR                    ; 0497
                            2010   CA  D4 0002A             CLRL    VAR_ARGS                                  ; 0498
                                   57  D4 0002E 1$:         CLRL    STATE                                     ; 0499
                      56           89  9A 00030 2$:         MOVZBL  (LIST_PTR)+, CHAR                         ; 0508
                                   2D  12 00033             BNEQ    5$                                        ; 0512
                      01           57  D1 00035             CMPL    STATE, #1                                 ; 0513
                                   0C  12 00038             BNEQ    3$                                        ;
                00    BA           3D  90 0003A             MOVB    #61, @ARG_PTR                             ;
                                   6A  D6 0003E             INCL    ARG_PTR                                   ;
                00    BA           30  90 00040             MOVB    #48, @ARG_PTR                             ;
                                   6A  D6 00044             INCL    ARG_PTR                                   ;
                                   57  D5 00046 3$:         TSTL    STATE                                     ; 0522
                                   15  15 00048             BLEQ    4$                                        ;
          50    1008  CA           58  C3 0004A             SUBL3   BEGIN_ARG, CALL_PTR, R0                   ; 0526
     0F   00          68           50  2D 00050             CMPC5   R0, (BEGIN_ARG), #0, #15, P.ABA           ;
                            0000'  CF     00055                                                               ;
                                   05  12 00058             BNEQ    4$                                        ;
                2010  CA           01  D0 0005A             MOVL    #1, VAR_ARGS                              ; 0531
                                 00CF  31 0005F 4$:         BRW     17$                                       ; 0513
                      24           56  D1 00062 5$:         CMPL    CHAR, #36                                 ; 0536
                                   25  13 00065             BEQL    8$                                        ;
                      30           56  D1 00067             CMPL    CHAR, #48                                 ;
```

```
                                           05  19 0006A       BLSS    6$                                   ;
                        39                 56  D1 0006C       CMPL    CHAR, #57                            ;
                                           1B  15 0006F       BLEQ    8$                                   ;
               00000041 8F                 56  D1 00071 6$:   CMPL    CHAR, #65                            ;
                                           09  19 00078       BLSS    7$                                   ;
               0000005A 8F                 56  D1 0007A       CMPL    CHAR, #90                            ;
                                           09  15 00081       BLEQ    8$                                   ;
               0000005F 8F                 56  D1 00083 7$:   CMPL    CHAR, #95                            ;
                                           1C  12 0008A       BNEQ    10$                                  ;
                                           57  D5 0008C 8$:   TSTL    STATE                                ; 0543
                                           08  12 0008E       BNEQ    9$                                   ;
                        57                 01  D0 00090       MOVL    #1, STATE                            ; 0546
                        58          1008   CA  D0 00093       MOVL    CALL_PTR, BEGIN_ARG                  ; 0547
                        01                 57  D1 00098 9$:   CMPL    STATE, #1                            ; 0550
                                           13  12 0009B       BNEQ    11$                                  ;
                   1008 DA                 56  90 0009D       MOVB    CHAR, @CALL_PTR                      ;
                                    1008   CA  D6 000A2       INCL    CALL_PTR                             ;
                                           08  11 000A6       BRB     11$                                  ; 0551
                        3D                 56  D1 000A8 10$:  CMPL    CHAR, #61                            ; 0554
                                           0B  12 000AB       BNEQ    12$                                  ;
                        57                 02  D0 000AD       MOVL    #2, STATE                            ; 0560
                     00 BA                 56  90 000B0 11$:  MOVB    CHAR, @ARG_PTR                       ; 0561
                                           6A  D6 000B4       INCL    ARG_PTR                              ;
                                           71  11 000B6       BRB     15$                                  ; 0509
                        2C                 56  D1 000B8 12$:  CMPL    CHAR, #44                            ; 0564
                                           23  12 000BB       BNEQ    14$                                  ;
                        01                 57  D1 000BD       CMPL    STATE, #1                            ; 0565
                                           0C  12 000C0       BNEQ    13$                                  ;
                     00 BA                 3D  90 000C2       MOVB    #61, @ARG_PTR                        ;
                                           6A  D6 000C6       INCL    ARG_PTR                              ;
                     00 BA                 30  90 000C8       MOVB    #48, @ARG_PTR                        ;
                                           6A  D6 000CC       INCL    ARG_PTR                              ;
                   1008 DA                 56  90 000CE 13$:  MOVB    CHAR, @CALL_PTR                      ; 0573
                                    1008   CA  D6 000D3       INCL    CALL_PTR                             ;
                     00 BA                 56  90 000D7       MOVB    CHAR, @ARG_PTR                       ; 0574
                                           6A  D6 000DB       INCL    ARG_PTR                              ;
                                      FF4E 31 000DD           BRW     1$                                   ; 0575
                        2D                 56  D1 000E0 14$:  CMPL    CHAR, #45                            ; 0578
                                           47  12 000E3       BNEQ    16$                                  ;
                                    1008   DA  94 000E5       CLRB    @CALL_PTR                            ; 0584
                                    1008   CA  D6 000E9       INCL    CALL_PTR                             ;
                                      00   BA  94 000ED       CLRB    @ARG_PTR                             ; 0585
                                           6A  D6 000F0       INCL    ARG_PTR                              ;
                   2134 CA                 06  90 000F2       MOVB    #6, IOB$+44                          ; 0587
                             00000000G     EF  9F 000F7       PUSHAB  XPO$FAILURE                          ;
                                           7E  D4 000FD       CLRL    -(SP)                                ;
                                    2108   CA  9F 000FF       PUSHAB  IOB$                                 ;
              00000000G EF                 03  FB 00103       CALLS   #3, XPO$GET                          ;
                   FF74 CA          213C   CA  3C 0010A       MOVZWL  INPUT+52, INPUT_LENGTH               ; 0588
                   FF70 CA          2140   CA  D0 00111       MOVL    INPUT+56, INPUT_PTR                  ; 0589
   0082   8F         00 FF70 DA     FF74   CA  2C 00118       MOVC5   INPUT_LENGTH, @INPUT_PTR, #0, #130, - ; 0592
                                      FC   BA     00123               @OUTPUT_PTR                          ;
                        59            FC   AA  D0 00125       MOVL    OUTPUT_PTR, LIST_PTR                 ; 0593
                                      FF04 31 00129 15$:      BRW     2$                                   ;
                        3B                 56  D1 0012C 16$:  CMPL    CHAR, #59                            ; 0596
                                           F8  12 0012F       BNEQ    15$                                  ;
                        50            08   AA  9E 00131 17$:  MOVAB   ARG_LIST, R0                         ; 0610
```

