


```
AAAAAA  CCCCCCCC  TTTTTTTTTT  000000  PPPPPPPP  CCCCCCCC
AAAAAA  CCCCCCCC  TTTTTTTTTT  000000  PPPPPPPP  CCCCCCCC
AA      AA  CC      TT      00      00  PP      PP  CC
AA      AA  CC      TT      00      00  PP      PP  CC
AA      AA  CC      TT      00      00  PP      PP  CC
AA      AA  CC      TT      00      00  PP      PP  CC
AA      AA  CC      TT      00      00  PPPPPPPP  CC
AAAAA      CC      TT      00      00  PPPPPPPP  CC
AAAAA      CC      TT      00      00  PP      CC
AA      AA  CC      TT      00      00  PP      CC
AA      AA  CC      TT      00      00  PP      CC
AA      AA  CC      TT      00      00  PP      CC
AA      AA  CCCCCCCC  TT      000000  PP      CCCCCCCC
AA      AA  CCCCCCCC  TT      000000  PP      CCCCCCCC
                                     ....
                                     ....
                                     ....
                                     ....
```

```
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
```


(2) 59
(3) 138

DECLARATIONS
DFNOPC ENTRY POINT TO DEFINE AN OPCODE

```
0000 1 .TITLE MAC$ACTOPC DEFINE USER OPCODES
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: 9-OCT-1978
0000 40 :
0000 41 : MODIFIED BY:
0000 42 :
0000 43 : V03-001 MTR0032 Mike Rhodes 21-Apr-1983
0000 44 : Fix computation of available opcode page space.
0000 45 :
0000 46 : V01.05 CNH0049 Chris Hume 12-Feb-1981
0000 47 : Added BB and BW to operand descriptor list for .OPDEF
0000 48 : directive.
0000 49 :
0000 50 : V01.04 RN0023 R. Newland 3-Nov-1979
0000 51 : New message codes to get error messages from system
0000 52 : message file.
0000 53 :
0000 54 : V01.03 RN0005 R. Newland 11-Aug-1979
0000 55 : Variable symbol name storage
0000 56 :
0000 57 :--
```



```
0000 59 .SBTTL DECLARATIONS
0000 60 :
0000 61 : INCLUDE FILES:
0000 62 :
0000 63 :
0000 64 :
0000 65 : MACROS:
0000 66 :
0000 67 :
0000 68 $MAC_GENVALDEF ;GENERAL VALUES
0000 69 $MAC_GRAMMARDEF ;GRAMMAR DEFINITIONS
0000 70 $MAC_SYMBLKDEF ;SYMBOL BLOCK OFFSETS
0000 71 $MAC_OPRDEF ;DEFINE OPERAND DESCRIPTORS
0000 72 $MACMSGDEF ; Define message codes
0000 73 :
0000 74 :
0000 75 : EQUATED SYMBOLS:
0000 76 :
0000 77 :
0000 78 :
0000 79 : OWN STORAGE:
0000 80 :
0000 81 :
00000000 82 .PSECT MAC$RO_DATA,NOWRT,GBL,NOEXE,LONG
0000 83 :
0000 84 :
0000 85 : LIST OF LEGAL OPERAND DESCRIPTORS. THE VALUE IS THE OPERAND DESCRIPTOR
0000 86 : WORD VALUE.
0000 87 :
00000000 0000 88 INSYP = 0
0000 89 :
0000 90 $MAC_INSERT_SYX BB,OPD$M_BB
000C 91 $MAC_INSERT_SYX BW,OPD$M_BW
0018 92 $MAC_INSERT_SYX AB,OPD$M_ADDR+B
0024 93 $MAC_INSERT_SYX AW,OPD$M_ADDR+W
0030 94 $MAC_INSERT_SYX AL,OPD$M_ADDR+L
003C 95 $MAC_INSERT_SYX AF,OPD$M_ADDR+F
0048 96 $MAC_INSERT_SYX AD,OPD$M_ADDR+D
0054 97 $MAC_INSERT_SYX AQ,OPD$M_ADDR+Q
0060 98 $MAC_INSERT_SYX AG,OPD$M_ADDR+G
006C 99 $MAC_INSERT_SYX AO,OPD$M_ADDR+O
0078 100 $MAC_INSERT_SYX AH,OPD$M_ADDR+H
0084 101 $MAC_INSERT_SYX RB,OPD$M_READ+B
0090 102 $MAC_INSERT_SYX RW,OPD$M_READ+W
009C 103 $MAC_INSERT_SYX RL,OPD$M_READ+L
00A8 104 $MAC_INSERT_SYX RF,OPD$M_READ+F
00B4 105 $MAC_INSERT_SYX RD,OPD$M_READ+D
00C0 106 $MAC_INSERT_SYX RQ,OPD$M_READ+Q
00CC 107 $MAC_INSERT_SYX RG,OPD$M_READ+G
00D8 108 $MAC_INSERT_SYX RO,OPD$M_READ+O
00E4 109 $MAC_INSERT_SYX RH,OPD$M_READ+H
00F0 110 $MAC_INSERT_SYX MB,OPD$M_MODIFY+B
00FC 111 $MAC_INSERT_SYX MW,OPD$M_MODIFY+W
0108 112 $MAC_INSERT_SYX ML,OPD$M_MODIFY+L
0114 113 $MAC_INSERT_SYX MF,OPD$M_MODIFY+F
0120 114 $MAC_INSERT_SYX MD,OPD$M_MODIFY+D
012C 115 $MAC_INSERT_SYX MQ,OPD$M_MODIFY+Q
```

