M 5

```
MMM         MMM     AAAAAAAAA        CCCCCCCCCCCC  RRRRRRRRRRRR        000000000
MMM         MMM     AAAAAAAAA        CCCCCCCCCCCC  RRRRRRRRRRRR        000000000
MMM         MMM     AAAAAAAAA        CCCCCCCCCCCC  RRRRRRRRRRRR        000000000
MMMMMM   MMMMMM  AAA         AAA  CCC              RRR         RRR  000         000
MMMMMM   MMMMMM  AAA         AAA  CCC              RRR         RRR  000         000
MMMMMM   MMMMMM  AAA         AAA  CCC              RRR         RRR  000         000
MMM   MMM   MMM  AAA         AAA  CCC              RRR         RRR  000         000
MMM   MMM   MMM  AAA         AAA  CCC              RRR         RRR  000         000
MMM   MMM   MMM  AAA         AAA  CCC              RRR         RRR  000         000
MMM         MMM  AAA         AAA  CCC              RRRRRRRRRRRR     000         000
MMM         MMM  AAA         AAA  CCC              RRRRRRRRRRRR     000         000
MMM         MMM  AAA         AAA  CCC              RRRRRRRRRRRR     000         000
MMM         MMM  AAAAAAAAAAAAAAA  CCC              RRR   RRR        000         000
MMM         MMM  AAAAAAAAAAAAAAA  CCC              RRR   RRR        000         000
MMM         MMM  AAAAAAAAAAAAAAA  CCC              RRR   RRR        000         000
MMM         MMM  AAA         AAA  CCC              RRR      RRR     000         000
MMM         MMM  AAA         AAA  CCC              RRR      RRR     000         000
MMM         MMM  AAA         AAA  CCC              RRR      RRR     000         000
MMM         MMM  AAA         AAA     CCCCCCCCCCCC  RRR         RRR     000000000
MMM         MMM  AAA         AAA     CCCCCCCCCCCC  RRR         RRR     000000000
MMM         MMM  AAA         AAA     CCCCCCCCCCCC  RRR         RRR     000000000
```

```
_$2

Sym
---
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC

MAC
MAC
MAC
MAC

MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
MAC
```

D 7

```
**FILE**ID**MACLIB




MM      MM    AAAAAA      CCCCCCCC  LL            IIIIII    BBBBBBBB
MM      MM    AAAAAA      CCCCCCCC  LL            IIIIII    BBBBBBBB
MMMM  MMMM  AA      AA  CC          LL              II      BB      BB
MMMM  MMMM  AA      AA  CC          LL              II      BB      BB
MM  MM  MM  AA      AA  CC          LL              II      BB      BB
MM  MM  MM  AA      AA  CC          LL              II      BB      BB
MM      MM  AA      AA  CC          LL              II      BBBBBBBB
MM      MM  AA      AA  CC          LL              II      BBBBBBBB
MM      MM  AAAAAAAAAA  CC          LL              II      BB      BB
MM      MM  AAAAAAAAAA  CC          LL              II      BB      BB
MM      MM  AA      AA  CC          LL              II      BB      BB       ....
MM      MM  AA      AA  CC          LL              II      BB      BB       ....
MM      MM  AA      AA    CCCCCCCC  LLLLLLLLLL    IIIIII    BBBBBBBB         ....
MM      MM  AA      AA    CCCCCCCC  LLLLLLLLLL    IIIIII    BBBBBBBB         ....


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

```
MA
Sy

$$
$C
AR
AU
BL
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CN
CR
DA
DA
DA
DA
DB
DC
DC
DC
DD
DE
DE
DG
DI
DI
DL
DM
DM
DO
DO
DO
DP
DP
DP
DS
DS
DS
DT
DU
DU
DU
DU
DU
DU
DU
```

Table of contents

```
0000         .TITLE  MAC$MACLIB        MACRO LIBRARY PROCESSOR
0000         .IDENT  'V04-000'
0000
0000   ;
0000   ;****************************************************************************
0000   ;*                                                                          *
0000   ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000   ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000   ;*  ALL RIGHTS RESERVED.                                                    *
0000   ;*                                                                          *
0000   ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000   ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000   ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000   ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000   ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000   ;*  TRANSFERRED.                                                            *
0000   ;*                                                                          *
0000   ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000   ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000   ;*  CORPORATION.                                                            *
0000   ;*                                                                          *
0000   ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000   ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000   ;*                                                                          *
0000   ;*                                                                          *
0000   ;****************************************************************************
0000   ;
0000
0000   ;++
0000   ; FACILITY:      VAX MACRO ASSEMBLER OBJECT LIBRARY
0000   ;
0000   ; ABSTRACT:
0000   ;
0000   ; The VAX-11 MACRO assembler translates MACRO-32 source code into object
0000   ; modules for input to the VAX-11 LINKER.
0000   ;
0000   ; ENVIRONMENT:  USER MODE
0000   ;
0000   ; AUTHOR: Benn Schreiber, CREATION DATE: 24-AUG-78
0000   ;
0000   ; MODIFIED BY:
0000   ;
0000   ;       V03.01  MTR0016         Mike Rhodes     07-Jun-1982
0000   ;               Add logic to MAC$IMPLMCALL to save/modify/restore the
0000   ;               listing directive for listing macro definitions.  This
0000   ;               corrects a problem where the .NOSHOW MD directive caused
0000   ;               the listing to lose the invocation lines or other assorted
0000   ;               parts when there were no macros explicitly defined in the
0000   ;               program text and the macros were being defined from a library.
0000   ;               Also included are fixes for broken branch destinations.
0000   ;
0000   ;       V02.17  MTR0001         Mike Rhodes     09-Nov-1981
0000   ;               Add default name descriptor use in opening the system
0000   ;               library STARLET.
0000   ;
0000   ;       V02.16  PCG0007         P. George       20-Aug-1981
0000   ;               Report .LIBRARY errors in listing and with exit status.
```

```
0000     ;
0000     ;       V01.15  RN0023          R. Newland       3-Nov-1979
0000     ;               New message codes to get error messages from system
0000     ;               message file.
0000     ;
0000     ;       V01.14  RN0011          R. Newland               11-Sep-1979
0000     ;               New librarian support
0000     ;
0000     ;       V01.11  004             B. Schreiber     10-JAN-1979
0000     ;               Correct "illegal block size" error if macro library
0000     ;               has MNT entries allocated but none used.
0000     ;       V01.12  008             B. Schreiber     23-JAN-1979
0000     ;               Clear index number of pages for SYSLIB FDB on setup.
0000     ;       V01.13  RN0003          R. Newland       18-Feb-1979
0000     ;               Correct initial values of macro library binary
0000     ;               chop routine
0000     ;--
```

```
                   0000          .SBTTL  DECLARATIONS
                   0000    ;
                   0000    ; INCLUDE FILES:
                   0000    ;
                   0000
                   0000    ;
                   0000    ; MACROS:
                   0000    ;
                   0000
                   0000            $MAC_CTLFLGDEF                  ;DEFINE CONTROL FLAGS
                   0000            $MAC_GRAMMARDEF                 ;DEFINE TERMINAL GRAMMAR SYMBOLS
                   0000            $MAC_MLFDEF                     ; Define MLF offsets
                   0177            $MAC_GENVALDEF                  ;DEFINE GENERAL VALUES
                   0177            $MAC_INTCODDEF                  ;DEFINE INT. FILE CODES
                   0177            $MAC_SYMBLKDEF                  ;DEFINE SYMBOL BLOCK OFFSETS
                   0000            $NAMDEF                         ;DEFINE NAM OFFSETS
                   0000            $RABDEF                         ;DEFINE RAB OFFSETS
                   0000            $RMEDEF                         ;DEFINE RME
                   0000            $FABDEF                         ;DEFINE FAB OFFSETS
                   0000            $MACMSGDEF                      ; Define message codes
                   0000
                   0000    ;
                   0000    ; LOCAL DATA
                   0000    ;
                   0000
               00000000            .PSECT  MAC$RO_DATA,NOEXE,NOWRT,GBL,LONG
                   0000
                   0000   MLF_ARGLIST:
                   0000            $ARGLST 2,MLF_SIZE,MAC$GL_BASEADDR
                   000C   ;
                   000C   MLF_SIZE:
         00000177  000C            .LONG   MLF$K_BLKSIZ
                   0010
               00000000            .PSECT  MAC$RW_DATA,NOEXE,LONG
                   0000
                   0000   MAC$TMPSYMDS:                            ; Temp sym descriptor (used for
00000001'00000000  0000            .LONG   0,MAC$AB_TMPSYM+1       ; LBR$LOOKUP_KEY calls)
                   0008
               00000000            .PSECT  MAC$RO_CODE_MAC,NOWRT,GBL,LONG
```