```
             04    AA          6A          50  C3 00135         SUBL3   R0, ARG_PTR, ARG_LENGTH                       ;
                               6A    08    AA  9E 0013A         MOVAB   ARG_LIST, ARG_PTR                             ; 0611
                               13  2010    CA  E9 0013E         BLBC    VAR_ARGS, 18$                                 ; 0613
  0FFF   8F          00  0000'  CF          0A  2C 00143         MOVC5   #10, P.ABB, #0, #4095, CALL_LIST              ; 0620
                                   1010    CA     0014C                                                               ;
                         100C   CA          0A  D0 0014F         MOVL    #10, CALL_LENGTH                              ; 0621
                                            0D  11 00154         BRB     19$                                           ; 0613
                               50  1010    CA  9E 00156  18$:   MOVAB   CALL_LIST, R0                                 ; 0624
             100C    CA  1008   CA          50  C3 0015B         SUBL3   R0, CALL_PTR, CALL_LENGTH                     ;
                         1008   CA  1010    CA  9E 00163  19$:   MOVAB   CALL_LIST, CALL_PTR                           ; 0626
                                                04 0016A         RET                                                   ; 0627

; Routine Size:  363 bytes,    Routine Base:  $CODE$ + 0521
```

```
;  633          0628  1 END                          ! End of module MARBLI
;  634          0629  0 ELUDOM



;                                PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $OWN$                         9292  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                         304  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                        1676  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                             Library Statistics
;
;                                      -------- Symbols --------      Pages       Processing
;       File                           Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]XPORT.L32;1       590       116        19       252          00:00.1


; Information:  1
; Warnings:     0
; Errors:       0


;                                    COMMAND QUALIFIERS

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

; Size:          1676 code + 9596 data bytes
; Run Time:        00:39.3
; Elapsed Time:    01:50.6
; Lines/CPU Min:     959
; Lexemes/CPU-Min: 74449
; Memory Used:  247 pages
; Compilation Complete
```

0233 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

MELDR

MDL32

LALOAD
MAP

MDL32
MAP

XFLOADER
MAP

LALOADER
LIS

XFLOADER
LIS

LADAMCODE
LIS

LALOAD
LIS

MARBLI

MARBLI
MAP

LALOADER
MAP

LAAMRMCODE
LIS

LAADMCODE
LIS

MARBLI
LIS

LADMDT
LIS