0138	116	\$MAC_INSERT_SYX	MG,OPDSM_MODIFY+G
0144	117	\$MAC_INSERT_SYX	MO,OPDSM_MODIFY+O
0150	118	\$MAC_INSERT_SYX	MH,OPDSM_MODIFY+H
015C	119	\$MAC_INSERT_SYX	WB,OPDSM_WRITE+B
0168	120	\$MAC_INSERT_SYX	WW,OPDSM_WRITE+W
0174	121	\$MAC_INSERT_SYX	WL,OPDSM_WRITE+L
0180	122	\$MAC_INSERT_SYX	WF,OPDSM_WRITE+F
018C	123	\$MAC_INSERT_SYX	WD,OPDSM_WRITE+D
0198	124	\$MAC_INSERT_SYX	WQ,OPDSM_WRITE+Q
01A4	125	\$MAC_INSERT_SYX	WG,OPDSM_WRITE+G
01B0	126	\$MAC_INSERT_SYX	WO,OPDSM_WRITE+O
01BC	127	\$MAC_INSERT_SYX	WH,OPDSM_WRITE+H
01C8	128	\$MAC_INSERT_SYX	VB,OPDSM_VIELD+B
01D4	129	\$MAC_INSERT_SYX	VW,OPDSM_VIELD+W
01E0	130	\$MAC_INSERT_SYX	VL,OPDSM_VIELD+L
01EC	131	\$MAC_INSERT_SYX	VF,OPDSM_VIELD+F
01F8	132	\$MAC_INSERT_SYX	VD,OPDSM_VIELD+D
0204	133	\$MAC_INSERT_SYX	VQ,OPDSM_VIELD+Q
0210	134	\$MAC_INSERT_SYX	VG,OPDSM_VIELD+G
021C	135	\$MAC_INSERT_SYX	VO,OPDSM_VIELD+O
0228	136	\$MAC_INSERT_SYX	VH,OPDSM_VIELD+H,MAC\$G_ARGDSCTAB