```
                             0000          .SBTTL  MCALL                      EXPLICIT MACRO CALL ROUTINE
                             0000
                             0000  ;++
                             0000  ; FUNCTIONAL DESCRIPTION:
                             0000  ;
                             0000  ;       THIS ROUTINE IS CALLED WHEN THE .MCALL DIRECTIVE IS FOUND.
                             0000  ;       ANY MACRO NAMES FOLLOWING THE DIRECTIVE THAT ARE NOT DEFINED
                             0000  ;       ARE DEFINED BY DOING IMPLICIT MACRO CALLS ON THEM.
                             0000  ;
                             0000  ;--
                             0000
                             0000  MCALL::                                            ;DIRECITVE = KMCALL
   50   0000'CF     01   C3  0000          SUBL3   #1,W^MAC$GL_LINEPT,R0              ;SET UP ERROR POINTER
        0000'CF     50   D0  0006          MOVL    R0,W^MAC$GL_ERRPT                  ;....
                  FFF2'  30  000B          BSBW    MAC$SYMSCNUP                       ;SCAN A MACRO NAME
                 08 50   E8  000E          BLBS    R0,10$                             ;BRANCH IF WE FOUND ONE
                             0011          $MAC_ERR DIRSYNX                           ; Else directive syntax error
                  FFE7'  31  0016          BRW     MAC$ERRORPT                        ;ISSUE ERROR AND RETURN
   53   0000'CF          9E  0019  10$:    MOVAB   W^MAC$AL_UMCHSHTB,R3               ;LOOK UP NAME IN MACRO HASH TABLE
                  FFDF'  30  001E          BSBW    MAC$SRCSYMTAB                      ;IF IT IS THERE WE CAN SAVE SOME TIME
                 0F 50   E8  0021          BLBS    R0,20$                             ;BRANCH IF FOUND
                    21   10  0024          BSBB    MAC$IMPLMCALL                      ;NO--DO AN IMPLICIT MACRO CALL
             0D     58   D1  0026          CMPL    R8,#MACTXT                         ;DID WE FIND IT?
                    08   13  0029          BEQL    20$                                ;IF EQL YES
                             002B          $MAC_ERR CANTLOCMAC                        ; No--set message code
                  FFCD'  30  0030          BSBW    MAC$ERRORPT                        ;ISSUE MESSAGE TO PASS 2
                  FFCA'  30  0033  20$:    BSBW    MAC$SKIPSP                         ;SKIP SPACES
             2C     5A   91  0036          CMPB    R10,#^A/,/                         ;STOP ON A COMMA?
                    06   12  0039          BNEQ    30$                                ;IF NEQ NO
                  FFC2'  30  003B          BSBW    MAC$GETCHR                         ;YES--SKIP IT
                  FFBF'  30  003E          BSBW    MAC$SKIPSP                         ;THEN SKIP SPACES
             0D     5A   91  0041  30$:    CMPB    R10,#CR                            ;GET TO EOL?
                    BA   12  0044          BNEQ    MCALL                              ;IF NEQ NO--KEEP GOING
                         05  0046          RSB                                        ;YES--MCALL IS DONE
                         0047         .SBTTL  MAC$IMPLMCALL   IMPLICIT MACRO CALL ROUTINE
                         0047
                         0047  ;++
                         0047  ; FUNCTIONAL DESCRIPTION:
                         0047  ;
                         0047  ;        THIS ROUTINE IS CALLED EITHER BY "MCALL" OR BY THE MAC$SYMBOL
                         0047  ;        ROUTINE WHEN AN UNIDENTIFIED NAME IS DETECTED.  THE IN-CORE
                         0047  ;        INDICES OF ALL "KNOWN" MACRO LIBRARIES ARE SEARCHED STARTING
                         0047  ;        WITH THE LAST ONE SPECIFIED.  THE FOLLOWING METHOD IS USED:
                         0047  ;
                         0047  ;                1) THE IN-CORE INDEX OF EACH MLB IS SEARCHED USING
                         0047  ;                   A MATCHC INSTRUCTION.  IF THE NAME IS FOUND, THE
                         0047  ;                   MLB IS OPENED AND THE MACRO IS DEFINED.
                         0047  ;
                         0047  ;                2) IF THE NAME IS NOT FOUND IN ANY OF THE MACRO
                         0047  ;                   LIBRARIES, A TOKEN OF "ERR03" IS RETURNED.
                         0047  ;
                         0047  ;--
                         0047
                         0047  MAC$IMPLMCALL::
                 57   DD 0047         PUSHL   R7                              ;SAVE R7
        56  0000'CF   9E 0049         MOVAB   W^MAC$AB_TMPSYM,R6              ;POINT TO THE NAME WE ARE LOOKING FOR
      0000'CF     66  9B 004E         MOVZBW  (R6),W^MAC$TMPSYMDS             ; Set up descriptor for macro name
        57  0000'CF   9E 0053         MOVAB   W^MAC$GL_MLB_QUE,R7             ;POINT TO MLB QUE HEADER
                         0058
                         0058  NEXT_MLB:
              57  67  D0 0058         MOVL    (R7),R7                         ;LINK TO NEXT MLB FDB
     00000000'8F  57  D1 005B         CMPL    R7,#MAC$GL_MLB_QUE              ;ARE WE DONE?
                  07  12 0062         BNEQ    10$                             ;IF NEQ NO
                 57 8ED0 0064         POPL    R7                              ;YES--RESTORE R7
              58  03  D0 0067         MOVL    #ERR03,R8                       ;RETURN ERROR TOKEN
                  05     006A         RSB
                         006B  ;
                         006B  ; Call librarian procedure to look up macro name
                         006B  ;
                         006B  10$:
          0000'CF     9F 006B         PUSHAB  W^MAC$GL_TXTRFA                 ; Address to store text RFA
          0000'CF     9F 006F         PUSHAB  W^MAC$TMPSYMDS                  ; Address of name descriptor
               14 A7  9F 0073         PUSHAB  MLF$L_CTINDEX(R7)               ; Address of control table index
     00000000'GF  03  FB 0076         CALLS   #3,G^LBR$LOOKUP_KEY             ; Look-up macro name
              D8  50  E9 007D         BLBC    R0,NEXT_MLB                     ; Not found if LBC
      0000'CF     57  D0 0080         MOVL    R7,W^MAC$GL_MLFPTR              ; Save MLF pointer
                         0085  ;
            FF78'     30 0085         BSBW    MAC$ALL_1_PAGE                  ;ALLOCATE A VIRTUAL PAGE
               50     DD 0088         PUSHL   R0                              ;SAVE ITS ADDRESS
 60   0000'CF   0086 8F  28 008A      MOVC3   #LST$K_BUFSIZ,W^MAC$AB_LINEBF,(R0) ;COPY CURRENT LINE OUT
        83  0000'CF   D0 0092         MOVL    W^MAC$GL_LINENUM,(R3)+          ;SAVE CURRENT STATE
 83   6B   FFFFFFF7 8F  CB 0097       BICL3   #^C<FLG$M_CONT>,(R11),(R3)+ ;SAVE CONTINUATION STATE
           83     5A  D0 009F         MOVL    R10,(R3)+                       ;SAVE CURRENT CHARACTER
        83  0000'CF   D0 00A2         MOVL    W^MAC$GL_LINEPT,(R3)+           ;SAVE LINE POINTER
        83  0000'CF   D0 00A7         MOVL    W^MAC$GL_LINELN,(R3)+           ;SAVE LINE LENGTH
            FF51'     30 00AC         BSBW    MAC$ALL_1_PAGE                  ;ALLOCATE AN INPUT BLOCK
        60  0000'CF   D0 00AF         MOVL    W^MAC$GL_INPUTP,(R0)            ;LINK TO LAST INPUT BLOCK
      0000'CF     80  DE 00B4         MOVAL   (R0)+,W^MAC$GL_INPUTP           ;MAKE NEW BLOCK CURRENT BLOCK
                  80  D4 00B9         CLRL    (R0)+                           ;CLEAR TEXT LINK
        80  017F'CF   9E 00BB         MOVAB   W^MAC$GET_MLB_LIN,(R0)+ ;SET NEW LINE ROUTINE
                         00C0  ;
```

