
MMM MMM AAAAAAAAA CCCCCCCCCCC RRRRRRRRRRR 000000000
MMM MMM AAAAAAAAA CCCCCCCCCCC RRRRRRRRRRR 000000000
MMM MMM AAAAAAAAA CCCCCCCCCCC RRRRRRRRRRR 000000000
MMMMMM MMMMMM AAA AAA CCC RRR RRR OOO OOO
MMMMMM MMMMMM AAA AAA CCC RRR RRR OOO OOO
MMMMMM MMMMMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM AAA AAA CCC RRRRRRRRRRR OOO OOO
MMM MMM AAA AAA CCC RRRRRRRRRRR OOO OOO
MMM MMM AAA AAA CCC RRRRRRRRRRR OOO OOO
MMM MMM AAAAAAAAAAAAAAA CCC RRR RRR OOO OOO
MMM MMM AAAAAAAAAAAAAAA CCC RRR RRR OOO OOO
MMM MMM AAAAAAAAAAAAAAA CCC RRR RRR OOO OOO
MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM AAA AAA CCC RRR RRR OOO OOO
MMM MMM AAA AAA CCCCCCCCCCC RRR RRR 000000000
MMM MMM AAA AAA CCCCCCCCCCC RRR RRR 000000000
MMM MMM AAA AAA CCCCCCCCCCC RRR RRR 000000000
MMM MMM AAA AAA CCCCCCCCCCC RRR RRR 000000000

FILEID**OHDOUT

```
  OOOOOO    HH      HH  DDDDDDDD     OOOOOO    UU      UU  TTTTTTTTTT
  OOOOOO    HH      HH  DDDDDDDD     OOOOOO    UU      UU  TTTTTTTTTT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HHHHHHHHHH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HHHHHHHHHH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
OO      OO  HH      HH  DD      DD  OO      OO  UU      UU      TT
  OOOOOO    HH      HH  DDDDDDDD     OOOOOO    UUUUUUUUUU      TT
  OOOOOO    HH      HH  DDDDDDDD     OOOOOO    UUUUUUUUUU      TT

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
```

Table of contents

```
0000     1          .TITLE  MAC$OHDOUT      OUTPUT OBJECT MODULE HEADER INFO
0000     2          .IDENT  'V04-000'
0000     3
0000     4  ;
0000     5  ;**********************************************************************************
0000     6  ;*                                                                                *
0000     7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                       *
0000     8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                        *
0000     9  ;*  ALL RIGHTS RESERVED.                                                          *
0000    10  ;*                                                                                *
0000    11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED          *
0000    12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE        *
0000    13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER        *
0000    14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY        *
0000    15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY        *
0000    16  ;*  TRANSFERRED.                                                                  *
0000    17  ;*                                                                                *
0000    18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE         *
0000    19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT        *
0000    20  ;*  CORPORATION.                                                                  *
0000    21  ;*                                                                                *
0000    22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS         *
0000    23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                       *
0000    24  ;*                                                                                *
0000    25  ;*                                                                                *
0000    26  ;**********************************************************************************
0000    27  ;
0000    28
0000    29  ;++
0000    30  ; FACILITY:     VAX MACRO ASSEMBLER OBJECT LIBRARY
0000    31  ;
0000    32  ; ABSTRACT:
0000    33  ;
0000    34  ; The VAX-11 MACRO assembler translates MACRO-32 source code into object
0000    35  ; modules for input to the VAX-11 LINKER.
0000    36  ;
0000    37  ; ENVIRONMENT:  USER MODE
0000    38  ;
0000    39  ; AUTHOR: Benn Schreiber, CREATION DATE: 20-AUG-78
0000    40  ;
0000    41  ; MODIFIED BY:
0000    42  ;
0000    43  ;       V05-002 MTR0033         Mike Rhodes     22-Apr-1983
0000    44  ;               Allow removal of the blank psect if its not referenced.
0000    45  ;
0000    46  ;       V03-001 MTR0031         Mike Rhodes     12-Apr-1983
0000    47  ;               Add code following WRT_GSD to support the .LINK directive.
0000    48  ;
0000    49  ;       V02.03  PCG0002         Peter George    16-Apr-1981
0000    50  ;               Set DBGOUT flag to prevent filtering of debugger
0000    51  ;               records for abs psects.  Also, filter out symbols
0000    52  ;               in abs psects that are not referenced by rel psects.