1 2

```
16-SEP-1984 01:59:49 VAX/VMS Macro V04-00
5-SEP-1984 01:47:00 [MACRO.SRC]ACTOPC.MAR:1
```

Page 4
(3)MAC
V04

```

0000'CF 20 00 61 50 2C 0012 156 MOVCS RO,(R1),#0,#SYM$K_MAXLEN+1, - ; Copy symbol count/name to TMP$SYM
1180 8F BB 0000 152 PUSHR #^M<R7,R8,R12> ;DIRECTIVE = KOPDEF ID EXPR DCOMMA
56 FFF8'CF47 DO 0004 153 MOVL W^MAC$AL_VALSTACK-8[R7],R6 ;SAVE SOME REGISTERS
50 04 A6 9A 000A 154 MOVZBL SYM$B_NAME(R6),R0 ;GET SYMBOL BLOCK ADDRESS
51 56 50 C3 000E 155 SUBL3 R0,R6-R1 ; Get offset to name
00 61 50 2C 0012 156 MOVCS RO,(R1),#0,#SYM$K_MAXLEN+1, - ; Compute address of name
FFEB' 30 001A 157 W^MAC$AB_TMP$SYM ;DIRECTIVE = KOPDEF ID EXPR DCOMMA
FFEB' 30 001A 158 BSBW MAC$HASH-SYM ;SAVE SOME REGISTERS
08 50 E8 0020 159 BSBW MAC$SRCUSR$SYMTB ;GET SYMBOL BLOCK ADDRESS
0000'CF 00 FB 0023 160 BLBS RO,10$ ; Get offset to name
FFD5' 31 0028 161 CALLS #0,W^MAC$ERR INTERN ; Compute address of name
62 66 DO 002B 162 BRW MAC$ABORT_PASS1 ;DIRECTIVE = KOPDEF ID EXPR DCOMMA
57 0000'CF 9E 002E 163 10$: MOVL SYM$L_LINK(R6),SYM$L_LINK(R2) ;UNLINK THIS SYMBOL BLOCK
58 D4 0033 164 MOVAB W^MAC$AB_TMP$BUF,R7 ;POINT TO TEMPORARY BUFFER
0035 165 CLRL R8 ;CLEAR COUNT OF OPERAND DESCRIPTORS
0035 166 ;
0035 167 ; COLLECT OPERAND DESCRIPTORS IN MAC$AB_TMP$BUF WITH A COUNT IN R8
0035 168 ;
0035 169 GET_ARG_DESC:
FFC8' 30 0035 170 BSBW MAC$SKIPSP ;SKIP SPACES
0D 5A 91 0038 171 CMPB R10,#CR ;ARE WE DONE?
3A 13 003B 172 BEQL 40$ ;IF EQL YES
2C 5A 91 003D 173 CMPB R10,#^A/,/ ;COMMA?
03 12 0040 174 BNEQ 5$ ;IF NEQ NO
FFBB' 30 0042 175 BSBW MAC$GETCHR ;YES--SKIP IT
FFBB' 30 0045 176 5$: BSBW MAC$SYMSCNUP ;SCAN A SYMBOL
0D 50 E8 0048 177 BLBS RO,10$ ;BRANCH IF GOT ONE
FFAD' 30 0050 178 $MAC_ERR ILLARGDESC ; Else illegal argument descriptor
5A 0D DO 0053 179 BSBW MAC$ERRORLN ;REPORT THE ERROR
1F 11 0056 180 MOVL #CR,R10 ;NO SENSE IN LOOPING ON ERRORS
55 022B'CF 9E 0058 181 BRB 40$ ;GO FINISH UP
FFA0' 30 005D 182 10$: MOVAB W^MAC$G_ARGDSCTAB,R5 ;POINT TO LIST OF LEGAL DESCRIPTORS
0C 50 E8 0060 183 BSBW MAC$SRC_LIST ;SEE IF A LEGAL DESCRIPTOR
FF95' 30 0063 184 BLBS RO,20$ ;IF LBS YES
87 B4 006B 185 $MAC_ERR ILLARGDESC ; No--get message code
04 11 006D 186 BSBW MAC$ERRORLN ;ISSUE ERROR TO PASS 2
87 05 A1 B0 006F 187 CLRW (R7)+ ;USE DESCRIPTOR OF 0
58 D6 0073 188 BRB 30$ ;CONTINUE
BE 11 0075 189 20$: MOVW SYM$L_VAL(R1),(R7)+ ;COPY ARG DESCRIPTOR VALUE
0077 190 30$: INCL R8 ;COUNT ARG DESCRIPTOR
0077 191 BRB GET_ARG_DESC ;LOOP TILL DONE
0077 192 ;
0077 193 ; ARG DESCRIPTORS HAVE BEEN ACCUMULATED. DEFINE THE OPCODE NOW
0077 194 ;