```
                           00C0   ; THE REST OF THE INPUT BLOCK IS NOT USED. HENCE IT IS NOT INITIALIZED.
                           00C0   ;
          5A   0D     9A   00C0            MOVZBL  #CR,R10                  ;FORCE READING OF NEW LINE
             FF3A'    30   00C3            BSBW    MAC$GETCHR               ;GET FIRST CHARACTER OF MACRO
             FF37'    30   00C6            BSBW    MAC$SYMSCNUP             ;SCAN FIRST SYMBOL
             03 50    E8   00C9            BLBS    R0,20$                   ; Branch if symbol was found
              00A5    31   00CC            BRW     MAC_LIB_FMT_ERR          ; ELSE If not found then error
     53   00000000'EF 9E   00CF   20$:     MOVAB   MAC$AL_PRMHSHTB,R3       ;LOOK IN PERMANENT SYMBOL TABLE
             FF27'    30   00D6            BSBW    MAC$SRCSYMTAB            ;...
             03 50    E8   00D9            BLBS    R0,30$                   ; Branch if symbol was found
              0095    31   00DC            BRW     MAC_LIB_FMT_ERR          ; ELSE, If not found then error
     50 8F   0B A1    91   00DF   30$:     CMPB    SYM$B_TOKEN(R1),#KMACRO  ;WAS IT ".MACRO"?
                03    13   00E4            BEQL    40$                      ; If EQL yes.
              008B    31   00E6            BRW     MAC_LIB_FMT_ERR          ; If NEQ no -- bad library
       00000005'GF    DD   00E9   40$:     PUSHL   G^LST$G_MACRODEF+SYM$L_VAL         ;SAVE THE LISTING STATE,
   0005'CF   FF 8F    98   00EF            CVTBL   #-1,W^LST$G_MACRODEF+SYM$L_VAL     ;ENABLE LISTING MACRO DEF's.
                           00F5            $INTOUT_LW      INT$_SETLONG,-   ;INDICATE THE (POSSIBLE) SWITCH...
                           00F5                    <G^LST$G_MACRODEF+SYM$L_VAL, #LST$G_MACRODEF+SYM$L_VAL> ;...TO PASS
             FEF6'    30   0107            BSBW    MACRO                    ; Define the MACRO
                           010A   ;
                           010A   ; NOW RESTORE THINGS TO PRE-MLB READING STATE
                           010A   ;
     00000005'GF 8ED0      010A            POPL    G^LST$G_MACRODEF+SYM$L_VAL ;RETRIEVE THE PRIOR LISTING STATE
                           0111            $INTOUT_LW      INT$_SETLONG,-   ;INDICATE THE (POSSIBLE) SWITCH...
                           0111                    <G^LST$G_MACRODEF+SYM$L_VAL, #LST$G_MACRODEF+SYM$L_VAL> ;...TO PASS
          0000'CF     D6   0123            INCL    W^MAC$GL_MLB_MDF         ;COUNT MACRO DEFINED FROM MACRO LIBRARY
            08 A7     D6   0127            INCL    MLF$L_MCDEF(R7)          ;ALSO COUNT IN MLB FDB
     50   0000'CF     D0   012A            MOVL    W^MAC$GL_INPUTP,R0       ;POINT TO THE INPUT BLOCK
   0000'CF     60     D0   012F            MOVL    (R0),W^MAC$GL_INPUTP     ;RESTORE LAST INPUT BLOCK
             FEC9'    30   0134            BSBW    MAC$DEA_1_PAGE           ;DEALLOCATE INPUT BLOCK
       50      6E     D0   0137            MOVL    (SP),R0                  ;GET ADDRESS OF SAVED LINE BLOCK
0000'CF  60  0086 8F  28   013A            MOVC3   #LST$K_BUFSIZ,(R0),W^MAC$AB_LINEBF ;RESTORE INPUT LINE
   0000'CF     81     D0   0142            MOVL    (R1)+,W^MAC$GL_LINENUM   ;RESTORE LINE NUMBER
               81     D5   0147            TSTL    (R1)+                    ;CHECK CONT FLAG
               06     13   0149            BEQL    50$                      ;IF EQL CLEAR IT
        06 6B  03     E3   014B            BBCS    #FLG$V_CONT,(R11),60$    ;NO--SET IT
               04     11   014F            BRB     60$
        00 6B  03     E5   0151   50$:     BBCC    #FLG$V_CONT,(R11),60$    ;CLEAR CONT FLAG
          5A   81     D0   0155   60$:     MOVL    (R1)+,R10                ;RESTORE CURRENT CHARACTER
   0000'CF     81     D0   0158            MOVL    (R1)+,W^MAC$GL_LINEPT    ;RESTORE LINE POINTER
   0000'CF     81     D0   015D            MOVL    (R1)+,W^MAC$GL_LINELN    ;RESTORE LINE LENGTH
               50   8ED0   0162            POPL    R0                       ;RESTORE ADDRESS OF SAVE PAGE
               57   8ED0   0165            POPL    R7                       ;RESTORE R7
          58   0D     D0   0168            MOVL    #MACTXT,R8               ;RETURN TOKEN FOR MACRO
             FE92'    30   016B            BSBW    MAC$DEA_1_PAGE           ;DEALLOCATE PAGE
     51   0000'CF     D0   016E            MOVL    W^MAC$GL_MACPTR,R1       ;RETURN PTR TO MNB IN R1
                      05   0173            RSB                              ;RETURN
                           0174
                           0174   ;
                           0174   ; MACRO LIBRARY FORMAT ERROR.  THE CODE TO REPORT THE ERROR IN PASS 2
                           0174   ; IS EMITTED TO THE INTERMEDIATE FILE, AND THE ASSEMBLY IS TERMINATED.
                           0174   ; PASS 2 IS THEN EXECUTED TO REPORT THE ERRORS.
                           0174   ;
                           0174   MAC_LIB_FMT_ERR:
                           0174            $MAC_ERR MACLBFMTER                      ; Get the message code
             FE84'    30   0179            BSBW    MAC$ERRORPT                      ;REPORT ERROR TO PASS 2
             FE81'    31   017C            BRW     MAC$ABORT_PASS1                  ;ABORT PASS 1
```