0000    53  ;
0000    54  ;       V02.02  CNH0039         Chris Hume       7-Oct-1980
0000    55  ;               Closed off a few remaining object record length checking
0000    56  ;               errors.  (IODAT.MAR 02.08, P2ACT1.MAR 02.09)
0000    57  ;--
```

```
                     59          .SBTTL  DECLARATIONS
                     60 ;
                     61 ; INCLUDE FILES:
                     62 ;
                     63
                     64 ;
                     65 ; MACROS:
                     66 ;
                     67          $MAC_OBJCODDEF                  ;DEFINE OBJECT CODE RECORD TYPES
                     68          $MAC_SYMBLKDEF                  ;DEFINE SYMBOL BLOCK OFFSETS
                     69          $MAC_CTLFLGDEF                  ;DEFINE CONTROL FLAGS
                     70
                     71 ;
                     72 ; EQUATED SYMBOLS:
                     73 ;
                     74
00000000             75          .PSECT  MAC$RO_CODE_P2,NOWRT,GBL,LONG
```

```
                                0000     77          .SBTTL  MAC$OBJHDROUT OUTPUT OBJECT MODULE HEADER RECORDS
                                0000     78
                                0000     79 ;++
                                0000     80 ; FUNCTIONAL DESCRIPTION:
                                0000     81 ;
                                0000     82 ;       THIS ROUTINE OUTPUTS THE OBJECT MODULE HEADER RECORDS AT THE
                                0000     83 ;       BEGINNING OF PASS 2.  THESE INCLUDE THE GSD RECORDS.
                                0000     84 ;
                                0000     85 ;--
                                0000     86
                                0000     87 MAC$OBJHDROUT::
          03 6B    2E    E2     0000     88          BBSS    #FLG$V_DBGOUT,(R11),.+4 ;SET DEBUGGER RECORD FLAG
      5A    0000'CF    9E     0004     89          MOVAB   W^MAC$AB_OBJBUF,R10     ;POINT TO OBJECT BUFFER
            6A    00   90     0009     90          MOVB    #OBJ$C_HDR,(R10)        ;SET RECORD TYPE
      0000'CF    8A    9A     000C     91          MOVZBL  (R10)+,W^MAC$GL_RECTYP  ;MAKE TYPE VISIBLE FOR MAC$WRTOBJ
            8A    00   90     0011     92          MOVB    #OBJ$C_HDR_MHD,(R10)+   ;MODULE HEADER RECORD
                  8A   94     0014     93          CLRB    (R10)+                  ;STRUCTURE LEVEL 0
      8A    0200 8F    B0     0016     94          MOVW    #512,(R10)+             ;RECORD SIZE MAX IS 512 BYTES
                                001B     95 ;
                                001B     96 ; OUTPUT TITLE INFORMATION
                                001B     97 ;
      50    0000'CF    9E     001B     98          MOVAB   W^MAC$AB_TITLE,R0       ;POINT TO TITLE STRING
            51   60    9A     0020     99          MOVZBL  (R0),R1                 ;GET COUNT OF TITLE CHARACTERS
                  0A   12     0023    100          BNEQ    10$                     ;IF NEQ THERE WAS A TITLE STATEMENT
50   00000000'EF    9E     0025    101          MOVAB   MAC$AB_DEF_TITL,R0      ;POINT TO DEFAULT TITLE STRING
            51   60    9A     002C    102          MOVZBL  (R0),R1                 ;GET LENGTH OF DEFAULT TITLE
                  51   D6     002F    103 10$:     INCL    R1                      ;COPY COUNT BYTE ALSO
      6A    60   51    28     0031    104          MOVC3   R1,(R0),(R10)           ;COPY INTO OBJECT BUFFER
            5A   53    D0     0035    105          MOVL    R3,R10                  ;UPDATE CODE POINTER
                                0038    106 ;
                                0038    107 ; OUTPUT IDENT
                                0038    108 ;
            50    0000'CF    9E     0038    109          MOVAB   W^MAC$AB_IDENT,R0       ;POINT TO IDENT STRING
                  51   80    9A     003D    110          MOVZBL  (R0)+,R1                ;GET LENGTH OF IDENT
                       0C   13     0040    111          BEQL    20$                     ;IF EQL NO IDENT--USE DEFAULT
                  8A   51    90     0042    112          MOVB    R1,(R10)+               ;SET LENGTH OF IDENT INTO BUFFER
            6A    60   51    28     0045    113          MOVC3   R1,(R0),(R10)           ;COPY IDENT INTO BUFFER
                  5A   53    D0     0049    114          MOVL    R3,R10                  ;UPDATE POINTER
                       06   11     004C    115          BRB     30$                     ;CONTINUE