```



```
57 0000'CF D0 0077 195 40$: MOVL W^MAC$GL_OPCPGPTR,R7 ;GET POINTER TO CURRENT PAGES
      14 12 007C 196 BNEQ 60$ ;BRANCH IF ALLOCATED
      FF7F' 30 007E 197 50$: BSBW MAC$ALL_2_PAGES ;NO--ALLOCATE 2 PAGES
      57 50 D0 0081 198 MOVL R0,R7 ;R7 POINTS TO THE PAGES
0000'CF 50 D0 0084 199 MOVL R0,W^MAC$GL_OPCPGPTR ;SAVE POINTER TO CURRENT PAGES
80 03F8 8F 3C 0089 200 MOVZWL #<<512*2>-85,(R0)+ ;SET BYTE COUNT IN FIRST WORD
80 04 A0 9E 008E 201 MOVAB 4(R0),(R0)+ ;SET POINTER TO FIRST FREE
50 58 02 C5 0092 202 60$: MULL3 #2,R8,R0 ;CALCULATE # BYTES FOR DESCRIPTORS
51 04 A6 9A 0096 203 MOVZBL SYMSB_NAME(R6),R1 ; Get size of symbol count/name
      67 51 C2 009A 204 SUBL2 R1,(R7) ; and subtract from space available.
      67 0D C2 009D 205 SUBL2 #SYMSK_BLKSI2, (R7) ; Adjust size for the symbol block.
      67 50 C2 00A0 206 SUBL2 R0,(R7) ;ROOM ON THESE PAGES?
      D9 19 00A3 207 BLSS 50$ ;IF LSS NO--ALLOCATE NEW PAGES
      5C 04 A7 D0 00A5 208 MOVL 4(R7),R12 ;YES--GET POINTER TO FREE SPOT
50 56 51 C3 00A9 209 SUBL3 R1,R6,R0 ; Get address of symbol count/name
6C 60 51 28 00AD 210 MOVCL3 R1,(R0),(R12) ; and copy into block
63 0000'CF D0 00B1 211 MOVL W^MAC$GL_OPCLSTPT,(R3) ; Link into front of list
0000'CF 83 DE 00B6 212 MOVAL (R3)+,W^MAC$GL_OPCLSTPT
83 04 A6 90 00BB 213 MOVAB SYMSB_NAME(R6),(R3)+
      83 DF 00BF 214 PUSHAL (R3)+
      83 B4 00C1 215 CLRW (R3)+
      83 0E 90 00C3 216 MOVAB #DOPCODE,(R3)+ ; Fill in symbol offset
      83 58 90 00C6 217 MOVAB R8,(R3)+ ;REMEMBER WHERE VALUE GOES
      50 58 02 C5 00C9 218 MULL3 #2,R8,R0 ;(AND BUMP POINTER)
      04 A7 53 D0 00CD 219 MOVCL3 R0,W^MAC$AB_TMPBUF,(R3) ;CLEAR FLAGS
63 0000'CF 50 28 00CD 220 MOVL R3,4(R7) ;TOKEN VALUE IS OPCODE
      50 8ED0 00D7 221 POPL R0 ;SEGMENT IS OPERAND COUNT
      1180 8F BA 00DA 222 POPR #*M<R7,R8,R12> ;FIGURE OPERAND BYTE COUNT
60 FFFC'CF47 D0 00DE 223 MOVL W^MAC$AL_VALSTACK-4[R7],(R0) ;AND COPY DESCRIPTORS
      02 A0 B5 00E4 224 TSTW 2(R0) ;SET POINTER TO NEW FIRST FREE
      08 13 00E7 225 BEQL 80$ ;RETRIEVE VALUE ADDRESS
      FFOF' 30 00EE 226 $MAC_ERR ILLOPDEFVL ;RESTORE REGISTERS
      05 00F1 227 BSBW MAC$ERRORLN ;(R0) ;SET OPCODE VALUE
      00F2 228 RSB ;SEE IF VALUE OK
      00F2 229 80$: ;IF EQL YES
      .END ; No--report the error now
      ;....
```