```
                                017F          .SBTTL  MAC$GET_MLB_LIN READ LINE FROM MACRO LIBRARY
                                017F
                                017F     ;++
                                017F     ; FUNCTIONAL DESCRIPTION:
                                017F     ;
                                017F     ;        THIS ROUTINE IS CALLED BY MAC$GETCHR WHEN IT IS TIME TO
                                017F     ;        READ ANOTHER MACRO DEFINITION LINE FROM A MACRO LIBRARY.
                                017F     ;        THE LINE IS PLACED IN THE INPUT BUFFER MAC$AB_LINELN.
                                017F     ;
                                017F     ;--
                                017F
                                017F     MAC$GET_MLB_LIN::
        51   0000'CF    D0      017F            MOVL    W^MAC$GL_MLFPTR,R1       ; Get current MLF pointer
             0000'CF    9F      0184            PUSHAB  W^MAC$GL_LINELN          ; Address to store line length
             0000'CF    9F      0188            PUSHAB  W^MAC$GQ_LINEBFDS        ; Address of buffer descriptor
                14 A1   9F      018C            PUSHAB  MLF$L_CTINDEX(R1)        ; Address of control table index
  00000000'GF      03   FB      018F            CALLS   #3,G^LBR$GET_RECORD      ; Get record
                DB 50   E9      0196            BLBC    R0,MAC_LIB_FMT_ERR       ;IF LBC THEN LIBRARY ERROR
             0000'CF    D6      0199     10$:   INCL    W^MAC$GL_MLB_GET         ;COUNT ANOTHER MLB GET
        50   0000'CF    D0      019D            MOVL    W^MAC$GL_LINELN,R0       ; Get line length
                   DB   13      01A2            BEQL    MAC$GET_MLB_LIN          ;SKIP NULL LINES
        51   0000'CF    9E      01A4            MOVAB   W^MAC$AB_LINEBF,R1       ;POINT TO THE LINE BUFFER
      0000'CF      51   D0      01A9            MOVL    R1,W^MAC$GL_LINEPT       ;SET UP THE LINE POINTER
      0000'CF      51   D0      01AE            MOVL    R1,W^MAC$GL_ERRPTX       ;AND THE ERROR TOKEN POINTER
         6041      0D   90      01B3            MOVB    #CR,(R0)[R1]             ;STORE CR AT END OF LINE
                        05      01B7            RSB
```

```
                                   01B8        .SBTTL  LIBRY                PROCESS .LIBRARY DIRECTIVE
                                   01B8
                                   01B8  ;++
                                   01B8  ; FUNCTIONAL DESCRIPTION:
                                   01B8  ;
                                   01B8  ;       THIS ROUTINE IS CALLED TO PROCESS THE .LIBRARY DIRECTIVE.
                                   01B8  ;       THE FILENAME WITHIN THE DELIMITERS IS SCANNED.  THE FILE
                                   01B8  ;       IS THEN OPENED, AND AN IN-CORE INDEX IS CREATED.  THE FDB
                                   01B8  ;       FOR THE NEW MLB IS THEN ADDED TO THE FRONT OF THE MLB QUEUE.
                                   01B8  ;
                                   01B8  ;--
                                   01B8
                                   01B8  LIBRY::                                           ;DIRECTIVE = KLIBRARY
                       58    DD    01B8          PUSHL   R8                                ;SAVE R8
                     FE43'   30    01BA          BSBW    MAC$SKIPSP                        ;FIND THE DELIMITER
                0D     5A    91    01BD          CMPB    R10,#CR                           ;DID WE GET TO END OF LINE?
                       07    12    01C0          BNEQ    10$                               ;IF NEQ NO
                                   01C2  5$:     $MAC_ERR DIRSYNX                          ; Yes-get message code
                       22    11    01C7          BRB     40$                               ;ISSUE ERROR AND RETURN
                56     5A    90    01C9  10$:    MOVB    R10,R6                            ;COPY THE DELIMITER
           58     0008'CF    9E    01CC          MOVAB   W^MAC$AB_TMPBUF+8,R8              ; Point to temp buffer to accumalate name
                                   01D1                                                    ; (a descriptor will be formed at start)
                6B     01    C8    01D1          BISL2   #FLG$M_ALLCHR,(R11)               ;PASS SEMI-COLONS IN CASE SOME
                                   01D4                                                    ;TURKEY PUTS A VERSION NUMBER ON
                                   01D4                                                    ;HIS LIBRARY FILE NAME
                     FE29'   30    01D4  20$:    BSBW    MAC$GETCHR                        ;GET A CHARACTER OF FILENAME
                56     5A    91    01D7          CMPB    R10,R6                            ;END OF FILENAME?
                       18    13    01DA          BEQL    50$                               ;IF EQL YES
                0D     5A    91    01DC          CMPB    R10,#CR                           ;BAD ENDING?
                       05    13    01DF          BEQL    30$                               ;IF EQL YES
                88     5A    90    01E1          MOVB    R10,(R8)+                         ;NO--STORE CHAR OF FILENAME
                       EE    11    01E4          BRB     20$                               ;CONTINUE LOOPING FOR NAME
                                   01E6  ;
                                   01E6  ; HERE IF CR BEFORE DELIMITER
                                   01E6  ;
                                   01E6  30$:    $MAC_ERR UNTERMARG                        ; Set message code
                     58 8ED0       01EB  40$:    POPL    R8                                ;RESTORE R8
                6B     01    CA    01EE          BICL2   #FLG$M_ALLCHR,(R11)               ;CLEAR ALLCHR FLAG
                     FE0C'   31    01F1          BRW     MAC$ERRORPT                       ;ISSUE ERROR AND RETURN
                                   01F4  ;
                                   01F4  ; HERE WHEN DELIMITER SEEN
                                   01F4  ;
                6B     01    CA    01F4  50$:    BICL2   #FLG$M_ALLCHR,(R11)               ;CLEAR ALLCHR FLAG
                     FE06'   30    01F7          BSBW    MAC$GETCHR                        ;SKIP OVER DELIMITER
                     FE03'   30    01FA          BSBW    MAC$SKIPSP                        ;SKIP SPACES
                0D     5A    91    01FD          CMPB    R10,#CR                           ;STOP ON EOL?
                       C0    12    0200          BNEQ    5$                                ;IF NEQ NO--LOSE
0000'CF     58    00000008'8F    C3    0202          SUBL3   #MAC$AB_TMPBUF+8,R8, -            ; Form descriptor to library
                                   020C                  W^MAC$AB_TMPBUF                   ; file name
        0004'CF       0008'CF    9E    020C          MOVAB   W^MAC$AB_TMPBUF+8, -
                                   0213                  W^MAC$AB_TMPBUF+4
                     00C5    30    0213          BSBW    MAC$ALL_MLB_MLF                   ; Get memory block for MLF
                                   0216  ;
                                   0216  ; Call librarian procedure to initialise library
                                   0216  ;
                     18 A6   9F    0216          PUSHAB  MLF$X_NAMBLK(R6)                  ; Address of NAM block
                   0000'CF   9F    0219          PUSHAB  W^MAC$GL_LIBTYPE                  ; Address of type = MLB
```

```
            0000'CF   9F  021D   350          PUSHAB  W^MAC$GL_LIBFUNC         ; Address of function = READ
              14 A6   9F  0221   351          PUSHAB  MLF$L_CTINDEX(R6)        ; Address of control table index
      00000000'GF 04  FB  0224   352          CALLS   #4,G^LBR$INI_CONTROL     ; Initialise library
              22 50   E9  022B   353          BLBC    R0,55$                   ; Error if LBC
                          022E   354  ;
                          022E   355  ; Call librarian procedure to open library file
                          022E   356  ;
              0C A6   9F  022E   357          PUSHAB  MLF$Q_FNAMDS(R6)         ; Address of resultant length
              0C A6   9F  0231   358          PUSHAB  MLF$Q_FNAMDS(R6)         ; Address of resultant descriptor
                 00   DD  0234   359          PUSHL   #0                       ; No related file name
            0000'CF   9F  0236   360          PUSHAB  W^MAC$MLB_DEFNAM         ; Address of default name
                 00   DD  023A   361          PUSHL   #0                       ; No create options
            0000'CF   9F  023C   362          PUSHAB  W^MAC$AB_TMPBUF          ; Address of file name
              14 A6   9F  0240   363          PUSHAB  MLF$L_CTINDEX(R6)        ; Address of control table index
      00000000'GF 07  FB  0243   364          CALLS   #7,G^LBR$OPEN            ; Open library file
              58 8ED0     024A   365          POPL    R8                       ;RESTORE R8
              14 50   E8  024D   366          BLBS    R0,60$                   ;BRANCH IF GOOD OPEN
                          0250   367  55$:
                 56   DD  0250   368          PUSHL   R6                       ; Address of MLF
      00000000'EF 01  FB  0252   369          CALLS   #1,MAC$ERR_LBROPEN       ; Report library open error
                          0259   370          $MAC_ERR MLBOPNERR               ;REPORT ERROR WITHIN LISTING
              FD9F'   30  025E   371          BSBW    MAC$ERRORPT              ;AND EXIT WITH ERROR STATUS
              FD9C'   31  0261   372          BRW     MAC$ABORT_PASS1          ;GO ABORT PASS 1
       0000'CF 66     0E  0264   373  60$:    INSQUE  (R6),W^MAC$GL_MLB_QUE    ;INSERT IN MLB QUEUE
            0000'CF   D6  0269   374          INCL    W^MAC$GL_MLB_CNT         ; Increment macro library count
                      05  026D   375          RSB
```

