```
MMM         MMM  TTTTTTTTTTTTTTT  HHH         HHH  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
MMM         MMM  TTTTTTTTTTTTTTT  HHH         HHH  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
MMM         MMM  TTTTTTTTTTTTTTT  HHH         HHH  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
MMMMMM   MMMMMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMMMMM   MMMMMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMMMMM   MMMMMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMM   MMM   MMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMM   MMM   MMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMM   MMM   MMM        TTT        HHH         HHH  RRR         RRR         TTT        LLL
MMM         MMM        TTT        HHHHHHHHHHHHHHH  RRRRRRRRRRR           TTT        LLL
MMM         MMM        TTT        HHHHHHHHHHHHHHH  RRRRRRRRRRR           TTT        LLL
MMM         MMM        TTT        HHHHHHHHHHHHHHH  RRRRRRRRRRR           TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR   RRR             TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR   RRR             TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR   RRR             TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR      RRR          TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR      RRR          TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR      RRR          TTT        LLL
MMM         MMM        TTT        HHH         HHH  RRR         RRR       TTT        LLLLLLLLLLLLLLL
MMM         MMM        TTT        HHH         HHH  RRR         RRR       TTT        LLLLLLLLLLLLLLL
MMM         MMM        TTT        HHH         HHH  RRR         RRR       TTT        LLLLLLLLLLLLLLL
```

```
MM      MM  TTTTTTTTTT  HH      HH  HH      HH  FFFFFFFFFF  LL          000000      000000    RRRRRRRR
MM      MM  TTTTTTTTTT  HH      HH  HH      HH  FFFFFFFFFF  LL          000000      000000    RRRRRRRR
MMMM  MMMM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR      RR
MMMM  MMMM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR      RR
MM  MM  MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR      RR
MM  MM  MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR      RR
MM      MM      TT      HHHHHHHHHH  HHHHHHHHHH  FFFFFFFF    LL        00      00  00      00  RRRRRRRR
MM      MM      TT      HHHHHHHHHH  HHHHHHHHHH  FFFFFFFF    LL        00      00  00      00  RRRRRRRR
MM      MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR  RR
MM      MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR  RR
MM      MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR    RR        ....
MM      MM      TT      HH      HH  HH      HH  FF          LL        00      00  00      00  RR    RR        ....
MM      MM      TT      HH      HH  HH      HH  FF          LLLLLLLLLL    000000      000000    RR      RR      ....
MM      MM      TT      HH      HH  HH      HH  FF          LLLLLLLLLL    000000      000000    RR      RR      ....


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      .TITLE  MTH$HFLOOR - Great. t integer routine for H_floating
0000      .IDENT  /1-002/                 ; File: MTHHFLOOR.MAR  EDIT: RH1002
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: Math Library
0000  ;
0000  ; ABSTRACT:
0000  ;
0000  ;       This routine finds the largest integer less than the input
0000  ;       value, i.e. it truncates toward negative infinity
0000  ;       for data type H_floating.
0000  ;
0000  ; ENVIRONMENT: User Mode, AST Reentrant
0000  ;
0000  ;--
0000  ; Author: John Sauter, Creation date: 27-JUL-1979
0000  ;
0000  ; MODIFIED BY:
0000  ;
0000  ; VERSION 00
0000  ; 1-001 - Original, from MTH$GFLOOR.
0000  ; 1-002 - CALL entry was modified to return the result in the address
0000  ;         specified by the leftmost argument in order to conform to the
0000  ;         calling standard for return values larger than 64 bits.  The
0000  ;         original version returned the results in R0-R3.
0000  ;         JSB entry was modify to correct typos: The last two operands
0000  ;         of the EMODH instruction and the operand of the TSTH instruction
0000  ;         were changed from R2 to R4.
0000  ;         Comments were changed to eliminate an erroneous calling sequence.
0000  ;         RNH 9-DEC-1980
```

```
     0000          .SBTTL  DECLARATIONS
     0000    ;
     0000    ; INCLUDE FILES:
     0000    ;

     0000    ;
     0000    ; EXTERNAL DECLARATIONS.
     0000    ;
     0000          .DSABL  GBL                             ; Prevent undeclared
     0000                                                  ; symbols from being
     0000                                                  ; automatically global.
     0000    ;
     0000    ; MACROS:
     0000    ;

     0000    ;
     0000    ; EQUATED SYMBOLS:
     0000    ;

     0000    ;
     0000    ; OWN STORAGE:
     0000    ;

     0000    ;
     0000    ; PSECT DECLARATIONS:
     0000    ;
 00000000          .PSECT _MTH$CODE PIC, USR, CON, REL, LCL, SHR, -
     0000                        EXE, RD, NOWRT, LONG

     0000
```

```
                        0000          .SBTTL  MTH$HFLOOR  - greatest integer H_floating routine
                        0000  ;++
                        0000  ; FUNCTIONAL DESCRIPTION:
                        0000  ;
                        0000  ;       This routine finds the floor by truncating, and then if the
                        0000  ;       input value is negative and not an integer subtracting 1.
                        0000  ;
                        0000  ; CALLING SEQUENCE:
                        0000  ;
                        0000  ;       CALL MTH$HFLOOR (result_int.wh.r, input.rh.r)
                        0000  ;
                        0000  ; INPUT PARAMETERS:
                        0000  ;
        00000008        0000          input_addr = 8                  ; address of the H_floating number
                        0000                                          ; to get the floor of
                        0000  ;
                        0000  ; IMPLICIT INPUTS:
                        0000  ;
                        0000  ;       NONE
                        0000  ;
                        0000  ; OUTPUT PARAMETERS:
                        0000  ;
        00000004        0000          output_addr = 4
                        0000  ;
                        0000  ; IMPLICIT OUTPUTS:
                        0000  ;
                        0000  ;       NONE
                        0000  ;
                        0000  ; FUNCTION VALUE:
                        0000  ; COMPLETION CODES:
                        0000  ;
                        0000  ;       NONE
                        0000  ;
                        0000  ; SIDE EFFECTS:
                        0000  ;
                        0000  ;       NONE
                        0000  ;
                        0000  ;--
                        0000
                  00FC  0000          .ENTRY  MTH$HFLOOR, ^M<R2, R3, R4. R5, R6, R7> ; entry point
                        0002
          50   08 BC 70FD 0002        MOVH    @input_addr(AP), R0           ; R0/R3 = input argument
54   54   08   00   50 74FD 0007      EMODH   R0, #0, #1, R4, R4            ; R4/R7 = fraction_part (arg)
               50   54 62FD 000E      SUBH2   R4, R0                        ; R0/R3 = integer_part (arg)
                        0012
               09   14 0012          BGTR    40$                           ; if > 0, have correct answer
                        0014
               54 73FD 0014          TSTH    R4                            ; look at fraction part
               04   18 0017          BGEQ    40$                           ; if > 0 then 0 < input < 1 and
                        0019                                                ;  we have the correct answer
                        0019                                                ; if = 0 then input was integer
                        0019                                                ;  and we have correct answer
                        0019
               50   08 62FD 0019      SUBH2   #1,R0                         ; subtract 1 from truncated
                        001D                                                ; negative non-integer
                        001D
          04 BC   50 70FD 001D  40$:  MOVH    R0, @output_addr(AP)          ; move result to output address
```

```
                                    H  1
 04  0022   144          RET
```

```
                                        .SBTTL  MTH$HFLOOR_R7  - greatest integer H_floating routine
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;       This is the JSB entry point to MTH$HFLOOR.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSB  MTH$HFLOOR_R7
                                ;
                                ; INPUT PARAMETERS:
                                ;
                                ;       R0 through R3 contain the input value
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       NONE
                                ;
                                ; OUTPUT PARAMETERS:
                                ;
                                ;       R0 through R3 contain the result value
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       NONE
                                ;
                                ; FUNCTION VALUE:
                                ; COMPLETION CODES:
                                ;
                                ;       NONE
                                ;
                                ; SIDE EFFECTS:
                                ;
                                ;       NONE
                                ;
                                ;--

                                MTH$HFLOOR_R7::                                 ; entry point

54   54   08   00   50 74FD    EMODH   R0, #0, #1, R4, R4                      ; R4/R7 = fraction_part (arg)
                50   54 62FD    SUBH2   R4, R0                                  ; R0/R3 = integer_part (arg)

                     09   14    BGTR    40$                                     ; if > 0, have correct answer

                     54 73FD    TSTH    R4                                      ; look at fraction part
                     04   18    BGEQ    40$                                     ; if > 0 then 0 < input < 1 and
                                                                                ;  we have the correct answer
                                                                                ; if = 0 then input was integer
                                                                                ;  and we have correct answer

                50   08 62FD    SUBH2   #1,R0                                   ; subtract 1 from truncated
                                                                                ; negative non-integer

                          05    40$:    RSB

                                        .END
```

```
INPUT_ADDR      = 00000008
MTH$HFLOOR        00000000 RG    01
MTH$HFLOOR_R7     00000023 RG    01
OUTPUT_ADDR     = 00000004
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes										
. ABS .	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
_MTH$CODE	0000003A (58.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	31	00:00:00.13	00:00:01.98
Command processing	128	00:00:00.51	00:00:07.16
Pass 1	72	00:00:00.54	00:00:01.82
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	50	00:00:00.47	00:00:02.05
Symbol table output	2	00:00:00.00	00:00:00.01
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	287	00:00:01.69	00:00:13.07

The working set limit was 900 pages.
2169 bytes (5 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 4 non-local and 2 local symbols.
201 source lines were read in Pass 1, producing 11 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

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