MAC\$ACTOPC
Symbol table

DEFINE USER OPCODES

K 2

16-SEP-1984 01:59:49
5-SEP-1984 01:47:00VAX/VMS Macro V04-00
[MACRO.SRC]ACTOPC.MAR;1Page 6
(3)MAC
V04

AB	=	00000001	DSUP	=	0000002F	KDSABL	=	00000056
AD	=	0000C008	DTIMES	=	0000001B	KENABL	=	00000057
AF	=	00008004	DUPA	=	00000023	KEND	=	00000076
AG	=	0000A008	DUPB	=	00000024	KENDC	=	0000004E
AH	=	00009010	DUPC	=	00000025	KENDM	=	00000053
AL	=	00000004	DUPD	=	00000026	KENDR	=	0000004F
AO	=	00000010	DUPF	=	00000028	KENTRY	=	00000058
AQ	=	00000008	DUPM	=	00000029	KERROR	=	00000071
ARG\$K_SIZE	=	000003E8	DUPQ	=	00000027	KEVEN	=	0000005B
AUD\$K_SIZE	=	00000010	DUPX	=	0000002A	KEXTRN	=	0000005D
AW	=	00000002	DWUP	=	00000030	KFIELD	=	0000003A
B	=	00000001	DXOR	=	0000001F	KFLOAT	=	0000003B
BLNK	=	00000020	ERR01	=	00000001	KGFLOAT	=	00000081
CHR\$M_COMMA_CR	=	00000020	ERR02	=	00000002	KGLOBL	=	0000005E
CHR\$M_ILL_CHR	=	00000040	ERR03	=	00000003	KHFLOAT	=	00000082
CHR\$M_NUM_BER	=	00000010	ERR04	=	00000004	KIDENT	=	0000006A
CHR\$M_SPA_MSK	=	00000001	ERR05	=	00000005	KIF	=	00000046
CHR\$M_SYM_CH1	=	00000008	ERR06	=	00000006	KIFF	=	00000048
CHR\$M_SYM_CHR	=	00000004	ERR07	=	00000007	KIFT	=	00000049
CHR\$M_SYM_DLM	=	00000002	ERR08	=	00000008	KIFTF	=	0000004A
CHR\$V_COMMA_CR	=	00000005	ERR09	=	00000009	KIIF	=	00000047
CHR\$V_CVTLWC	=	00000061	F	=	00008004	KINCLUDE	=	0000005F
CHR\$V_ILL_CHR	=	00000006	FF	=	0000000C	KIRP	=	0000004B
CHR\$V_NOCVT	=	0000007F	G	=	0000A008	KIRPC	=	0000004C
CHR\$V_NUM_BER	=	00000004	GET_ARG_DESC	=	00000035 R	KLIBRARY	=	00000060
CHR\$V_SPA_MSK	=	00000000	GOALSY	=	0000000A	KLINK	=	00000085
CHR\$V_SYM_CH1	=	00000003	H	=	00009010	KLIST	=	00000061
CHR\$V_SYM_CHR	=	00000002	HASHSZ	=	0000007F	KLONG	=	0000003C
CHR\$V_SYM_DLM	=	00000001	HYPHEN	=	0000002D	KMACRO	=	00000050
CR	=	0000000D	ID	=	0000000C	KMCALL	=	00000051
D	=	0000C008	INP\$K_BUFSIZ	=	000003E8	KMDELETE	=	00000054
DAND	=	0000001D	INSYMC	=	00000002	KMEXIT	=	00000052
DANGCLS	=	00000016	INSYMP	=	0000022B R	KNARG	=	00000063
DANGOPN	=	00000015	INSYTM	=	0000022B R	KNCHR	=	00000064
DAT	=	00000020	INT\$K_BUFSIZ	=	000013F4	KNCROS	=	0000007A
DBUP	=	0000002B	INT\$K_BUFWRN	=	00001390	KNLIST	=	00000062
DCLS	=	00000018	KADDRESS	=	00000037	KNTYPE	=	00000074
DCOLON	=	00000010	KALIGN	=	0000005A	KOCTA	=	00000083
DCOMMA	=	0000000F	KASCIC	=	00000033	KODD	=	0000005C
DDIV	=	0000001C	KASCID	=	00000078	KOPDEF	=	00000075
DEOL	=	0000000B	KASCI1	=	00000034	KPACKED	=	00000036
DEQ	=	00000011	KASCI2	=	00000035	KPAGE	=	00000065
DFNOPC	=	00000000 RG	KBLKA	=	0000003F	KPRINT	=	00000072
DGUP	=	0000002C	KBLKB	=	00000040	KPSECT	=	00000066
DINTEGER	=	00000022	KBLKD	=	00000041	KQUAD	=	0000003D
DIUP	=	0000002D	KBLKF	=	00000042	KREF1	=	0000006D
DLUP	=	0000002E	KBLKG	=	0000007E	KREF16	=	00000084
DMASK	=	00000032	KBLKH	=	0000007F	KREF2	=	0000006E
DMINUS	=	0000001A	KBLKL	=	00000043	KREF4	=	0000006F
DOPCODE	=	0000000E	KBLKO	=	00000080	KREF8	=	00000070
DOPN	=	00000017	KBLKQ	=	00000044	KREPT	=	0000004D
DOR	=	0000001E	KBLKW	=	00000045	KRESTORE	=	00000067
DPC	=	00000012	KBYTE	=	00000038	KSAVE	=	00000068
DPLUS	=	00000019	KCROSS	=	00000079	KSBTTL	=	0000006B
DPOUND	=	00000021	KDEBUG	=	00000055	KSGNB	=	0000007C
DSQCLS	=	00000014	KDFLT	=	0000007B	KSGNW	=	0000007D
DSQOPN	=	00000013	KDOUBLE	=	00000039	KTITLE	=	00000069