```
                          026E          .SBTTL  MAC$SYSLIB_SET  SET UP THE SYSTEM LIBRARY
                          026E
                          026E  ;++
                          026E  ; FUNCTIONAL DESCRIPTION:
                          026E  ;
                          026E  ;       THIS ROUTINE IS CALLED AT MACRO-32 INITIALIZATION TO ENSURE
                          026E  ;       THAT THE SYSTEM LIBRARY EXISTS, AND CREATES AN IN-CORE INDEX
                          026E  ;       OF IT.  THE MACRO LIBRARY QUEUE IS ALSO INITIALIZED.
                          026E  ;
                          026E  ;--
                          026E
                          026E  MAC$SYSLIB_SET::
       50  0000'CF   9E   026E          MOVAB   W^MAC$GL_MLB_QUE,R0     ;INIT THE MLB QUEUE
       60     60     DE   0273          MOVAL   (R0),(R0)               ;...
       60     80     DE   0276          MOVAL   (R0)+,(R0)              ;...
       56  0000'CF   9E   0279          MOVAB   W^MAC$SYSLIB_MLF,R6     ; Point to SYSLIB MLF entry
           14 A6     D4   027E          CLRL    MLF$L_CTINDEX(R6)       ; Clear control table index
    0C A6  00FF 8F   B0   0281          MOVW    #MLF$K_RSFNLN,MLF$Q_FNAMDS(R6) ; Initialise file
    10 A6     78 A6  DE   0287          MOVAL   MLF$T_FNAM(R6),MLF$Q_FNAMDS+4(R6) ; name descriptor
                          028C  ;
                          028C  ; Call librarian procedure to initialise library
                          028C  ;
           18 A6     9F   028C          PUSHAB  MLF$X_NAMBLK(R6)        ; Address of NAM block
         0000'CF     9F   028F          PUSHAB  W^MAC$GL_LIBTYPE        ; Address of type = MLB
         0000'CF     9F   0293          PUSHAB  W^MAC$GL_LIBFUNC        ; Address of function = READ
           14 A6     9F   0297          PUSHAB  MLF$L_CTINDEX(R6)       ; Address of control table index
 00000000'GF  04     FB   029A          CALLS   #4,G^LBR$INI_CONTROL    ; Initialise library
           2B 50     E9   02A1          BLBC    R0,10$                  ; Error if LBC
                          02A4  ;
                          02A4  ; Call librarian procedure to open library file
                          02A4  ;
           0C A6     9F   02A4          PUSHAB  MLF$Q_FNAMDS(R6)        ; Address of resultant length
           0C A6     9F   02A7          PUSHAB  MLF$Q_FNAMDS(R6)        ; Address of resultant descriptor
              00     DD   02AA          PUSHL   #0                      ; No related file name
     00000000'GF     9F   02AC          PUSHAB  G^MAC$SYSLIB_DFN        ; Address of default name descriptor
              00     DD   02B2          PUSHL   #0                      ; No create options
         0000'CF     9F   02B4          PUSHAB  W^MAC$SYSLIB_FNM        ; Address of file name descriptor
           14 A6     9F   02B8          PUSHAB  MLF$L_CTINDEX(R6)       ; Address of control table index
 00000000'GF  07     FB   02BB          CALLS   #7,G^LBR$OPEN           ; Open library file
           0A 50     E9   02C2          BLBC    R0,10$                  ;IF LBC THEN ERROR
 0000'CF      66     0E   02C5          INSQUE  (R6),W^MAC$GL_MLB_QUE   ;INSERT IN MLB QUEUE
         0000'CF     D6   02CA          INCL    W^MAC$GL_MLB_CNT        ; Increment macro library count
              05          02CE          RSB
                          02CF  10$:
              56     DD   02CF          PUSHL   R6                      ; Address of MLF
 00000000'EF  01     FB   02D1          CALLS   #1,MAC$ERR_LBROPEN      ; Report library open error
          FD25'      31   02D8          BRW     MAC$LAST_CHANCE         ; and go die
```

```
                                         02DB          .SBTTL  ALLOCATE MLF BLOCK
                                         02DB    ;
                                         02DB    ; Functional description:
                                         02DB    ;
                                         02DB    ;       This routine is called to allocate a blcok of memory for
                                         02DB    ;       a MLF (Macro Library File) entry.  The allocated memory block
                                         02DB    ;       is zeroed and the NAM block and file name descriptors initialised.
                                         02DB    ;
                                         02DB    ; Inputs:
                                         02DB    ;
                                         02DB    ;       None
                                         02DB    ;
                                         02DB    ; Outputs:
                                         02DB    ;
                                         02DB    ;       R6 = Address of MLF
                                         02DB    ;
                                         02DB    ;--
                                         02DB
                                         02DB    MAC$ALL_MLB_MLF::
   00000000'GF    00000000'EF        FA  02DB          CALLG   MLF_ARGLIST,G^LIB$GET_VM ; Get memory block
                        1F 50        E9  02E6          BLBC    R0,10$                   ; Error if LBC
                  56    0000'CF      D0  02E9          MOVL    W^MAC$GL_BASEADDR,R6     ; Get base address of memory block
 66    0177 8F    00    6E    00     2C  02EE          MOVC5   #0,(SP),#0,#MLF$K_BLKSIZ,(R6) ; Clear MLF
             0C A6     00FF 8F       B0  02F6          MOVW    #MLF$K_RSFNLN,MLF$Q_FNAMDS(R6) ; Initialise file name
          10 A6          78 A6       DE  02FC          MOVAL   MLF$T_FNAM(R6),MLF$Q_FNAMDS+4(R6) ; descriptor
                       6002 8F       B0  0301          MOVW    #<<NAM$C_BLN>@8+NAM$C_BID>,- ; Identify NAM block
                          18 A6          0305                  MLF$X_NAMBLK(R6)
                                     05  0307          RSB
                                         0308    10$:
                0000'CF    00        FB  0308          CALLS   #0,W^MAC$ERR_NOMEM_0     ; Report the error
                       FCF0'         31  030D          BRW     MAC$LAST_CHANCE          ; and go die
                                         0310
                                         0310          .END
```