                  8A   01    90     004E    116 20$:     MOVB    #1,(R10)+               ;DEFAULT LENGTH OF 1 CHARACTER
                  8A   30    90     0051    117          MOVB    #^A/0/,(R10)+           ;DEFAULT IDENT = '0'
                                0054    118 ;
                                0054    119 ; OUTPUT ASSEMBLY DATE/TIME AND PATCH DATE/TIME
                                0054    120 ;
      6A    0000'CF    11    28     0054    121 30$:     MOVC3   #17,W^MAC$AB_ATIM_BUF,(R10) ;PUT ASSEMBLY TIME INTO BUFFER
63    11    20    6E   00    2C     005A    122          MOVC5   #0,(SP),#^A/ /,#17,(R3) ;BLANK FILL PATCH TIME
                  5A   53    D0     0060    123          MOVL    R3,R10                  ;UPDATE POINTER
                     FF9A'   30     0063    124          BSBW    MAC$WRTOBJ              ;WRITE THE MODULE HEADER RECORD
                                0066    125 ;
                                0066    126 ; OUTPUT SUBHEADER RECORDS
                                0066    127 ;
                  8A   01    90     0066    128          MOVB    #OBJ$C_HDR_LNM,(R10)+   ;LANGUAGE PROCESSOR SUB-HEADER RECORD
      50   00000000'EF    9E     0069    129          MOVAB   MAC$AB_VERSION,R0       ;POINT TO VERSION STRING
            51   80    9A     0070    130          MOVZBL  (R0)+,R1                ;GET LENGTH OF VERSION STRING
      6A    60   51    28     0073    131          MOVC3   R1,(R0),(R10)           ;COPY INTO OBJECT BUFFER
            5A   53    D0     0077    132          MOVL    R3,R10                  ;UPDATE BUFFER POINTER
                     FF83'   30     007A    133          BSBW    MAC$WRTOBJ              ;WRITE OUT THE RECORD
```

```
                                 007D   134 ;
                                 007D   135 ; OUTPUT COMMAND LINE
                                 007D   136 ;
                     8A    02    90  007D   137          MOVB    #OBJ$C_HDR_SRC,(R10)+   ;COMMAND LINE
0000'DF   0000'CF    28          0080   138          MOVC3   W^MAC$GL_CMDLEN,@W^MAC$GL_CMDLIN,- ;COPY COMMAND LINE
                6A               0087   139                                  (R10)              ;INTO CODE BUFFER
                     5A    53    D0  0088   140          MOVL    R3,R10                  ;UPDATE CODE POINTER
                        FF72'    30  008B   141          BSBW    MAC$WRTOBJ              ;OUTPUT THE RECORD
                                 008E   142 ;
                                 008E   143 ; OUTPUT TITLE SUB-TEXT
                                 008E   144 ;
                     8A    03    90  008E   145          MOVB    #OBJ$C_HDR_TTL,(R10)+   ;TITLE RECORD
          50    0000'CF    D0  0091   146          MOVL    W^MAC$GL_TTX_SIZ,R0     ;GET LENGTH OF TITLE SUB-TEXT
                     0B    13  0096   147          BEQL    80$                     ;IF EQL NONE
6A        0000'CF    50    28  0098   148          MOVC3   R0,W^MAC$AB_HD_TSTRG,(R10) ;COPY SUB-TEXT INTO BUFFER
                     5A    53    D0  009E   149          MOVL    R3,R10                  ;UPDATE POINTER
                     06    11  00A1   150          BRB     90$                     ;CONTINUE
                     8A    01    90  00A3   151 80$:     MOVB    #1,(R10)+               ;IF NONE USE DEFAULT
                     8A    20    90  00A6   152          MOVB    #^A/ /,(R10)+           ;SPACE IS DEFAULT
          0000'CF    01    9A  00A9   153 90$:     MOVZBL  #OBJ$C_GSD,W^MAC$GL_RECTYP ;NEXT RECORD IS GSD
                        FF4F'    30  00AE   154          BSBW    MAC$WRTOBJ              ;WRITE OUT LAST HEADER RECORD
```

```
                                00B1   156          .SBTTL  OUTPUT GLOBAL SYMBOL DIRECTORY
                                00B1   157
 56    0000'CF    D0            00B1   158          MOVL    W^MAC$GL_LINK_PTR,R6    ;POINT TO LINKED SYMBOL LIST
                 67    13       00B6   159          BEQL    GSD_DONE                ;IF EQL NO SYMBOL LIST
                                00B8   160 ;
                                00B8   161 ; CHAIN THROUGH THE SYMBOL TABLE (ALREADY FORMED INTO A LINKED LIST)
                                00B8   162 ; AND OUTPUT SYMBOLS TO OBJECT CODE
                                00B8   163 ;
                                00B8   164          .DEBUG  WRT_GSD
                                00B8   165
                                00B8   166 WRT_GSD:
                          0C    B3    00B8   167          BITW    #SYM$M_GLOBL!SYM$M_EXTRN,- ;GLOBAL OR EXTERNAL?