MAC\$ACTOPC
Symbol table

DEFINE USER OPCODES

L 2

16-SEP-1984 01:59:49
5-SEP-1984 01:47:00

VAX/VMS Macro V04-00
[MACRO.SRC]ACTOPC.MAR;1

Page 7
(3)

KVECTOR = 00000059
KWARN = 00000073
KWEAK = 0000006C
KWORD = 0000003E
KXFER = 00000077
L = 00000004
LST\$K_BUFSIZ = 00000086
LST\$K_L_P_PAGE = 0000003C
LST\$K_TITLE_SIZ = 00000028
MAC\$ABORT_PASS1 ***** X 04
MAC\$AB_TMPBUF ***** X 04
MAC\$AB_TMPSYM ***** X 04
MAC\$ALC_2_PAGES ***** X 04
MAC\$AL_VACSTACK ***** X 04
MAC\$ERRORLN ***** X 04
MAC\$ERR_INTERN ***** X 04
MAC\$GETCHR ***** X 04
MAC\$GL_OPCLSTPT ***** X 04
MAC\$GL_OPCPGPTR ***** X 04
MAC\$G_ARGDSCTAB 0000022B RG 03
MAC\$HASH_SYM ***** X 04
MAC\$SKIPSP ***** X 04
MAC\$SRCUSRSYMTB ***** X 04
MAC\$SRC_LIST ***** X 04
MAC\$SYM\$CNUP ***** X 04
MAC\$_ILLARGDESC = 007D90AA
MAC\$_ILLOPDEFVL = 007D9112
MAC\$XT = 0000000D
MAC\$_SUBSYS = 0000007D
MB = 00000041
MD = 0000C048
MF = 00008044
MG = 0000A048
MH = 00009050
ML = 00000044
MO = 00000050
MQ = 00000048
MW = 00000042
O = 00000010
OBJ\$K_BUFSIZ = 00000200
OPD\$M_ADDR = 00000000
OPD\$M_BB = 000000A1
OPD\$M_BW = 000000C2
OPD\$M_D_FLOAT = 0000C000
OPD\$M_F_FLOAT = 00008000
OPD\$M_G_FLOAT = 0000A000
OPD\$M_H_FLOAT = 00009000
OPD\$M_MODE = 000003E0
OPD\$M_MODIFY = 00000040
OPD\$M_NOT_32F = 00007000
OPD\$M_READ = 00000020
OPD\$M_VIELD = 00000080
OPD\$M_WRITE = 00000060
OPD\$S_MODE = 00000005
OPD\$S_SIZE = 00000005
OPD\$V_D_FLOAT = 0000000E
OPD\$V_F_FLOAT = 0000000F