```
$$T1                            = 00000002         DUPX                            = 0000002A
$COUNT                          = 0000003B         DWUP                            = 00000030
ARG$K_SIZE                      = 000003E8         DXOR                            = 0000001F
AUD$K_SIZE                      = 00000010         ERR                             = 00000000
BLNK                            = 00000020         ERR01                           = 00000001
CHR$M_COMMA_CR                  = 00000020         ERR02                           = 00000002
CHR$M_ILL_CHR                   = 00000040         ERR03                           = 00000003
CHR$M_NUM_BER                   = 00000010         ERR04                           = 00000004
CHR$M_SPA_MSK                   = 00000001         ERR05                           = 00000005
CHR$M_SYM_CH1                   = 00000008         ERR06                           = 00000006
CHR$M_SYM_CHR                   = 00000004         ERR07                           = 00000007
CHR$M_SYM_DLM                   = 00000002         ERR08                           = 00000008
CHR$V_COMMA_CR                  = 00000005         ERR09                           = 00000009
CHR$V_CVTLWC                    = 00000061         FF                              = 0000000C
CHR$V_ILL_CHR                   = 00000006         FLG$M_ALLCHR                    = 00000001
CHR$V_NOCVT                     = 0000007F         FLG$M_BOL                       = 00000002
CHR$V_NUM_BER                   = 00000004         FLG$M_CHKLPND                   = 00100000
CHR$V_SPA_MSK                   = 00000000         FLG$M_COMPEXPR                  = 00000004
CHR$V_SYM_CH1                   = 00000003         FLG$M_CONT                      = 00000008
CHR$V_SYM_CHR                   = 00000002         FLG$M_CRF                       = 40000000
CHR$V_SYM_DLM                   = 00000001         FLG$M_CRSEEN                    = 00000001
CNT                             = 00000002         FLG$M_DATRPT                    = 00000010
CR                              = 0000000D         FLG$M_DBGOUT                    = 00004000
DAND                            = 0000001D         FLG$M_DLIMSTR                   = 00008000
DANGCLS                         = 00000016         FLG$M_ENDMCH                    = 00000020
DANGOPN                         = 00000015         FLG$M_EVALEXPR                  = 00000040
DAT                             = 00000020         FLG$M_EXPOPT                    = 00000080
DBUP                            = 0000002B         FLG$M_EXTERR                    = 00010000
DCLS                            = 00000018         FLG$M_EXTWRN                    = 00020000
DCOLON                          = 00000010         FLG$M_FIRSTLN                   = 00000200
DCOMMA                          = 0000000F         FLG$M_IFSTAT                    = 00800000
DDIV                            = 0000001C         FLG$M_IIF                       = 00400000
DEOL                            = 0000000B         FLG$M_INSERT                    = 00000100
DEQ                             = 00000011         FLG$M_IRPC                      = 20000000
DGUP                            = 0000002C         FLG$M_LEXOP                     = 00000002
DINTEGER                        = 00000022         FLG$M_LSTXST                    = 00000200
DIUP                            = 0000002D         FLG$M_MAC2COL                   = 00000800
DLUP                            = 0000002E         FLG$M_MACL                      = 00000800
DMASK                           = 00000032         FLG$M_MACLTB                    = 08000000
DMINUS                          = 0000001A         FLG$M_MACTXT                    = 00010000
DOPCODE                         = 0000000E         FLG$M_MEBLST                    = 00001000
DOPN                            = 00000017         FLG$M_MOREARG                   = 00002000
DOR                             = 0000001E         FLG$M_MOREINP                   = 00000008
DPC                             = 00000012         FLG$M_NEWPND                    = 00000400
DPLUS                           = 00000019         FLG$M_NOREF                     = 01000000
DPOUND                          = 00000021         FLG$M_NTYPEPC                   = 00000020
DSQCLS                          = 00000014         FLG$M_NULCHR                    = 00040000
DSQOPN                          = 00000013         FLG$M_OBJXST                    = 00200000
DSUP                            = 0000002F         FLG$M_OPNDCHK                   = 00000100
DTIMES                          = 0000001B         FLG$M_OPRND                     = 00002000
DUPA                            = 00000023         FLG$M_OPTVFLIDX                 = 00001000
DUPB                            = 00000024         FLG$M_ORDLST                    = 00020000
DUPC                            = 00000025         FLG$M_P2                        = 00004000
DUPD                            = 00000026         FLG$M_RPTIRP                    = 10000000
DUPF                            = 00000028         FLG$M_SEQFIL                    = 02000000
DUPM                            = 00000029         FLG$M_SKAN                      = 00008000
DUPO                            = 00000027         FLG$M_SPECOP                    = 00000004
```

```
FLG$M_SPLALL                    = 04000000          FLG$V_UPMARG                    = 00000026
FLG$M_STOIMF                    = 00040000          FLG$V_XCRF                      = 0000001F
FLG$M_SYM2COL                   = 00000400          GOALSY                          = 0000000A
FLG$M_TOCFLG                    = 00080000          HASHSZ                          = 0000007F
FLG$M_UPAFLG                    = 00000010          HYPHEN                          = 0000002D
FLG$M_UPDFIL                    = 00000080          ID                              = 0000000C
FLG$M_UPMARG                    = 00000040          INP$K_BUFSIZ                    = 000003E8
FLG$M_XCRF                      = 80000000          INT$K_BUFSIZ                    = 000013F4
FLG$V_ALLCHR                    = 00000000          INT$K_BUFWRN                    = 00001390
FLG$V_BOL                       = 00000001          INT$_ADD                        = 00000001
FLG$V_CHKLPND                   = 00000014          INT$_AND                        = 00000002
FLG$V_COMPEXPR                  = 00000002          INT$_ASH                        = 00000003
FLG$V_CONT                      = 00000003          INT$_ASN                        = 0000000C
FLG$V_CRF                       = 0000001E          INT$_AUGPC                      = 0000000D
FLG$V_CRSEEN                    = 00000020          INT$_BDST                       = 0000000E
FLG$V_DATRPT                    = 00000004          INT$_CHKL                       = 0000000F
FLG$V_DBGOUT                    = 0000002E          INT$_DIV                        = 00000004
FLG$V_DLIMSTR                   = 0000002F          INT$_END                        = 00000010
FLG$V_ENDMCH                    = 00000005          INT$_EPT                        = 00000011
FLG$V_EVALEXPR                  = 00000006          INT$_ERR                        = 00000012
FLG$V_EXPOPT                    = 00000007          INT$_ETX                        = 00000013
FLG$V_EXTERR                    = 00000030          INT$_FNEWL                      = 00000014
FLG$V_EXTWRN                    = 00000031          INT$_ILG                        = 00000000
FLG$V_FIRSTLN                   = 00000029          INT$_INFO                       = 0000003A
FLG$V_IFSTAT                    = 00000017          INT$_LGLAB                      = 00000015
FLG$V_IIF                       = 00000016          INT$_MACL                       = 00000016
FLG$V_INSERT                    = 00000008          INT$_MUL                        = 00000005
FLG$V_IRPC                      = 0000001D          INT$_NEG                        = 00000006
FLG$V_LEXOP                     = 00000021          INT$_NEWL                       = 00000017
FLG$V_LSTXST                    = 00000009          INT$_NEWP                       = 00000018
FLG$V_MAC2COL                   = 0000002B          INT$_NOT                        = 00000007
FLG$V_MACL                      = 0000000B          INT$_OP                         = 00000019
FLG$V_MACLTB                    = 0000001B          INT$_OR                         = 00000008
FLG$V_MACTXT                    = 00000010          INT$_PRIL                       = 0000001A
FLG$V_MEBLST                    = 0000000C          INT$_PRT                        = 0000001B
FLG$V_MOREARG                   = 0000002D          INT$_PSECT                      = 0000001C
FLG$V_MOREINP                   = 00000023          INT$_REDEF                      = 0000001D
FLG$V_NEWPND                    = 0000000A          INT$_REF                        = 0000001E
FLG$V_NOREF                     = 00000018          INT$_REST                       = 0000001F
FLG$V_NTYPEPC                   = 00000025          INT$_SAME                       = 00000009
FLG$V_NULCHR                    = 00000032          INT$_SAVE                       = 00000020
FLG$V_OBJXST                    = 00000015          INT$_SBTTL                      = 00000021
FLG$V_OPNDCHK                   = 00000028          INT$_SETFLAG                    = 00000022
FLG$V_OPRND                     = 0000000D          INT$_SETLONG                    = 00000023
FLG$V_OPTVFLIDX                 = 0000002C          INT$_SPIC                       = 00000024
FLG$V_ORDLST                    = 00000011          INT$_SPID                       = 00000025
FLG$V_P2                        = 0000000E          INT$_STIB                       = 00000026
FLG$V_RPTIRP                    = 0000001C          INT$_STIL                       = 00000028
FLG$V_SEQFIL                    = 00000019          INT$_STIW                       = 00000027
FLG$V_SKAN                      = 0000000F          INT$_STKEPT                     = 00000029
FLG$V_SPECOP                    = 00000022          INT$_STKG                       = 0000002A
FLG$V_SPLALL                    = 0000001A          INT$_STKL                       = 0000002B
FLG$V_STOIMF                    = 00000012          INT$_STKPC                      = 0000002C
FLG$V_SYM2COL                   = 0000002A          INT$_STKS                       = 0000002D
FLG$V_TOCFLG                    = 00000013          INT$_STOB                       = 00000034
FLG$V_UPAFLG                    = 00000024          INT$_STOL                       = 0000002E
FLG$V_UPDFIL                    = 00000027          INT$_STOW                       = 00000035
```