                       09 A6          00BA   168                  SYM$W_FLAG(R6)
                          5C    13    00BC   169          BEQL    60$                     ;IF EQL NO
                09 A6     09    E0    00BE   170          BBS     #SYM$V_EPT,SYM$W_FLAG(R6),60$ ;YES--BRANCH IF EPT
                09 A6     0B    E1    00C3   171          BBC     #SYM$V_RELPSECT,-       ;IF NOT REFERENCED IN A REL PSECT
                                00C8   172                  SYM$W_FLAG(R6),60$      ;THEN DO NOT OUTPUT RECORD
                                00C8   173 ;
                                00C8   174 ; SYMBOL IS GLOBAL OR EXTERNAL, AND IS NOT AN EPT
                                00C8   175 ;
                                00C8   176          $OBJ_OUTBYT_0 #OBJ$C_GSD_SYM    ;SYMBOL DEFINITION
                                00CB   177          $OBJ_OUTBYT_0 #0                ;DATA TYPE (UNKNOWN?)
                          50    D4    00CE   178          CLRL    R0                      ;ASSUME REFERENCE
                09 A6     00    E1    00D0   179          BBC     #SYM$V_DEF,SYM$W_FLAG(R6),20$ ;IS SYMBOL DEFINED?
                          50    02    9A     00D5   180          MOVZBL  #SYM$F_DEF,R0           ;YES
                09 A6     04    E0    00D8   181 20$:     BBS     #SYM$V_ABS,SYM$W_FLAG(R6),30$ ;IS SYMBOL ABSOLUTE?
                          50    08    88     00DD   182          BISB    #SYM$F_REL,R0           ;NO--MARK AS RELOCATABLE
                09 A6     01    E1    00E0   183 30$:     BBC     #SYM$V_WEAK,SYM$W_FLAG(R6),40$ ;IS SYMBOL WEAKLY DEFINED?
                          50    01    88     00E5   184          BISB    #SYM$F_WEAK,R0          ;YES--MARK AS SUCH
                                00E8   185 40$:     $OBJ_OUTBYT_0 R0                ;OUTPUT FLAGS (LOW BYT OE)
                                00EB   186          $OBJ_OUTBYT_0 #0                ;OUTPUT FLAGS (HIGH BYT OE)
             1A 09 A6     00    E1    00EE   187          BBC     #SYM$V_DEF,SYM$W_FLAG(R6),50$ ;BRANCH IF NOT DEFINITION
                50    0C A6     9A    00F3   188          MOVZBL  SYM$B_SEG(R6), R0       ;GET THE PSECT INDEX
                          0A    13    00F7   189          BEQL    45$                     ;DON'T FIDDLE WITH THE ABS PSECT!
                          07    E0    00F9   190          BBS     #SYM$V_REF,-            ;IF THE BLANK PSECT HASN'T BEEN REF'D
          02 00000009'EF          00FB   191                  PSECT$BLANK+SYM$W_FLAG,45$ ;IT'LL BE REMOVED.  DECR THE PSECT
                          50    D7    0101   192          DECL    R0                      ;NUMBER TO PRESERVE ALIGNMENT.
                                0103   193 45$:     $OBJ_OUTBYT_0 R0                ;OUTPUT SYMBOL PSECT INDEX
                55    05 A6     9E    0106   194          MOVAB   SYM$L_VAL(R6),R5        ;POINT TO SYMBOL VALUE
                        FEF3'   30    010A   195          BSBW    MAC$OUT_LW              ;WRITE VALUE TO OBJECT CODE
                        FEF0'   30    010D   196 50$:     BSBW    MAC$SYMNAMOUT           ;WRITE OUT SYMBOL NAME
              0000'8F     5A    B1    0110   197          CMPW    R10,#MAC$AB_OBJWRN      ;ROOM FOR ANOTHER SYMBOL?
                          03    1F    0115   198          BLSSU   60$                     ;IF LSS YES
                        FEE6'   30    0117   199          BSBW    MAC$WRTOBJ              ;NO--DUMP THE BUFFER
                56      66    D0    011A   200 60$:     MOVL    (R6),R6                 ;LINK TO NEXT SYMBOL
                        99    12    011D   201          BNEQ    WRT_GSD                 ;LOOP IF THERE IS MORE
                                011F   202 ;
                                011F   203 ; GSD HAS BEEN OUTPUT
                                011F   204 ;
                                011F   205 70$:
                                011F   206 GSD_DONE:
              0000'CF     06    9A    011F   207          MOVZBL  #OBJ$C_LNK,W^MAC$GL_RECTYP ;ASSUME LINKER OPTION RECORDS PRESENT
              0001'8F     5A    B1    0124   208          CMPW    R10,#MAC$AB_OBJBUF+T    ;FLUSH BUFFER IF ANYTHING THERE
                          03    1B    0129   209          BLEQU   10$                     ;IF LEQ NOTHING THERE
                        FED2'   30    012B   210          BSBW    MAC$WRTOBJ
                                012E   211 ;
                                012E   212 ; WRITE OUT THE LINKER OPTION RECORD(S).
```

```
                     012E   213 ;
00000000'GF    00    FB 012E   214 10$:    CALLS   #0, G^MAC$WRT_LNKOPT    ; WRITE THE RECORDS.
    0000'CF    02    9A 0135   215         MOVZBL  #OBJ$C_TIR,W^MAC$GL_RECTYP ;ASSUME TIR FROM NOW ON
      03 6B    2E    E5 013A   216         BBCC    #FLG$V_DBGOUT,(R11),.+4 ;CLEAR DEBUGGER OUTPUT FLAG
                     05 013E   217         RSB
                        013F   218
                        013F   219         .END
```

```
EOM$C_ABORT     = 00000003
EOM$C_ERROR     = 00000002
EOM$C_SUCCESS   = 00000000
EOM$C_WARNING   = 00000001
FLG$M_ALLCHR    = 00000001
FLG$M_BOL       = 00000002
FLG$M_CHKLPND   = 00100000
FLG$M_COMPEXPR  = 00000004
FLG$M_CONT      = 00000008
FLG$M_CRF       = 40000000
FLG$M_CRSEEN    = 00000001
FLG$M_DATRPT    = 00000010
FLG$M_DBGOUT    = 00004000
FLG$M_DLIMSTR   = 00008000
FLG$M_ENDMCH    = 00000020
FLG$M_EVALEXPR  = 00000040
FLG$M_EXPOPT    = 00000080
FLG$M_EXTERR    = 00010000
FLG$M_EXTWRN    = 00020000
FLG$M_FIRSTLN   = 00000200
FLG$M_IFSTAT    = 00800000
FLG$M_IIF       = 00400000
FLG$M_INSERT    = 00000100
FLG$M_IRPC      = 20000000
FLG$M_LEXOP     = 00000002
FLG$M_LSTXST    = 00000200
FLG$M_MAC2COL   = 00000800
FLG$M_MACL      = 00000800
FLG$M_MACLTB    = 08000000
FLG$M_MACTXT    = 00010000
FLG$M_MEBLST    = 00001000
FLG$M_MOREARG   = 00002000
FLG$M_MOREINP   = 00000008
FLG$M_NEWPND    = 00000400
FLG$M_NOREF     = 01000000
FLG$M_NTYPEPC   = 00000020
FLG$M_NULCHR    = 00040000
FLG$M_OBJXST    = 00200000
FLG$M_OPNDCHK   = 00000100
FLG$M_OPRND     = 00002000
FLG$M_OPTVFLIDX= 00001000
FLG$M_ORDLST    = 00020000
FLG$M_P2        = 00004000
FLG$M_RPTIRP    = 10000000
FLG$M_SEQFIL    = 02000000