OPD\$V_G_FLOAT = 0000000D
OPD\$V_H_FLOAT = 0000000C
OPD\$V_MODE = 00000005
OPD\$V_SIZE = 00000000
PSC\$B_NAME = 00000004
PSC\$B_SEG = 0000000C
PSC\$B_UNUSED = 0000000B
PSC\$K_BLK\$IZ = 00000013
PSC\$K_NO_OPTS = 0000000A
PSC\$K_CURLOC = 0000000F
PSC\$K_LINK = 00000000
PSC\$K_MAXLGTH = 00000005
PSC\$M_ABS = FFFFFFFF7
PSC\$M_ALIGNFLG = 00004000
PSC\$M_ALLOPTNS = 000003FF
PSC\$M_BYTE = 00004000
PSC\$M_CON = FFFFFFFFB
PSC\$M_DEFAULT = 000001C8
PSC\$M_EXE = 000000C0
PSC\$M_GBL = 00000010
PSC\$M_LCL = FFFFFFFEF
PSC\$M_LIB = 00000002
PSC\$M_LONG = 00004800
PSC\$M_NOEXE = FFFFFFFBF
PSC\$M_NOPIC = FFFFFFFFE
PSC\$M_NORD = FFFFFFF7F
PSC\$M_NOSHR = FFFFFFFDF
PSC\$M_NOVEC = FFFFFFFDF
PSC\$M_NOWRT = FFFFFFFEF
PSC\$M_OVR = 00000004
PSC\$M_PAGE = 00006400
PSC\$M_PIC = 00000001
PSC\$M_QUAD = 00004C00
PSC\$M_RD = 00000080
PSC\$M_REL = 00000008
PSC\$M_SHR = 00000020
PSC\$M_USR = FFFFFFFFD
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
Q = 00000008
RB = 00000021

RD = 0000C028
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
RF = 00008024
RG = 0000A028
RH = 00009030
RL = 00000024
RO = 00000030
RQ = 00000028
RRREG = 00000031
RW = 00000022
SEMI = 0000003B
STB\$K_PG_MISS = 0000000A
SYM\$B_NAME = 00000004
SYM\$B_SEG = 0000000C
SYM\$B_TOKEN = 0000000B
SYM\$K_BLK\$IZ = 0000000D
SYM\$K_MAXLEN = 0000001F
SYM\$K_TWOCOL = 00000010
SYM\$K_LINK = 00000000
SYM\$K_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

MAC
V04

MAC\$ACTOPC
Symbol table

DEFINE USER OPCODES

M 2

16-SEP-1984 01:59:49 VAX/VMS Macro V04-00
5-SEP-1984 01:47:00 [MACRO.SRC]ACTOPC.MAR;1

Page 8
(3)

SYMSV_SUPR = 0000000E
SYMSV_WEAK = 00000001
SYMSV_XCRF = 0000000C
SYMSW_FLAG = 00000009
TAB = 00000009
VB = 00000081
VD = 0000C088
VF = 00008084
VG = 0000A088
VH = 00009090
VL = 00000084
VO = 00000090
VQ = 00000088
VW = 00000082
W = 00000002
WB = 00000061
WD = 0000C068
WF = 00008064
WG = 0000A068
WH = 00009070
WL = 00000064
WO = 00000070
WQ = 00000068
WW = 00000062
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
\$ABSS	00000013 (19.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
MAC\$RO_DATA	00000234 (564.)	03 (3.)	NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG
MAC\$RO_CODE_P1	000000F2 (242.)	04 (4.)	NOPIC USR CON REL GBL NOSHR EXE RD NOWRT NOVEC LONG

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.06	00:00:01.60
Command processing	103	00:00:00.34	00:00:02.64
Pass 1	199	00:00:03.11	00:00:14.52
Symbol table sort	0	00:00:00.38	00:00:01.01
Pass 2	61	00:00:00.65	00:00:02.38
Symbol table output	30	00:00:00.15	00:00:00.62
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	426	00:00:04.72	00:00:22.81

The working set limit was 1050 pages.
27628 bytes (54 pages) of virtual memory were used to buffer the intermediate code.

There were 30 pages of symbol table space allocated to hold 446 non-local and 9 local symbols.
231 source lines were read in Pass 1, producing 19 object records in Pass 2.
14 pages of virtual memory were used to define 10 macros.

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

Macro library name	Macros defined
-----	-----
\$255\$DUA28:[MACRO.OBJ]MACRO.MLB;1	7
\$255\$DUA28:[SYSLIB]STARLET.MLB;2	3
TOTALS (all libraries)	10

537 GETS were required to define 10 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:ACTOPC/OBJ=OBJ\$:ACTOPC MSRC\$:ACTOPC/UPDATE=(ENH\$:ACTOPC)+LIB\$:MACRO/LIB

0224 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