```
INT$_STRB                = 0000002F          KLINK                    = 00000085
INT$_STRL                = 00000031          KLIST                    = 00000061
INT$_STRSB               = 00000032          KLONG                    = 0000003C
INT$_STRSW               = 00000033          KMACRO                   = 00000050
INT$_STRW                = 00000030          KMCALL                   = 00000051
INT$_STSB                = 00000036          KMDELETE                 = 00000054
INT$_STSW                = 00000037          KMEXIT                   = 00000052
INT$_SUB                 = 0000000A          KNARG                    = 00000063
INT$_SUME                = 00000039          KNCHR                    = 00000064
INT$_WRN                 = 00000038          KNCROS                   = 0000007A
INT$_XOR                 = 0000000B          KNLIST                   = 00000062
KADDRESS                 = 00000037          KNTYPE                   = 00000074
KALIGN                   = 0000005A          KOCTA                    = 00000083
KASCIC                   = 00000033          KODD                     = 0000005C
KASCID                   = 00000078          KOPDEF                   = 00000075
KASCII                   = 00000034          KPACKED                  = 00000036
KASCIZ                   = 00000035          KPAGE                    = 00000065
KBLKA                    = 0000003F          KPRINT                   = 00000072
KBLKB                    = 00000040          KPSECT                   = 00000066
KBLKD                    = 00000041          KQUAD                    = 0000003D
KBLKF                    = 00000042          KREF1                    = 0000006D
KBLKG                    = 0000007E          KREF16                   = 00000084
KBLKH                    = 0000007F          KREF2                    = 0000006E
KBLKL                    = 00000043          KREF4                    = 0000006F
KBLKO                    = 00000080          KREF8                    = 00000070
KBLKQ                    = 00000044          KREPT                    = 0000004D
KBLKW                    = 00000045          KRESTORE                 = 00000067
KBYTE                    = 00000038          KSAVE                    = 00000068
KCROSS                   = 00000079          KSBTTL                   = 0000006B
KDEBUG                   = 00000055          KSGNB                    = 0000007C
KDFLT                    = 0000007B          KSGNW                    = 0000007D
KDOUBLE                  = 00000039          KTITLE                   = 00000069
KDSABL                   = 00000056          KVECTOR                  = 00000059
KENABL                   = 00000057          KWARN                    = 00000073
KEND                     = 00000076          KWEAK                    = 0000006C
KENDC                    = 0000004E          KWORD                    = 0000003E
KENDM                    = 00000053          KXFER                    = 00000077
KENDR                    = 0000004F          LBR$GET_RECORD             ********     X     05
KENTRY                   = 00000058          LBR$INI_CONTROL            ********     X     05
KERROR                   = 00000071          LBR$LOOKUP_KEY             ********     X     05
KEVEN                    = 0000005B          LBR$OPEN                   ********     X     05
KEXTRN                   = 0000005D          LIB$GET_VM                 ********     X     05
KFIELD                   = 0000003A          LIBRY                      000001B8 RG        05
KFLOAT                   = 0000003B          LST$G_MACRODEF             ********     X     05
KGFLOAT                  = 00000081          LST$K_BUFSIZ             = 00000086
KGLOBL                   = 0000005E          LST$K_L_P_PAGE           = 0000003C
KHFLOAT                  = 00000082          LST$K_TITLE_SIZ          = 00000028
KIDENT                   = 0000006A          MAC$ABORT_PASS1            ********     X     05
KIF                      = 00000046          MAC$AB_LINEBF              ********     X     05
KIFF                     = 00000048          MAC$AB_TMPBUF              ********     X     05
KIFT                     = 00000049          MAC$AB_TMPSYM              ********     X     04
KIFTF                    = 0000004A          MAC$ALC_1_PAGE             ********     X     05
KIIF                     = 00000047          MAC$ALL_MCB_MLF            000002DB RG        05
KINCLUDE                 = 0000005F          MAC$AL_PRMHSHTB            ********     X     05
KIRP                     = 0000004B          MAC$AL_UMCHSHTB            ********     X     05
KIRPC                    = 0000004C          MAC$DEA_1_PAGE             ********     X     05
KLIBRARY                 = 00000060          MAC$ERRORPT                ********     X     05
```