FLG$M_SKAN      = 00008000
FLG$M_SPECOP    = 00000004
FLG$M_SPLALL    = 04000000
FLG$M_STOIMF    = 00040000
FLG$M_SYM2COL   = 00000400
FLG$M_TOCFLG    = 00080000
FLG$M_UPAFLG    = 00000010
FLG$M_UPDFIL    = 00000080
FLG$M_UPMARG    = 00000040
FLG$M_XCRF      = 80000000
FLG$V_ALLCHR    = 00000000
FLG$V_BOL       = 00000001
FLG$V_CHKLPND   = 00000014
FLG$V_COMPEXPR  = 00000002
FLG$V_CONT      = 00000003
FLG$V_CRF       = 0000001E
FLG$V_CRSEEN    = 00000020
FLG$V_DATRPT    = 00000004
FLG$V_DBGOUT    = 0000002E
FLG$V_DLIMSTR   = 0000002F
FLG$V_ENDMCH    = 00000005
FLG$V_EVALEXPR  = 00000006
FLG$V_EXPOPT    = 00000007
FLG$V_EXTERR    = 00000030
FLG$V_EXTWRN    = 00000031
FLG$V_FIRSTLN   = 00000029
FLG$V_IFSTAT    = 00000017
FLG$V_IIF       = 00000016
FLG$V_INSERT    = 00000008
FLG$V_IRPC      = 0000001D
FLG$V_LEXOP     = 00000021
FLG$V_LSTXST    = 00000009
FLG$V_MAC2COL   = 0000002B
FLG$V_MACL      = 0000000B
FLG$V_MACLTB    = 0000001B
FLG$V_MACTXT    = 00000010
FLG$V_MEBLST    = 0000000C
FLG$V_MOREARG   = 0000002D
FLG$V_MOREINP   = 00000023
FLG$V_NEWPND    = 0000000A
FLG$V_NOREF     = 00000018
FLG$V_NTYPEPC   = 00000025
FLG$V_NULCHR    = 00000032
FLG$V_OBJXST    = 00000015
FLG$V_OPNDCHK   = 00000028
FLG$V_OPRND     = 0000000D
FLG$V_OPTVFLIDX= 0000002C
FLG$V_ORDLST    = 00000011
FLG$V_P2        = 0000000E
FLG$V_RPTIRP    = 0000001C
FLG$V_SEQFIL    = 00000019
FLG$V_SKAN      = 0000000F
FLG$V_SPECOP    = 00000022
FLG$V_SPLALL    = 0000001A
FLG$V_STOIMF    = 00000012
FLG$V_SYM2COL   = 0000002A
FLG$V_TOCFLG    = 00000013
FLG$V_UPAFLG    = 00000024
FLG$V_UPDFIL    = 00000027
FLG$V_UPMARG    = 00000026
FLG$V_XCRF      = 0000001F
GSD_DONE          0000011F R      03
MAC$AB_ATIM_BUF   ********    X   03
MAC$AB_DEF_TITL   ********    X   03
MAC$AB_HD_TSTRG   ********    X   03
MAC$AB_IDENT      ********    X   03
MAC$AB_OBJBUF     ********    X   03
MAC$AB_OBJWRN     ********    X   03
MAC$AB_TITLE      ********    X   03
MAC$AB_VERSION    ********    X   03
MAC$GL_CMDLEN     ********    X   03
MAC$GL_CMDLIN     ********    X   03
MAC$GL_LINK_PTR   ********    X   03
MAC$GL_RECTYP     ********    X   03
MAC$GL_TTX_SIZ    ********    X   03
MAC$OBJHDROUT     00000000 RG     03
MAC$OUT_LW        ********    X   03
MAC$SYMNAMOUT     ********    X   03
MAC$WRTOBJ        ********    X   03
MAC$WRT_LNKOPT    ********    X   03
OBJ$C_EOM_ABORT= 00000003
OBJ$C_EOM_ERR   = 00000002
OBJ$C_EOM_OK    = 00000000
OBJ$C_EOM_WRN   = 00000001
OBJ$C_GSD       = 00000001
OBJ$C_GSD_SYM   = 00000001
OBJ$C_HDR       = 00000000
OBJ$C_HDR_LNM   = 00000001
OBJ$C_HDR_MHD   = 00000000
OBJ$C_HDR_SRC   = 00000002
OBJ$C_HDR_TTL   = 00000003
OBJ$C_LNK       = 00000006
OBJ$C_TIR       = 00000002
OPF$M_LASTOPR   = 00002000
OPF$M_OPTEXP    = 00001000
OPF$V_LASTOPR   = 0000000D
OPF$V_OPTEXP    = 0000000C
PSC$B_NAME        00000004
PSC$B_SEG         0000000C
PSC$B_UNUSED      0000000B
PSC$K_BLKSIZ      00000013
PSC$K_NO_OPTNS  = 0000000A
PSC$L_CURLOC      0000000F
PSC$L_LINK        00000000
PSC$L_MAXLGTH     00000005
PSC$M_ABS       = FFFFFFF7
PSC$M_ALIGNFLG  = 00004000
PSC$M_ALLOPTNS  = 000003FF
PSC$M_BYTE      = 00004000
PSC$M_CON       = FFFFFFFB
PSC$M_DEFAULT   = 000001C8
PSC$M_EXE       = 000000C0
PSC$M_GBL       = 00000010
PSC$M_LCL       = FFFFFFEF
PSC$M_LIB       = 00000002
PSC$M_LONG      = 00004800
PSC$M_NOEXE     = FFFFFFBF
PSC$M_NOPIC     = FFFFFFFE
PSC$M_NORD      = FFFFFF7F
PSC$M_NOSHR     = FFFFFFDF
PSC$M_NOVEC     = FFFFFCFF
PSC$M_NOWRT     = FFFFFEFF
PSC$M_OVR       = 00000004
PSC$M_PAGE      = 00006400
PSC$M_PIC       = 00000001
PSC$M_QUAD      = 00004C00
```

```
PSC$M_RD          = 00000080
PSC$M_RFL         = 00000008
PSC$M_SHR         = 00000020
PSC$M_USR         = FFFFFF0D
PSC$M_VEC         = 00000200
PSC$M_WORD        = 00004400
PSC$M_WRT         = 00000180
PSC$S_ALIGNMENT   = 00000004
PSC$V_ALIGNFLG    = 0000000E
PSC$V_ALIGNMENT   = 0000000A
PSC$V_EXE         = 00000006
PSC$V_GBL         = 00000004
PSC$V_LIB         = 00000001
PSC$V_OVR         = 00000002
PSC$V_PIC         = 00000000
PSC$V_RD          = 00000007
PSC$V_REL         = 00000003
PSC$V_SHR         = 00000005
PSC$V_VEC         = 00000009
PSC$V_WRT         = 00000008
PSC$W_FLAG          00000009
PSC$W_OPTIONS       0000000D
PSECT$BLANK         ********    X   03
SYM$B_NAME          00000004
SYM$B_SEG           0000000C
SYM$B_TOKEN         0000000B
SYM$F_DEF         = 00000002
SYM$F_REL         = 00000008
SYM$F_UNI         = 00000004
SYM$F_VALIDATE    = 00000010
SYM$F_WEAK        = 00000001
SYM$K_BLKSIZ        0000000D
SYM$K_MAXLEN      = 0000001F
SYM$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
TIR$C_STO_L       = 00000016
TIR$C_STO_LW      = 00000016
WRT_GSD             000000B8 R   D   03
X1                = 00000033
X2                = 00080000
```

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	00000013 (19.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
MAC$RO_CODE_P2	0000013F (319.)	03 (3.)	NOPIC USR CON REL GBL NOSHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.02	00:00:01.59
Command processing	127	00:00:00.34	00:00:02.48
Pass 1	303	00:00:05.85	00:00:25.14
Symbol table sort	0	00:00:00.65	00:00:02.83
Pass 2	58	00:00:01.14	00:00:03.94
Symbol table output	21	00:00:00.11	00:00:00.38
Psect synopsis output	1	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	547	00:00:08.14	00:00:36.39

The working set limit was 1350 pages.
45919 bytes (90 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 696 non-local and 13 local symbols.
219 source lines were read in Pass 1, producing 16 object records in Pass 2.
35 pages of virtual memory were used to define 34 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[MACRO.OBJ]MACRO.MLB;1	4
_255DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	8

861 GETS were required to define 8 macros.

There were no errors, warnings or information messages.

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