```
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR      RRR            TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR      RRR            TTT        LLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
```

FILEID**BASDET

```
BBBBBBBB      AAAAAA      SSSSSSSS  DDDDDDDD    EEEEEEEEEE  TTTTTTTTTT
BBBBBBBB      AAAAAA      SSSSSSSS  DDDDDDDD    EEEEEEEEEE  TTTTTTTTTT
BB      BB   AA    AA   SS        DD      DD  EE              TT
BB      BB   AA    AA   SS        DD      DD  EE              TT
BB      BB   AA    AA   SS        DD      DD  EE              TT
BB      BB   AA    AA   SS        DD      DD  EE              TT
BBBBBBBB     AA    AA     SSSSSS  DD      DD  EEEEEEEE        TT
BBBBBBBB     AA    AA     SSSSSS  DD      DD  EEEEEEEE        TT
BB      BB   AAAAAAAAAA         SS DD      DD  EE              TT
BB      BB   AAAAAAAAAA         SS DD      DD  EE              TT
BB      BB   AA    AA           SS DD      DD  EE              TT          ....
BB      BB   AA    AA           SS DD      DD  EE              TT          ....
BBBBBBBB     AA    AA   SSSSSSSS   DDDDDDDD    EEEEEEEEEE      TT          ....
BBBBBBBB     AA    AA   SSSSSSSS   DDDDDDDD    EEEEEEEEEE      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  BAS$DET                 ; fetch and store DET
0000     2          .IDENT  /1-008/                 ; File: BASDET.MAR      Edit: MDL1008
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: BASIC Language Support
0000    31 ;
0000    32 ; ABSTRACT:
0000    33 ;
0000    34 ; This module has routines to store a double, gfloat, or hfloat value into the
0000    35 ; OWN storage that has the determinant of the last matrix inverted.  The store
0000    36 ; entry point is used by the BASIC initializer to initialize the DET to 0
0000    37 ; and by the matrix inversion routines to store the determinant.  There are
0000    38 ; entry points to retrieve the determinant as either a float, double, g floating,
0000    39 ; or h floating.  (The proper entry point depends on how the determinant was
0000    40 ; saved.)
0000    41 ;
0000    42 ; ENVIRONMENT: User Mode, AST Reentrant
0000    43 ;
0000    44 ;--
0000    45 ; AUTHOR: R. Will, CREATION DATE:  24-Jul-79
0000    46 ;
0000    47 ; MODIFIED BY:
0000    48 ;
0000    49 ;++      , : VERSION 1
0000    50 ; 1-001 - Original
0000    51 ; 1-002 - Correct some typos.  JBS 25-JUL-1979
0000    52 ; 1-003 - Add scaling comments.  RW  31-Dec-1979
0000    53 ; 1-004 - Add entry points for g and h floating.  PL 30-Sep-81
0000    54 ; 1-005 - Change entry point BAS$DET_H_R3 to BAS$DET_H since it is
0000    55 ;         a CALL entry point.  PL 15-Oct-81
0000    56 ; 1-006 - Add G^ to BAS$$SCALE_R1 call.  PLL 22-Mar-1982
0000    57 ; 1-007 - store the determinant in H_floating, and then convert it to the
```

```
0000     58 ;          desired data type when called for.  MDL 16-Jan-1984
0000     59 ; 1-008 - fix a bug in BAS$$STORE_DET.  MDL 6-Mar-1984
0000     60 ;--
```

```
                     0000     62          .SBTTL  DECLARATIONS
                     0000     63 ;
                     0000     64 ; INCLUDE FILES:
                     0000     65 ;
                     0000     66          $SFDEF                                  ; use to get scale
                     0000     67 ;
                     0000     68 ; EXTERNAL DECLARATIONS:
                     0000     69 ;
                     0000     70          .DSABL  GBL                             ; Prevent undeclared
                     0000     71                                                  ; symbols from being
                     0000     72                                                  ; automatically global.
                     0000     73          .EXTRN  BAS$$SCALE_R1                   ; get the scale for double
                     0000     74 ;
                     0000     75 ; MACROS:
                     0000     76 ;
                     0000     77
                     0000     78 ;
                     0000     79 ; EQUATED SYMBOLS:
                     0000     80 ;
                     0000     81
                     0000     82 ;
                     0000     83 ; OWN STORAGE:
                     0000     84 ;
                     0000     85
                 00000000     86          .PSECT _BAS$DATA PIC, USR, CON, REL, LCL, NOSHR, NOEXE, -
                     0000     87                         RD, WRT
            00000010 0000     88 DET:     .BLKL   4               ; 4 longwords can contain hfloat,
                     0010     89                                  ; gfloat, or dbl determinant
                     0010     90
                     0010     91 ;
                     0010     92 ; PSECT DECLARATIONS:
                     0010     93 ;
                     0010     94          .PSECT _BAS$CODE PIC, USR, CON, REL, LCL, SHR, -
                 00000000     95                         EXE, RD, NOWRT, LONG
                     0000     96
```

```
                                98          .SBTTL  BAS$DET_F  -  Fetch the determinant as a floating point value
                                99 ;++
                               100 ; FUNCTIONAL DESCRIPTION:
                               101 ;
                               102 ;       The determinant is stored in H_floating. Convert it to F_floating and
                               103 ;       return the value in R0.
                               104 ;
                               105 ; CALLING SEQUENCE:
                               106 ;
                               107 ;       CALL BAS$DET_F
                               108 ;
                               109 ; INPUT PARAMETERS:
                               110 ;
                               111 ;       NONE
                               112 ;
                               113 ; IMPLICIT INPUTS:
                               114 ;
                               115 ;       NONE
                               116 ;
                               117 ; OUTPUT PARAMETERS:
                               118 ;
                               119 ;       NONE
                               120 ;
                               121 ; IMPLICIT OUTPUTS:
                               122 ;
                               123 ;       NONE
                               124 ;
                               125 ; FUNCTION VALUE:
                               126 ; COMPLETION CODES:
                               127 ;
                               128 ;       The rounded floating point value of the OWN storage
                               129 ;
                               130 ; SIDE EFFECTS:
                               131 ;
                               132 ;       NONE
                               133 ;
                               134 ;--
                               135
                     001C      136          .ENTRY BAS$DET_F, ^M<R2,R3,R4>  ; Entry point
                               137
          50    0C AD    D0    138          MOVL    SF$L_SAVE_FP(FP), R0    ; pass FP to get scale
      00000000'GF      16    139          JSB     G^BAS$$SCALE_R1         ; get scale in R0 & R1
          54    50     76    140          CVTDF   R0, R4                  ; cvt scale factor to desired data type
50    00000000'EF F6FD    141          CVTHF   DET, R0                 ; cvt determinant to desired data type
          50    54     46    142          DIVF2   R4, R0                  ; descale
                     04     143          RET
```

```
                        001B   145          .SBTTL  BAS$DET_D  -   Return the double precision value of DET
                        001B   146 ;++
                        001B   147 ; FUNCTIONAL DESCRIPTION:
                        001B   148 ;
                        001B   149 ;       The determinant is stored in H_floating. Convert it to D_floating and
                        001B   150 ;       return the value in R0.
                        001B   151 ;
                        001B   152 ; CALLING SEQUENCE:
                        001B   153 ;
                        001B   154 ;       CALL BAS$DET_D
                        001B   155 ;
                        001B   156 ; INPUT PARAMETERS:
                        001B   157 ;
                        001B   158 ;       NONE
                        001B   159 ;
                        001B   160 ; IMPLICIT INPUTS:
                        001B   161 ;
                        001B   162 ;       NONE
                        001B   163 ;
                        001B   164 ; OUTPUT PARAMETERS:
                        001B   165 ;
                        001B   166 ;       NONE
                        001B   167 ;
                        001B   168 ; IMPLICIT OUTPUTS:
                        001B   169 ;
                        001B   170 ;       NONE
                        001B   171 ;
                        001B   172 ; FUNCTION VALUE:
                        001B   173 ; COMPLETION CODES:
                        001B   174 ;
                        001B   175 ;       The double precision value in DET, scaled if scaling is present
                        001B   176 ;
                        001B   177 ; SIDE EFFECTS:
                        001B   178 ;
                        001B   179 ;       NONE
                        001B   180 ;
                        001B   181 ;--
                        001B   182
                  0000  001B   183          .ENTRY BAS$DET_D, ^M<>          ; Entry point
                        001D   184
50   00000000'EF  F7FD  001D   185          CVTHD   DET, R0
                  04    0025   186          RET
```

```
                        0026   188          .SBTTL  BAS$DET_G  -   Return the gfloat value of DET
                        0026   189 ;++
                        0026   190 ; FUNCTIONAL DESCRIPTION:
                        0026   191 ;
                        0026   192 ;       The determinant is stored in H_floating. Convert it to D_floating and
                        0026   193 ;       return the value in R0.
                        0026   194 ;
                        0026   195 ; CALLING SEQUENCE:
                        0026   196 ;
                        0026   197 ;       CALL BAS$DET_G
                        0026   198 ;
                        0026   199 ; INPUT PARAMETERS:
                        0026   200 ;
                        0026   201 ;       NONE
                        0026   202 ;
                        0026   203 ; IMPLICIT INPUTS:
                        0026   204 ;
                        0026   205 ;       NONE
                        0026   206 ;
                        0026   207 ; OUTPUT PARAMETERS:
                        0026   208 ;
                        0026   209 ;       NONE
                        0026   210 ;
                        0026   211 ; IMPLICIT OUTPUTS:
                        0026   212 ;
                        0026   213 ;       NONE
                        0026   214 ;
                        0026   215 ; FUNCTION VALUE:
                        0026   216 ; COMPLETION CODES:
                        0026   217 ;
                        0026   218 ;       The g floating value in DET
                        0026   219 ;
                        0026   220 ; SIDE EFFECTS:
                        0026   221 ;
                        0026   222 ;       NONE
                        0026   223 ;
                        0026   224 ;--
                        0026   225
                  00FC  0026   226          .ENTRY BAS$DET_G, ^M<R2,R3,R4,R5,R6,R7> ; Entry point
                        0028   227
      50    0C AD     D0  0028   228          MOVL    SF$L_SAVE_FP(FP), R0    ; pass FP to get scale
    00000000'GF       16  002C   229          JSB     G^BAS$$SCALE_R1         ; get scale in R0 & R1
        54    50    32FD  0032   230          CVTDH   R0, R4                  ; cvt scale factor to desired data type
                        0036   231                                          ; no CVTDG so we promote to H
50  00000000'EF     70FD  0036   232          MOVH    DET, R0                 ; cvt determinant to desired data type
        50    54    66FD  003E   233          DIVH2   R4, R0                  ; descale
        54    50    70FD  0042   234          MOVH    R0, R4                  ; move out of the way
        50    54    76FD  0046   235          CVTHG   R4, R0                  ; cvt to desired data type
                    04    004A   236          RET
```

```
                     004B   238          .SBTTL  BAS$DET_H  -  Return the hfloat value of DET
                     004B   239 ;++
                     004B   240 ; FUNCTIONAL DESCRIPTION:
                     004B   241 ;
                     004B   242 ;       Return the value in DET
                     004B   243 ;
                     004B   244 ; CALLING SEQUENCE:
                     004B   245 ;
                     004B   246 ;       CALL BAS$DET_H
                     004B   247 ;
                     004B   248 ; INPUT PARAMETERS:
                     004B   249 ;
                     004B   250 ;       NONE
                     004B   251 ;
                     004B   252 ; IMPLICIT INPUTS:
                     004B   253 ;
                     004B   254 ;       NONE
                     004B   255 ;
                     004B   256 ; OUTPUT PARAMETERS:
                     004B   257 ;
                     004B   258 ;       NONE
                     004B   259 ;
                     004B   260 ; IMPLICIT OUTPUTS:
                     004B   261 ;
                     004B   262 ;       NONE
                     004B   263 ;
                     004B   264 ; FUNCTION VALUE:
                     004B   265 ; COMPLETION CODES:
                     004B   266 ;
                     004B   267 ;       The h floating value in DET
                     004B   268 ;
                     004B   269 ; SIDE EFFECTS:
                     004B   270 ;
                     004B   271 ;       NONE
                     004B   272 ;
                     004B   273 ;--
                     004B   274
                 00FC 004B   275          .ENTRY BAS$DET_H, ^M<R2,R3,R4,R5,R6,R7>          ; Entry point
                     004D   276
      50   0C AD    D0 004D   277          MOVL    SF$L_SAVE_FP(FP), R0     ; pass FP to get scale
    00000000'GF    16 0051   278          JSB     G^BAS$$SCALE_R1          ; get scale in R0 & R1
      54    50 32FD 0057   279          CVTDH   R0, R4                   ; cvt scale factor to desired data type
50  00000000'EF 70FD 005B   280          MOVH    DET, R0                  ; cvt determinant to desired data type
      50    54 66FD 0063   281          DIVH2   R4, R0                   ; descale
                 04 0067   282          RET
```

```
                     0068   284          .SBTTL  BAS$$STORE_DET  - Put a value into the OWN storage
                     0068   285 ;++
                     0068   286 ; FUNCTIONAL DESCRIPTION:
                     0068   287 ;
                     0068   288 ;       Store the value passed.
                     0068   289 ;
                     0068   290 ; CALLING SEQUENCE:
                     0068   291 ;
                     0068   292 ;       CALL BAS$$STORE_DET (determinant_value.rd.r)
                     0068   293 ;
                     0068   294 ; INPUT PARAMETERS:
                     0068   295 ;
                     0068   296
00000004             0068   297          determinant = 4                 ; determinant must already be scaled
                     0068   298                                          ; if scaling is present
                     0068   299
                     0068   300 ;
                     0068   301 ; IMPLICIT INPUTS:
                     0068   302 ;
                     0068   303 ;       NONE
                     0068   304 ;
                     0068   305 ; OUTPUT PARAMETERS:
                     0068   306 ;
                     0068   307 ;       NONE
                     0068   308 ;
                     0068   309 ; IMPLICIT OUTPUTS:
                     0068   310 ;
                     0068   311 ;       NONE
                     0068   312 ;
                     0068   313 ; FUNCTION VALUE:
                     0068   314 ; COMPLETION CODES:
                     0068   315 ;
                     0068   316 ;       NONE
                     0068   317 ;
                     0068   318 ; SIDE EFFECTS:
                     0068   319 ;
                     0068   320 ;       NONE
                     0068   321 ;
                     0068   322 ;--
                     0068   323
                0000 0068   324          .ENTRY BAS$$STORE_DET, ^M<>             ; Entry point
                     006A   325
00000000'EF    04 BC 32FD   006A   326          CVTDH   @determinant(AP), DET   ; store the value
                   04       0073   327          RET
                     0074   328
```

```
                     0074   330          .SBTTL  BAS$$STORE_DET_G - Put a value into the OWN storage
                     0074   331 ;++
                     0074   332 ; FUNCTIONAL DESCRIPTION:
                     0074   333 ;
                     0074   334 ;       Store the value passed.
                     0074   335 ;
                     0074   336 ; CALLING SEQUENCE:
                     0074   337 ;
                     0074   338 ;       CALL BAS$$STORE_DET_G (determinant_value.rg.r)
                     0074   339 ;
                     0074   340 ; INPUT PARAMETERS:
                     0074   341 ;
                     0074   342
            00000004 0074   343          determinant = 4
                     0074   344
                     0074   345 ;
                     0074   346 ; IMPLICIT INPUTS:
                     0074   347 ;
                     0074   348 ;       NONE
                     0074   349 ;
                     0074   350 ; OUTPUT PARAMETERS:
                     0074   351 ;
                     0074   352 ;       NONE
                     0074   353 ;
                     0074   354 ; IMPLICIT OUTPUTS:
                     0074   355 ;
                     0074   356 ;       NONE
                     0074   357 ;
                     0074   358 ; FUNCTION VALUE:
                     0074   359 ; COMPLETION CODES:
                     0074   360 ;
                     0074   361 ;       NONE
                     0074   362 ;
                     0074   363 ; SIDE EFFECTS:
                     0074   364 ;
                     0074   365 ;       NONE
                     0074   366 ;
                     0074   367 ;--
                     0074   368
                0000 0074   369          .ENTRY BAS$$STORE_DET_G, ^M<>          ; Entry point
                     0076   370
00000000'EF    04 BC 56FD   0076   371          CVTGH   @determinant(AP), DET          ; store the value
                   04 007F   372          RET
                     0080   373
```

```
                     0080   375          .SBTTL  BAS$$STORE_DET_H -  Put a value into the OWN storage
                     0080   376 ;++
                     0080   377 ; FUNCTIONAL DESCRIPTION:
                     0080   378 ;
                     0080   379 ;       Store the value passed.
                     0080   380 ;
                     0080   381 ; CALLING SEQUENCE:
                     0080   382 ;
                     0080   383 ;       CALL BAS$$STORE_DET_H (determinant_value.rh.r)
                     0080   384 ;
                     0080   385 ; INPUT PARAMETERS:
                     0080   386 ;
                     0080   387
           00000004  0080   388          determinant = 4
                     0080   389
                     0080   390 ;
                     0080   391 ; IMPLICIT INPUTS:
                     0080   392 ;
                     0080   393 ;       NONE
                     0080   394 ;
                     0080   395 ; OUTPUT PARAMETERS:
                     0080   396 ;
                     0080   397 ;       NONE
                     0080   398 ;
                     0080   399 ; IMPLICIT OUTPUTS:
                     0080   400 ;
                     0080   401 ;       NONE
                     0080   402 ;
                     0080   403 ; FUNCTION VALUE:
                     0080   404 ; COMPLETION CODES:
                     0080   405 ;
                     0080   406 ;       NONE
                     0080   407 ;
                     0080   408 ; SIDE EFFECTS:
                     0080   409 ;
                     0080   410 ;       NONE
                     0080   411 ;
                     0080   412 ;--
                     0080   413
                0000 0080   414          .ENTRY BAS$$STORE_DET_H, ^M<>          ; Entry point
                     0082   415
00000000'EF   04 BC 70FD 0082   416          MOVH    @determinant(AP), DET          ; store the value
                   04    008B   417          RET
                     008C   418
                     008C   419          .END
```

```
BAS$$SCALE_R1                   ********  X   00
BAS$$STORE_DET                  00000068 RG   03
BAS$$STORE_DET_G                00000074 RG   03
BAS$$STORE_DET_H                00000080 RG   03
BAS$DET_D                       0000001B RG   03
BAS$DET_F                       00000000 RG   03
BAS$DET_G                       00000026 RG   03
BAS$DET_H                       0000004B RG   03
DET                             00000000 R    02
DETERMINANT                   = 00000004
SF$L_SAVE_FP                  = 0000000C
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
_BAS$DATA	00000010 (16.)	02 (2.)	PIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC BYTE
_BAS$CODE	0000008C (140.)	03 (3.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.07	00:00:00.68
Command processing	118	00:00:00.45	00:00:03.07
Pass 1	118	00:00:01.27	00:00:04.55
Symbol table sort	0	00:00:00.02	00:00:00.02
Pass 2	77	00:00:00.78	00:00:01.96
Symbol table output	3	00:00:00.02	00:00:00.03
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	349	00:00:02.64	00:00:10.34

The working set limit was 1050 pages.
6122 bytes (12 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 39 non-local and 0 local symbols.
419 source lines were read in Pass 1, producing 33 object records in Pass 2.
8 pages of virtual memory were used to define 7 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	4

88 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:BASDET/OBJ=OBJ$:BASDET MSRC$:BASDET/UPDATE=(ENH$:BASDET)

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

BASDET
LIS

BASDISPAT
LIS

BASCVTTP
LIS

BASDELETE
LIS

BASDATETI
LIS

BASECHO
LIS

BASCVTRP
LIS