```
MAC$ERR_LBROPEN                 ********    X   05     OBJ$K_BUFSIZ                  = 00000200
MAC$ERR_NOMEM_O                 ********    X   05     OPF$M_LASTOPR                 = 00002000
MAC$GETCHR                      ********    X   05     OPF$M_OPTEXP                  = 00001000
MAC$GET_MLB_LIN                 0000017F RG     05     OPF$V_LASTOPR                 = 0000000D
MAC$GL_BASEADDR                 ********    X   03     OPF$V_OPTEXP                  = 0000000C
MAC$GL_ERRPT                    ********    X   05     PSC$B_NAME                      00000004
MAC$GL_ERRPTX                   ********    X   05     PSC$B_SEG                       0000000C
MAC$GL_INPUTP                   ********    X   05     PSC$B_UNUSED                    0000000B
MAC$GL_LIBFUNC                  ********    X   05     PSC$K_BLKSIZ                    00000013
MAC$GL_LIBTYPE                  ********    X   05     PSC$K_NO_OPTNS                = 0000000A
MAC$GL_LINELN                   ********    X   05     PSC$L_CURLOC                    0000000F
MAC$GL_LINENUM                  ********    X   05     PSC$L_LINK                      00000000
MAC$GL_LINEPT                   ********    X   05     PSC$L_MAXLGTH                   00000005
MAC$GL_MACPTR                   ********    X   05     PSC$M_ABS                     = FFFFFFF7
MAC$GL_MLB_CNT                  ********    X   05     PSC$M_ALIGNFLG                = 00004000
MAC$GL_MLB_GET                  ********    X   05     PSC$M_ALLOPTNS                = 000003FF
MAC$GL_MLB_MDF                  ********    X   05     PSC$M_BYTE                    = 00004000
MAC$GL_MLB_QUE                  ********    X   05     PSC$M_CON                     = FFFFFFFB
MAC$GL_MLFPTR                   ********    X   05     PSC$M_DEFAULT                 = 000001C8
MAC$GL_TXTRFA                   ********    X   05     PSC$M_EXE                     = 000000C0
MAC$GQ_LINEBFDS                 ********    X   05     PSC$M_GBL                     = 00000010
MAC$IMPLMCALL                   00000047 RG     05     PSC$M_LCL                     = FFFFFFEF
MAC$INTOUT_2_LW                 ********    X   05     PSC$M_LIB                     = 00000002
MAC$LAST_CHANCE                 ********    X   05     PSC$M_LONG                    = 00004800
MAC$MLB_DEFNAM                  ********    X   05     PSC$M_NOEXE                   = FFFFFFBF
MAC$SKIPSP                      ********    X   05     PSC$M_NOPIC                   = FFFFFFFE
MAC$SRCSYMTAB                   ********    X   05     PSC$M_NORD                    = FFFFFF7F
MAC$SYMSCNUP                    ********    X   05     PSC$M_NOSHR                   = FFFFFFDF
MAC$SYSLIB_DFN                  ********    X   05     PSC$M_NOVEC                   = FFFFFDFF
MAC$SYSLIB_FNM                  ********    X   05     PSC$M_NOWRT                   = FFFFFEFF
MAC$SYSLIB_MLF                  ********    X   05     PSC$M_OVR                     = 00000004
MAC$SYSLIB_SET                  0000026E RG     05     PSC$M_PAGE                    = 00006400
MAC$TMPSYMDS                    00000000 R      04     PSC$M_PIC                     = 00000001
MAC$_CANTLOCMAC               = 007D905A               PSC$M_QUAD                    = 00004C00
MAC$_DIRSYNX                  = 007D906A               PSC$M_RD                      = 00000080
MAC$_MACLBFMTER               = 007D913A               PSC$M_REL                     = 00000008
MAC$_MLBOPNERR                = 007D923A               PSC$M_SHR                     = 00000020
MAC$_UNTERMARG                = 007D922A               PSC$M_USR                     = FFFFFFFD
MACRO                           ********    X   05     PSC$M_VEC                     = 00000200
MACTXT                        = 0000000D               PSC$M_WORD                    = 00004400
MAC_LIB_FMT_ERR                 00000174 R      05     PSC$M_WRT                     = 00000180
MAC_SUBSYS                    = 0000007D               PSC$S_ALIGNMENT               = 00000004
MCALL                           00000000 RG     05     PSC$V_ALIGNFLG                = 0000000E
MLF$K_BLKSIZ                    00000177               PSC$V_ALIGNMENT               = 0000000A
MLF$K_RSFNLN                  = 000000FF               PSC$V_EXE                     = 00000006
MLF$L_CTINDEX                   00000014               PSC$V_GBL                     = 00000004
MLF$L_MCDEF                     00000008               PSC$V_LIB                     = 00000001
MLF$L_QLINK                     00000000               PSC$V_OVR                     = 00000002
MLF$Q_FNAMDS                    0000000C               PSC$V_PIC                     = 00000000
MLF$T_FNAM                      00000078               PSC$V_RD                      = 00000007
MLF$X_NAMBLK                    00000018               PSC$V_REL                     = 00000003
MLF_ARGLIST                     0G000000 R      03     PSC$V_SHR                     = 00000005
MLF_SIZE                        0000000C R      03     PSC$V_VEC                     = 00000009
NAM$C_BID                     = 00000002               PSC$V_WRT                     = 00000008
NAM$C_BLN                     = 00000060               PSC$W_FLAG                      00000009
NAM$C_MAXRSS                  = 000000FF               PSC$W_OPTIONS                   0000000D
NEXT_MLB                        00000058 R      05     RDX$V_BINARY                  = 00000000
```

```
RDX$V_DECIMAL                    = 00000002
RDX$V_DOUBLE                     = 00000005
RDX$V_FLOAT                      = 00000004
RDX$V_GFLOAT                     = 00000006
RDX$V_HEX                        = 00000003
RDX$V_HFLOAT                     = 00000007
RDX$V_OCTAL                      = 00000001
REG$_PC                          = 0000000F
RRREG                            = 00000031
SEMI                             = 0000003B
STB$K_PG_MISS                    = 0000000A
SYM$B_NAME                         00000004
SYM$B_SEG                          0000000C
SYM$B_TOKEN                        0000000B
SYM$K_BLKSIZ                       0000000D
SYM$K_MAXLEN                     = 0000001F
SYM$K_TWOCOL                     = 00000010
SYM$L_LINK                         00000000
SYM$L_VAL                          00000005
SYM$M_ABS                        = 00000010
SYM$M_ASN                        = 00000100
SYM$M_CRFO                       = 00002000
SYM$M_DEBUG                      = 00000020
SYM$M_DEF                        = 00000001
SYM$M_DELMAC                     = 00000200
SYM$M_EPT                        = 00000200
SYM$M_EXTRN                      = 00000008
SYM$M_GLOBL                      = 00000004
SYM$M_LOCAL                      = 00000040
SYM$M_ODBG                       = 00000400
SYM$M_REF                        = 00000080
SYM$M_RELPSECT                   = 00000800
SYM$M_SUPR                       = 00004000
SYM$M_WEAK                       = 00000002
SYM$M_XCRF                       = 00001000
SYM$V_ABS                        = 00000004
SYM$V_ASN                        = 00000008
SYM$V_CRFO                       = 0000000D
SYM$V_DEBUG                      = 00000005
SYM$V_DEF                        = 00000000
SYM$V_DELMAC                     = 00000009
SYM$V_EPT                        = 00000009
SYM$V_EXTRN                      = 00000003
SYM$V_GLOBL                      = 00000002
SYM$V_LOCAL                      = 00000006
SYM$V_ODBG                       = 0000000A
SYM$V_REF                        = 00000007
SYM$V_RELPSECT                   = 0000000B
SYM$V_SUPR                       = 0000000E
SYM$V_WEAK                       = 00000001
SYM$V_XCRF                       = 0000000C
SYM$W_FLAG                         00000009
TAB                              = 00000009
X1                               = 00000400
X2                               = 0000000F
```

```
                                        +----------------+
                                        ! Psect synopsis !
                                        +----------------+

PSECT name                 Allocation          PSECT No.  Attributes
----------                 ----------          ---------  ----------
.  ABS  .                  00000000  (    0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD   NOWRT NOVEC BYTE
. BLANK .                  00000000  (    0.)  01 (  1.)  NOPIC   USR   CON   REL   LCL NOSHR   EXE   RD     WRT NOVEC BYTE
$ABS$                      00000177  (  375.)  02 (  2.)  NOPIC   USR   CON   ABS   LCL NOSHR   EXE   RD     WRT NOVEC BYTE
MAC$RO_DATA                00000010  (   16.)  03 (  3.)  NOPIC   USR   CON   REL   GBL NOSHR NOEXE   RD   NOWRT NOVEC LONG
MAC$RW_DATA                00000008  (    8.)  04 (  4.)  NOPIC   USR   CON   REL   LCL NOSHR NOEXE   RD     WRT NOVEC LONG
MAC$RO_CODE_MAC            00000310  (  784.)  05 (  5.)  NOPIC   USR   CON   REL   GBL NOSHR   EXE   RD   NOWRT NOVEC LONG

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

Phase                  Page faults   CPU Time       Elapsed Time
-----                  -----------   --------       ------------
Initialization                  30   00:00:00.04    00:00:01.80
Command processing             103   00:00:00.39    00:00:02.31
Pass 1                         287   00:00:06.03    00:00:29.58
Symbol table sort                5   00:00:00.98    00:00:02.90
Pass 2                          99   00:00:01.29    00:00:05.01
Symbol table output             60   00:00:00.26    00:00:01.51
Psect synopsis output            2   00:00:00.02    00:00:00.03
Cross-reference output           0   00:00:00.00    00:00:00.00
Assembler run totals           588   00:00:09.01    00:00:43.14

The working set limit was 1350 pages.
55114 bytes (108 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1042 non-local and 20 local symbols.
457 source lines were read in Pass 1, producing 24 object records in Pass 2.
17 pages of virtual memory were used to define 16 macros.

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

Macro library name                      Macros defined
------------------                      --------------
_$255$DUA28:[MACRO.OBJ]MACRO.MLB;1                  11
_$255$DUA28:[SYSLIB]STARLET.MLB;2                    8
TOTALS (all libraries)                              19

1142 GETS were required to define 19 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:MACLIB/OBJ=OBJ$:MACLIB MSRC$:MACLIB/UPDATE=(ENH$:MACLIB)+LIB$:MACRO/LIB
```

0226 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY