

To our customers,

---

## Old Company Name in Catalogs and Other Documents

---

On April 1<sup>st</sup>, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1<sup>st</sup>, 2010  
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

## Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
  - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
  - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
  - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

# H8/300L Super Low Power Series

## Multiplication of Single-Precision Floating-Point Numbers (FMUL)

---

### Introduction

The software FMUL performs multiplication of single-precision floating-point numbers, which are placed in general-purpose registers, and places the result of multiplication in the general purpose registers.

### Target Device

H8/38024

### Contents

1. Arguments.....	2
2. Changes to Internal Registers and Flags .....	2
3. Specifications .....	2
4. Notes.....	3
5. Description .....	3
6. Flowchart.....	7
7. Program List.....	16
About Single-Precision Floating-Point Numbers <Reference> .....	21

### 1. Arguments

Description		Memory area	Data length (bytes)
Input	Multiplicand	R0, R1	4
	Multiplier	R2, R3	4
Output	Result of multiplication	R0, R1	4

### 2. Changes to Internal Registers and Flags

R0	R1	R2	R3	R4	R5	R6	R7
○	○	×	×	×	×	×	—
I	U	H	U	N	Z	V	C
—	—	×	—	×	×	×	×

#### Legend

- : No change
- ×: Undefined
- : Result

### 3. Specifications

Program memory (bytes)	348
Data memory (bytes)	0
Stack (bytes)	16
Clock cycle count	1078
Reentrant	Possible
Relocation	Possible
Interrupt	Possible

### 4. Notes

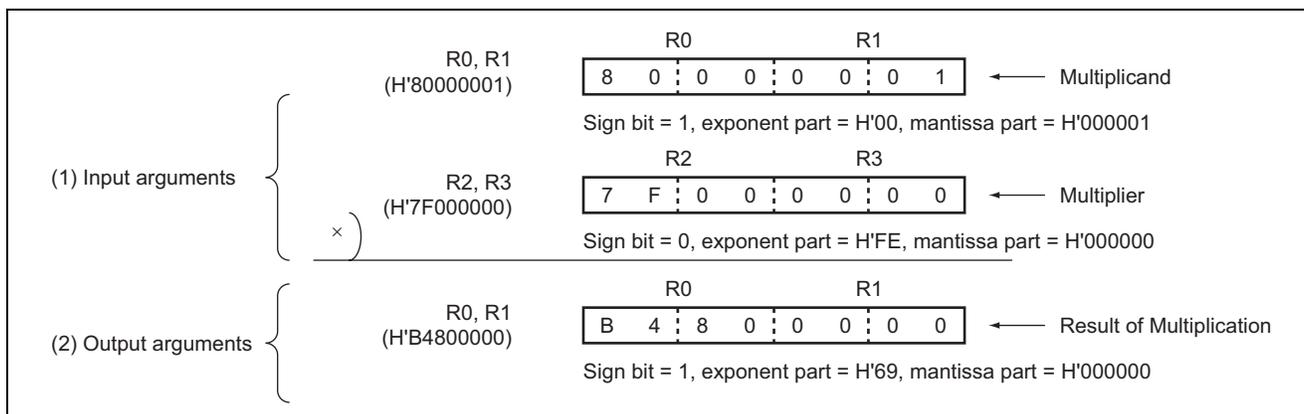
The clock cycle count (16) in the specifications is for the example shown in figure 1.

For the format of floating-point numbers, see "About Single-Precision Floating-Point Numbers <Reference>."

### 5. Description

#### 5.1 Details of functions

1. The following arguments are used with the software FMUL:
  - a. Input arguments:
    - R0: Sets the upper 2 bytes of a single-precision floating-point as multiplicand.
    - R1: Sets the lower 2 bytes of the single-precision floating-point as multiplicand.
    - R2: Sets the upper 2 bytes of a single-precision floating-point as multiplier.
    - R3: Sets the lower 2 bytes of the single-precision floating-point as multiplier.
  - b. Output arguments:
    - R0: The upper 2 bytes of a single-precision floating-point are placed here as the result of multiplication.
    - R1: The lower 2 bytes of a single-precision floating-point are placed here as the result of multiplication.
2. The following figure illustrates the execution of the software FMUL. When the input arguments are set as shown in (1), the result of multiplication is placed in R0 and R1 as shown in (2).



**Figure 1 Example of Software FMUL Execution**

## 5.2 Notes on usage

1. The maximum and minimum values that can be handled by the software FADD are as follows:

{ Positive maximum H'7F800000  
 Positive minimum H'00000001

{ Negative maximum H'80000001  
 Negative minimum H'FF800000

2. All positive single-precision floating-point numbers H'7F800001 to H'7FFFFFFF are treated as a maximum value (H'7F800000). All negative single-precision floating-point numbers H'FF800000 to H'FFFFFFFF are treated as a minimum value (H'FF800000).

3. As a maximum value is treated as infinity ( $\infty$ ),  $\infty \times 100 = \infty$  or  $\infty \times (-100) = -\infty$  (see table 1).

**Table 1 Examples of Operation with Maximum Values Used as Arguments**

Multiplicand	Multiplier	Result
>H'7F800000	Positive number	H'7F800000 (+ $\infty$ )
(+ $\infty$ )	Negative number	H'FF800000 (- $\infty$ )
<H'FF800000	Positive number	H'FF800000 (- $\infty$ )
(- $\infty$ )	Negative number	H'7F800000 (+ $\infty$ )
Positive number	>H'7F800000 (+ $\infty$ )	H'7F800000 (+ $\infty$ )
	<H'FF800000 (- $\infty$ )	H'FF800000 (- $\infty$ )
Negative number	>H'7F800000 (+ $\infty$ )	H'FF800000 (- $\infty$ )
	<H'FF800000 (- $\infty$ )	H'7F800000 (+ $\infty$ )

4. H'80000000 is treated as H'00000000 (zero).

5. After execution of the software FMUL, the multiplicand and multiplier data will be lost. When the input arguments are still needed after software FMUL execution, save them in memory.

## 5.3 Description of data memory

The software FMUL uses no data memory.

### 5.4 Example of usage

Set a multiplicand and a multiplier in the general-purpose registers and call the software FMUL as a subroutine.

WORK1	. RES. B	2	-----	{ Reserve a data memory area in which the user program places       }	{ a multiplicand. a multiplier. a result of multiplication.       }
WORK2	. RES. B	2			
WORK3	. RES. B	2			
MOV. W		@WORK1, R0	}-----	{ Place the multiplicand set by the user program in R0 and R1.       }	
MOV. W		@WORK1+2, R1			
MOV. W		@WORK2, R2	}-----	{ Place the multiplier set by the user program in R2 and R3.       }	
MOV. W		@WORK2+2, R3			
		JSR	@FMUL	-----	{ Call the software FMUL as a subroutine.
MOV. W		R0, @WORK3	}-----	{ Place the result of multiplication set in the output argument in R0 and R1 .       }	
MOV. W		R1 @WORK3+2			
	.				
	.				
	.				

## 5.5 Operation

Multiplication of single-precision floating-point numbers is done in the following steps:

1. The software checks whether the multiplicand and multiplier are "0".
  - a. If either the multiplicand or multiplier is "0", H'00000000 is output.
2. The software checks whether the multiplicand and multiplier are infinite.  
 If they are infinite, the result is as given in table 1.
3. Assume that the multiplicand is  $R_1$  (sign bit =  $S_1$ , exponent =  $\alpha_1$ , mantissa =  $\beta_1$ ) and the multiplier is  $R_2$  (sign bit =  $S_2$ , exponent =  $\alpha_2$ , mantissa =  $\beta_2$ ). Then  $R_1$  and  $R_2$  are given by

$$R_1 = (-1)^{S_1} \times 2^{\alpha_1 - 127} \times \beta_1$$

$$R_2 = (-1)^{S_2} \times 2^{\alpha_2 - 127} \times \beta_2$$

Multiplication of these two numbers is given by

$$R_1 \times R_2 = (-1)^{S_1 + S_2} \times 2^{\alpha_1 + \alpha_2 - 127 - 127} \times \beta_1 \times \beta_2$$

Since, in the case of the floating-point format, H'7F (D'127) is added to the result of multiplication of the exponents, the multiplication equation changes as follows:

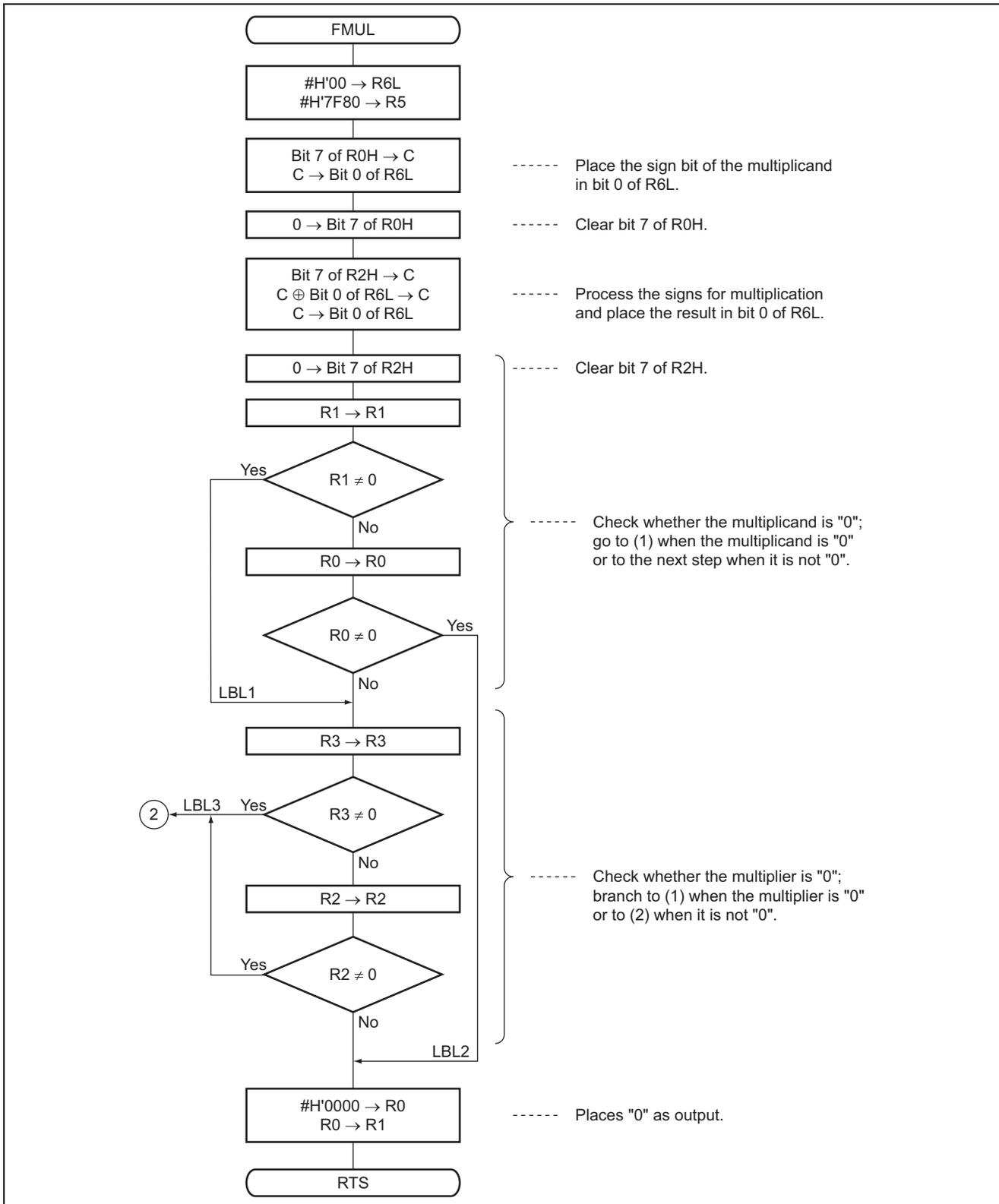
$$R_1 \times R_2 = (-1)^{S_1 + S_2} \times 2^{\alpha_1 + \alpha_2 - 127} \times \beta_1 \times \beta_2$$

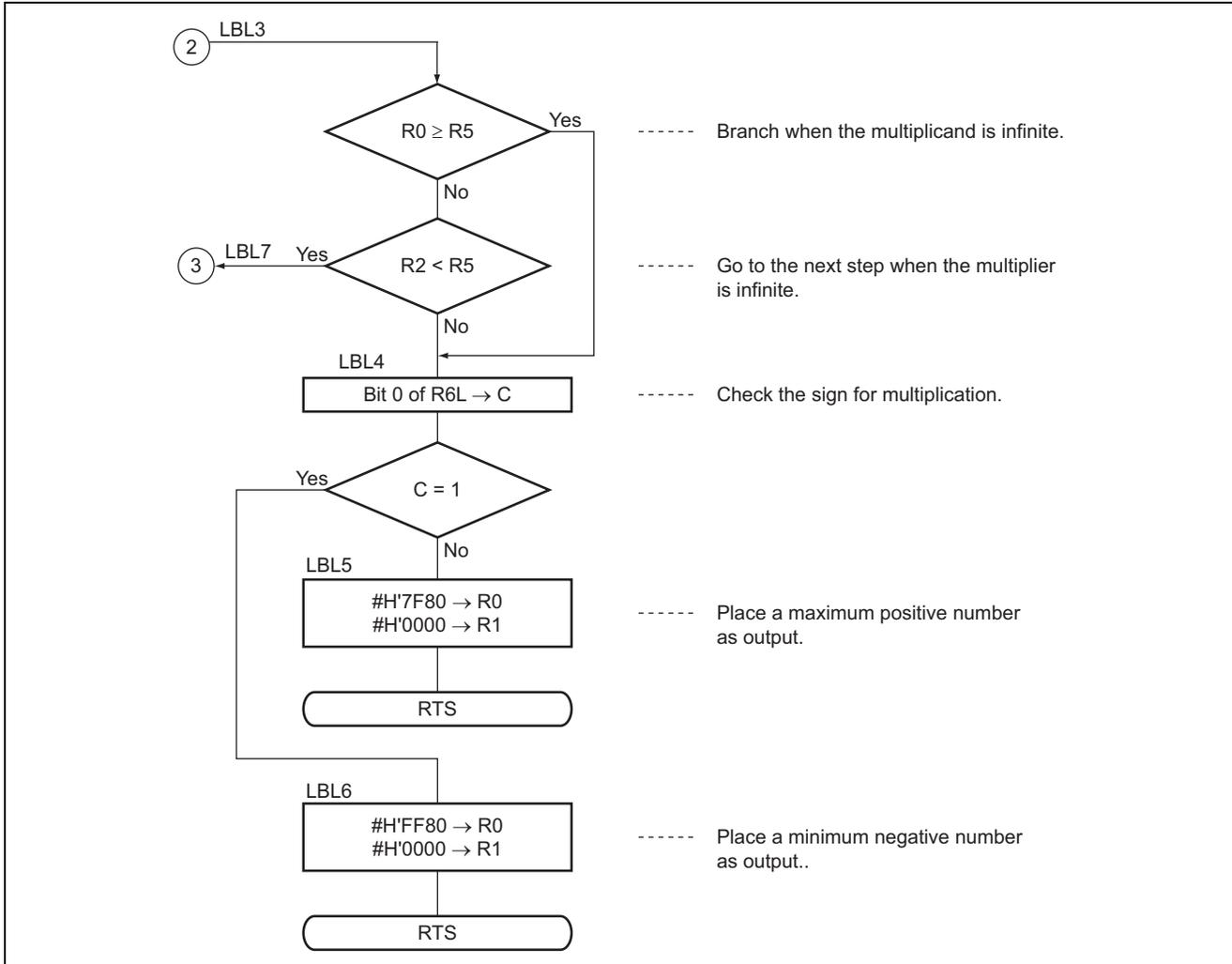
Thus, the multiplication is performed in the steps below:

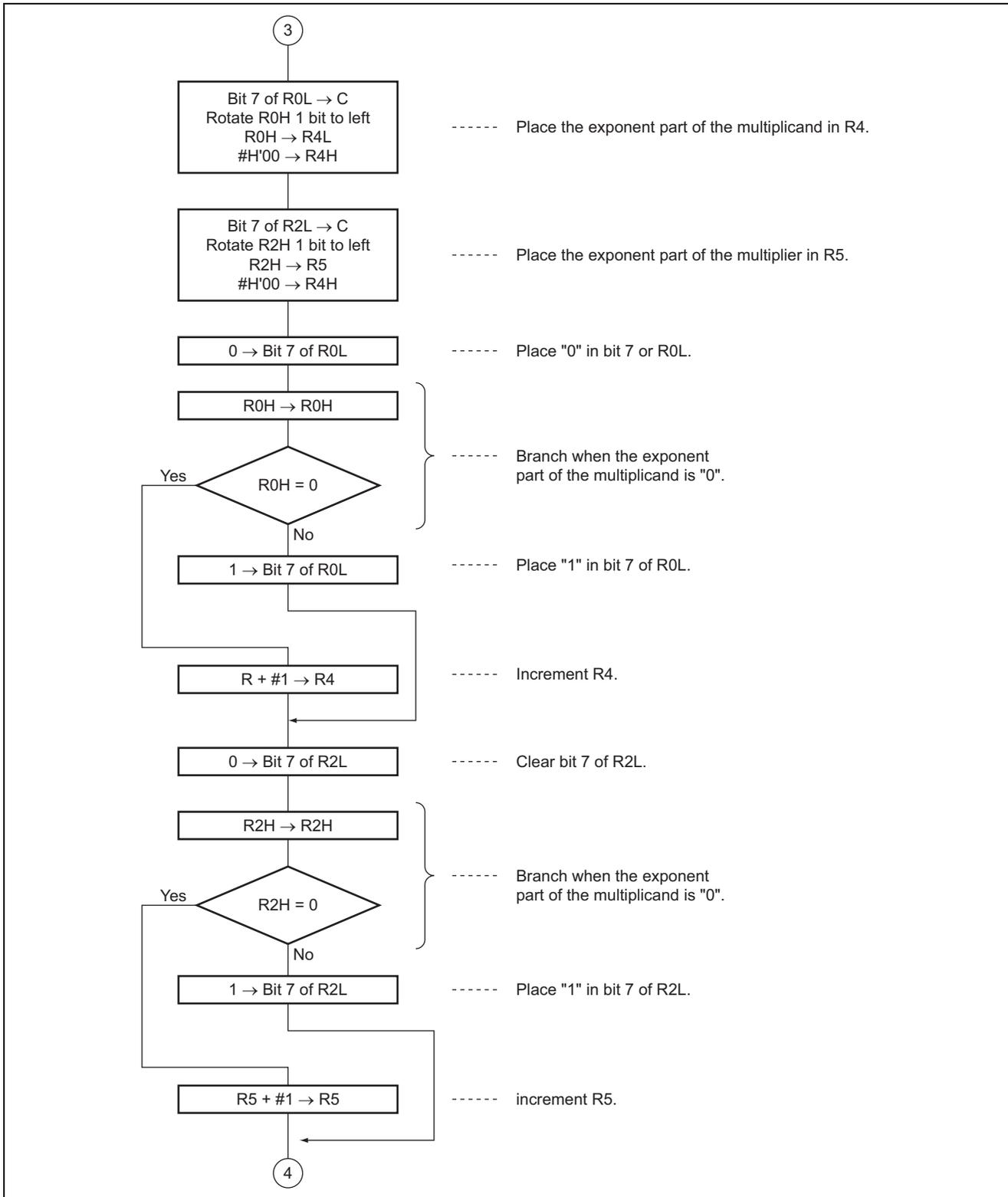
- a. The software checks the sign bits of  $R_1 \times R_2$ .
- b. Addition is done on the exponents.  
 H'7F (D'127) is added to the actual exponent of a number in the floating-point data format; H'7F (D'127) is thus subtracted from both  $\alpha_1$  and  $\alpha_2$ , and H'7F (D'127) is added to the exponent of the result. The result may thus be expressed as follows.  

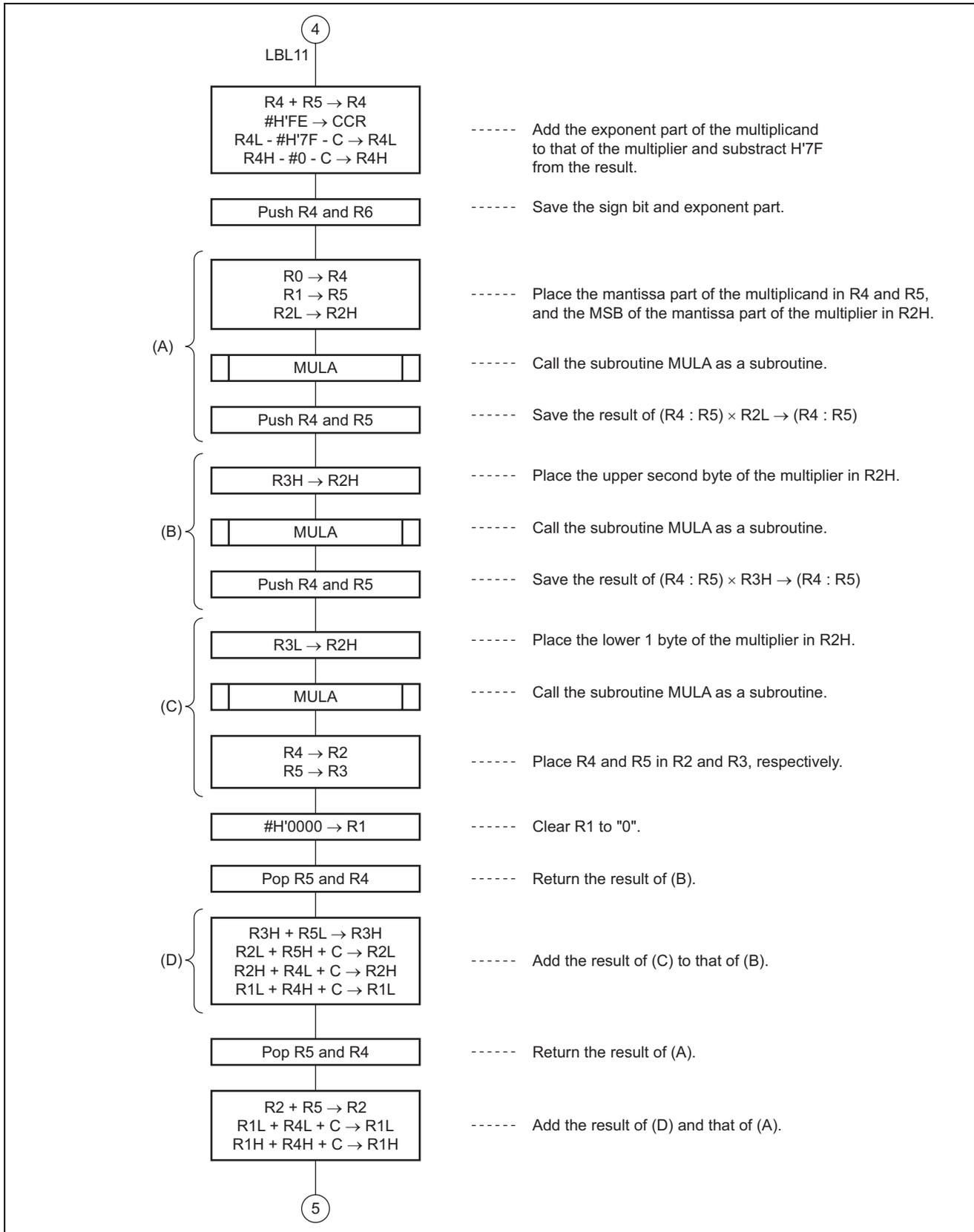
$$(\alpha_1 - \text{H'7F}) + (\alpha_2 - \text{H'7F}) + \text{H'7F} = \alpha_1 + \alpha_2 - \text{H'7F}$$
 (In the case of the denormalized format, 1 is added to the exponent before the calculation.)
- c. Multiplication is done on the mantissas.  
 The implicit MSB is included in this operation.  
 (In the case of the denormalized format, the implicit MSB of the mantissa is treated as "0".)
- d. The result of multiplication is corrected to produce a number in the floating-point data format.

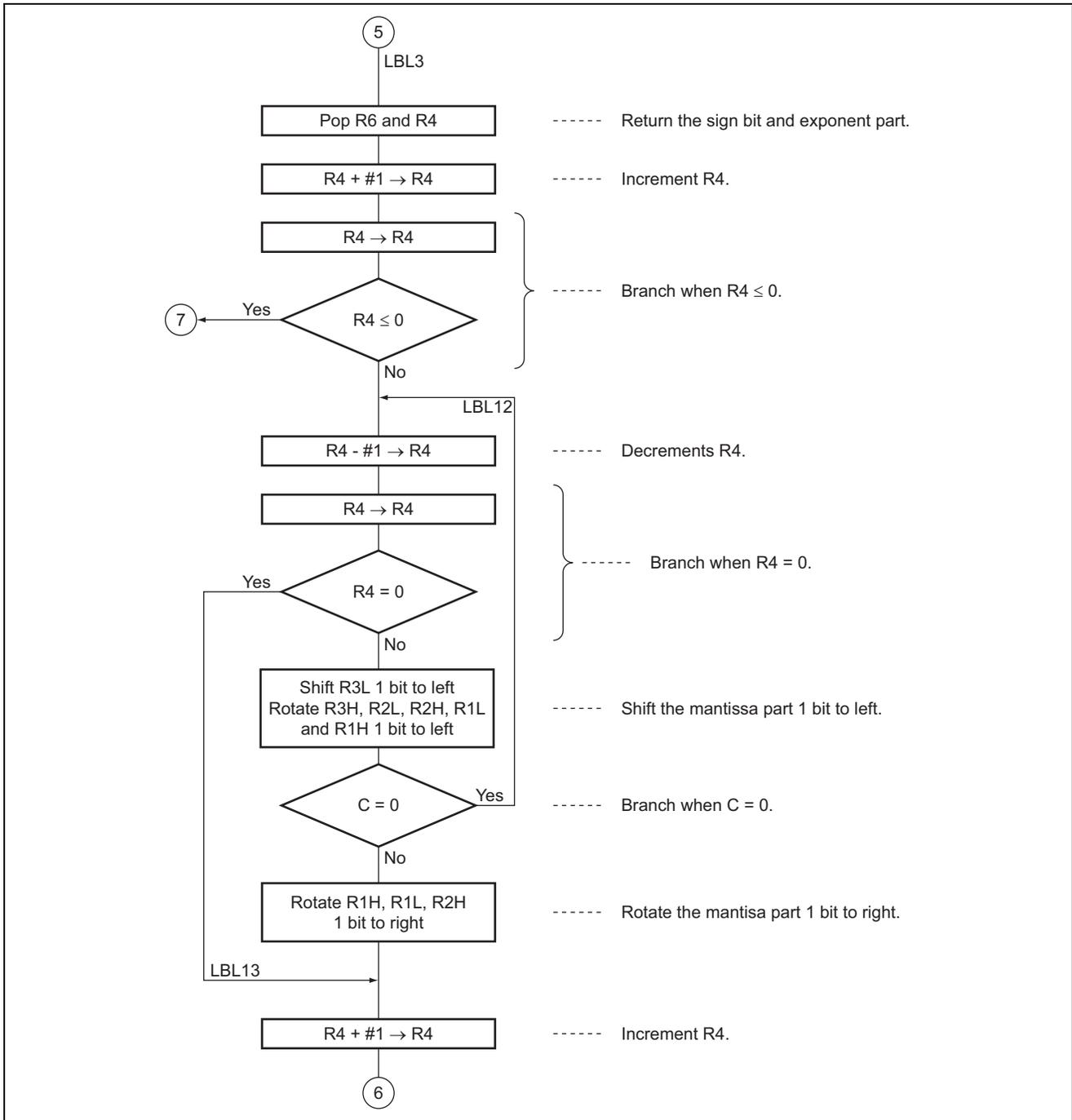
### 6. Flowchart

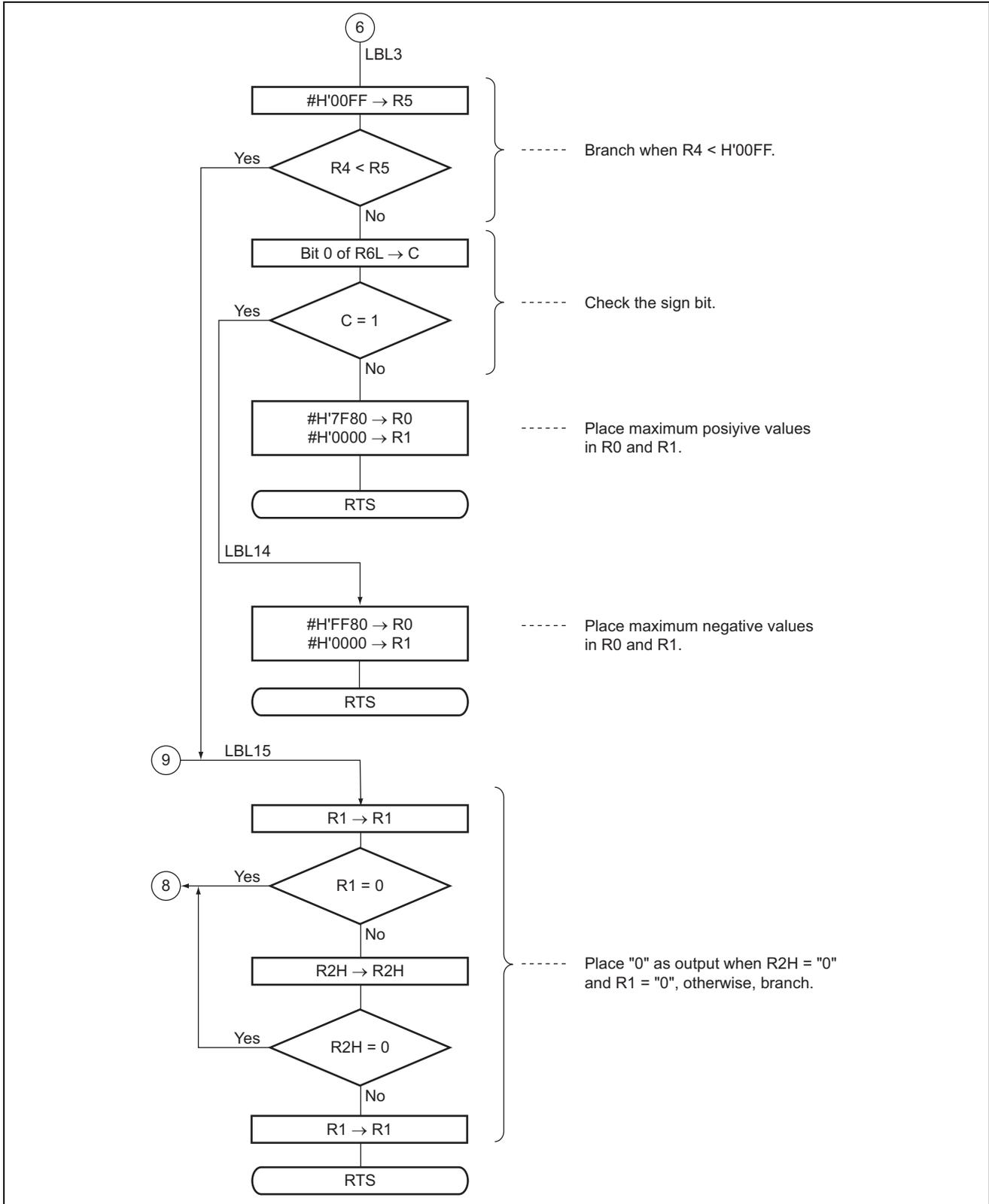


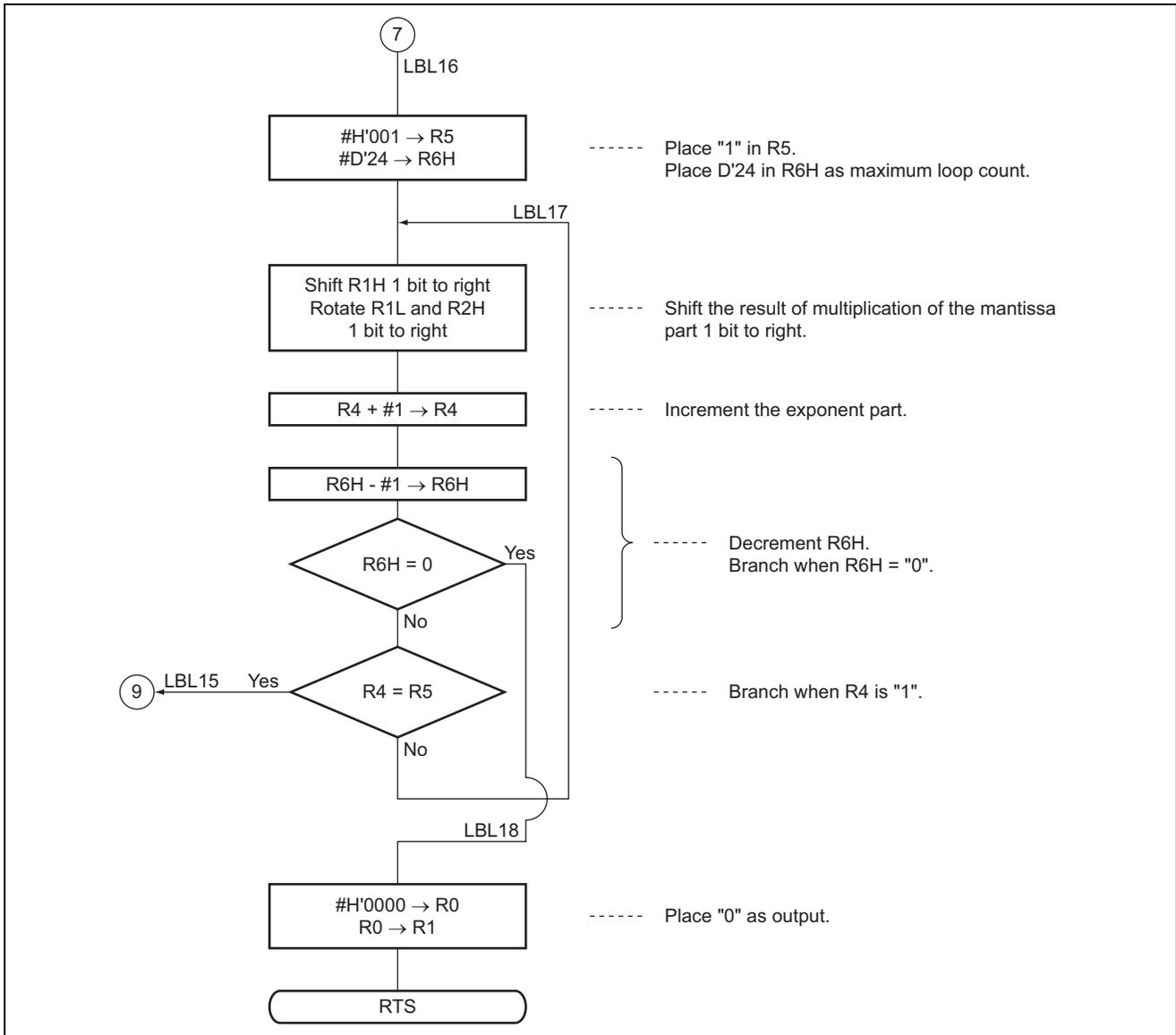


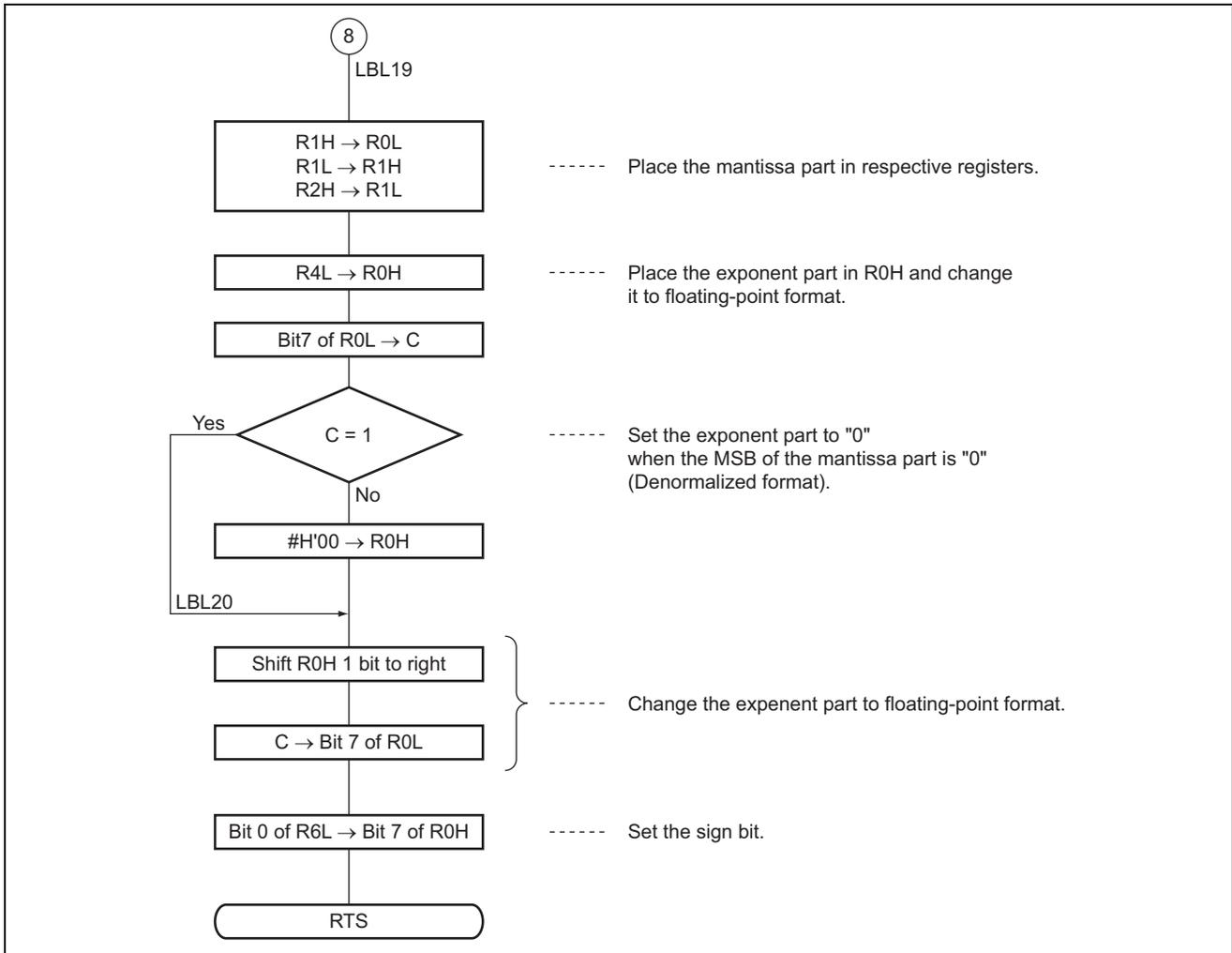


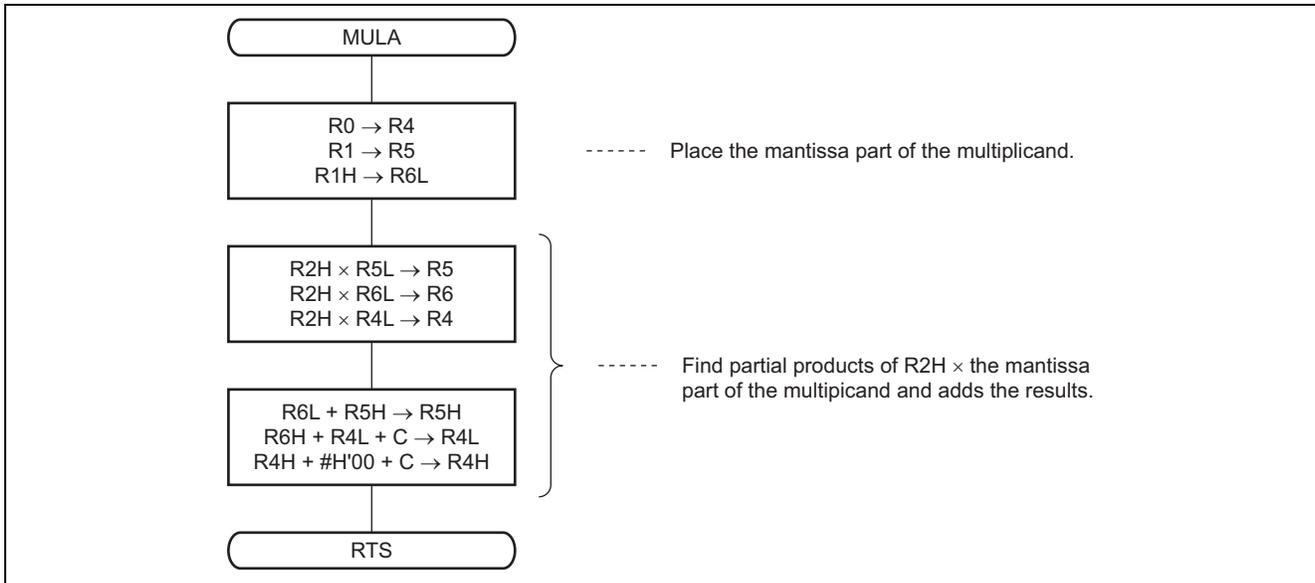












## 7. Program List

```

*** H8/300 ASSEMBLER VER 1.0B ** 08/18/92 10:22:23
PROGRAM NAME =
1          ;*****
2          ;*
3          ;*      00 - NAME :FLOATING POINT MULTIPLICATION (FMUL)
4          ;*
5          ;*****
6          ;*
7          ;*      ENTRY      :R0 (UPPER WORD OF MULTI PLICAND)
8          ;*
9          ;*      R1 (LOWER WORD OF MULTI PLICAND)
10         ;*      R2 (UPPER WORD OF MULTIPLIER)
11         ;*      R3 (LOWER WORD OF MULTIPLIER)
12         ;*
13         ;*      RETURNS   :R0 (UPPER WORD OF RESULT)
14         ;*      R1 (LOWER WORD OF RESULT)
15         ;*****
16         ;
17 FMUL_cod C 0000          .SECTION          FMUL_code, CODE, ALIGN=2
18         .EXPORT FMUL
19         ;
20 FMUL_cod C      00000000 FMUL .EQU $          ;Entry point
21 FMUL_cod C 0000 FE00      MOV.B   #H'00,R6L   ;Clear R6L
22 FMUL_cod C 0002 79057F80  MOV.W   #H'7F80,R5   ;Set "H'7F80"
23         ;
24 FMUL_cod C 0006 7770      BLD     #7,R0H     ;Set sign bit of multiplicand
25 FMUL_cod C 0008 670E      BST     #0,R6L     ; to bit 0 of R6L
26 FMUL_cod C 000A 7270      BCLR   #7,R0H     ;Bit clear bit 7 of R0H
27         ;
28 FMUL_cod C 000C 7772      BLD     #7,R2H     ;
29 FMUL_cod C 000E 750E      BXOR   #0,R6L     ;Set sign bit of result
30 FMUL_cod C 0010 670E      BST     #0,R6L     ; to bit 0 of R6L
31 FMUL_cod C 0012 7272      BCLR   #7,R2H     ;Bit clear bit 7 of R2H
32         ;
33 FMUL_cod C 0014 0D11      MOV.W   R1,R1
34 FMUL_cod C 0016 4604      BNE     LBL1
35 FMUL_cod C 0018 0D00      MOV.W   R0,R0
36 FMUL_cod C 001A 4708      BEQ     LBL2          ;Branch if R1=R0=0
37 FMUL_cod C 001C          LBL1
38 FMUL_cod C 001C 0D33      MOV.W   R3,R3
39 FMUL_cod C 001E 460C      BNE     LBL3          ;Branch if not R3=0
40 FMUL_cod C 0020 0D22      MOV.W   R2,R2
41 FMUL_cod C 0022 4608      BNE     LBL3          ;Branch if not R2=0
42         ;
43 FMUL_cod C 0024          LBL2
44 FMUL_cod C 0024 79000000  MOV.W   #H'0000,R0   ;Set 0 to result
45 FMUL_cod C 0028 0D01      MOV.W   R0,R1
46 FMUL_cod C 002A 5470      RTS
47         ;

```

```

48 FMUL_cod C 002C          LBL3
49 FMUL_cod C 002C 1D05          CMP.W   R0,R5
50 FMUL_cod C 002E 4304          BLS     LBL4          ;Branch if R0>=R5
51 FMUL_cod C 0030 1D25          CMP.W   R2,R5
52 FMUL_cod C 0032 4218          BHI     LBL7          ;Branch if R2>=R5
53 FMUL_cod C 0034          LBL4
54 FMUL_cod C 0034 770E          BLD     #0,R6L        ;Load sign bit
55 FMUL_cod C 0036 450A          BCS     LBL6          ;Branch if C = 1
56 FMUL_cod C 0038          LBL5
57 FMUL_cod C 0038 79007F80      MOV.W   #H'7F80,R0    ;Set #H'7F800000 to result
58 FMUL_cod C 003C 79010000      MOV.W   #H'0000,R1
59 FMUL_cod C 0040 5470          RTS
60 FMUL_cod C 0042          LBL6
61 FMUL_cod C 0042 7900FF80      MOV.W   #H'FF80,R0    ;Set #H'FF800000 to result
62 FMUL_cod C 0046 79010000      MOV.W   #H'0000,R1
63 FMUL_cod C 004A 5470          RTS
64                               ;
65 FMUL_cod C 004C          LBL7
66 FMUL_cod C 004C 7778          BLD     #7,R0L        ;
67 FMUL_cod C 004E 1200          ROTXL   R0H           ;
68 FMUL_cod C 0050 0C0C          MOV.B   R0H,R4L       ;Set exponent of multiplicand to R4
69 FMUL_cod C 0052 F400          MOV.B   #H'00,R4H
70                               ;
71 FMUL_cod C 0054 777A          BLD     #7,R2L        ;
72 FMUL_cod C 0056 1202          ROTXL   R2H           ;
73 FMUL_cod C 0058 0C2D          MOV.B   R2H,R5L       ;Set exponent of multiplier to R5
74 FMUL_cod C 005A F500          MOV.B   #H'00,R5H
75                               ;
76 FMUL_cod C 005C 7278          BCLR    #7,R0L        ;Clear bit 7 of R0L
77 FMUL_cod C 005E 0C00          MOV.B   R0H,R0H
78 FMUL_cod C 0060 4704          BEQ     LBL8          ;Branch if multiplicand is
                               ;denormalized
79 FMUL_cod C 0062 7078          BSET    #7,R0L        ;Set implicit MSB
80 FMUL_cod C 0064 4002          BRA     LBL9          ;Branch always
81 FMUL_cod C 0066          LBL8
82 FMUL_cod C 0066 0B04          ADDS.W #1,R4
83                               ;
84 FMUL_cod C 0068          LBL9
85 FMUL_cod C 0068 727A          BCLR    #7,R2L        ;Clear bit 7 of R2L
86 FMUL_cod C 006A 0C22          MOV.B   R2H,R2H
87 FMUL_cod C 006C 4704          BEQ     LBL10         ;Branch if multiplier is
                               ;denormalized
88 FMUL_cod C 006E 707A          BSET    #7,R2L        ;Set implicit MSB
89 FMUL_cod C 0070 4002          BRA     LBL11         ;Branch always
90 FMUL_cod C 0072          LBL10
91 FMUL_cod C 0072 0B05          ADDS.W #1,R5
92                               ;
93 FMUL_cod C 0074          LBL11
94 FMUL_cod C 0074 0954          ADD.W   R5,R4         ;addition exponents
95 FMUL_cod C 0076 06FE          ANDC   #H'FE,CCR      ;Clear C flag of CCR
96 FMUL_cod C 0078 BC7F          SUBX.B #H'7F,R4L      ;R4L - #H'7F - C -> R4L
97 FMUL_cod C 007A B400          SUBX.B #H'00,R4H

```

```

98                                     ;
99  FMUL_cod C 007C 6DF4              PUSH  R4              ;Push R4
100 FMUL_cod C 007E 6DF6              PUSH  R6              ;Push R6
101                                     ;
102  FMUL_cod C 0080 0D04              MOV.W  R0,R4          ;
103  FMUL_cod C 0082 0D15              MOV.W  R1,R5
104                                     ;
105  FMUL_cod C 0084 0CA2              MOV.B  R2L,R2H
106  FMUL_cod C 0086 5E000000         JSR    @MULA          ;R2L * (R0L:R1) -> (R4:R5)
107  FMUL_cod C 008A 6DF4              PUSH  R4              ;Push R4
108  FMUL_cod C 008C 6DF5              PUSH  R5              ;Push R5
109                                     ;
110  FMUL_cod C 008E 0C32              MOV.B  R3H,R2H
111  FMUL_cod C 0090 5E000000         JSR    @MULA          ;R3L * (R0L:R1) -> (R4:R5)
112  FMUL_cod C 0094 6DF4              PUSH  R4              ;Push R4
113  FMUL_cod C 0096 6DF5              PUSH  R5              ;Push R5
114                                     ;
115  FMUL_cod C 0098 0CB2              MOV.B  R3L,R2H          ;
116  FMUL_cod C 009A 5E000000         JSR    @MULA          ;R3L * (R0L:R1) -> (R4:R5)
117  FMUL_cod C 009E 0D42              MOV.W  R4,R2          ;Push R4
118  FMUL_cod C 00A0 0D53              MOV.W  R5,R3          ;Push R5
119                                     ;
120  FMUL_cod C 00A2 79010000         MOV.W  #H'0000,R1    ;Clear R1
121  FMUL_cod C 00A6 6D75              POP    R5              ;Pop R5
122  FMUL_cod C 00A8 6D74              POP    R4              ;Pop R4
123                                     ;
124  FMUL_cod C 00AA 08D3              ADD.B  R5L,R3H        ;R3H + R5L -> R3H
125  FMUL_cod C 00AC 0E5A              ADDX.B R5H,R2L        ;R2L + R5H + C -> R2L
126  FMUL_cod C 00AE 0EC2              ADDX.B R4L,R2H        ;R2H + R4L + C -> R2H
127  FMUL_cod C 00B0 0E49              ADDX.B R4H,R1L        ;R1L + R4H + C -> R1L
128                                     ;
129  FMUL_cod C 00B2 6D75              POP    R5              ;Pop R5
130  FMUL_cod C 00B4 6D74              POP    R4              ;Pop R4
131  FMUL_cod C 00B6 0952              ADD.W  R5,R2          ;R2 + R5 -> R2
132  FMUL_cod C 00B8 0EC9              ADDX.B R4L,R1L        ;R1L + R4L + C -> R1L
133  FMUL_cod C 00BA 0E41              ADDX.B R4H,R1H        ;R1H + R4H + C -> R1H
134                                     ;
135  FMUL_cod C 00BC 6D76              POP    R6              ;Pop R6
136  FMUL_cod C 00BE 6D74              POP    R4              ;Pop R4
137  FMUL_cod C 00C0 0B04              ADDS.W #1,R4
138  FMUL_cod C 00C2 0D44              MOV.W  R4,R4
139                                     ;
140  FMUL_cod C 00C4 474A              BEQ    LBL16          ;Branch if R4=0
141  FMUL_cod C 00C6 4B48              BMI    LBL16          ;Branch if R4<0
142  FMUL_cod C 00C8                   LBL12
143  FMUL_cod C 00C8 1B04              SUBS.W #1,R4
144  FMUL_cod C 00CA 0D44              MOV.W  R4,R4
145  FMUL_cod C 00CC 4714              BEQ    LBL13          ;Branch if R4=0
146  FMUL_cod C 00CE 100B              SHLL  R3L            ;Shift mantissa 1 bit left
147  FMUL_cod C 00D0 1203              ROTXL  R3H
148  FMUL_cod C 00D2 120A              ROTXL  R2L
149  FMUL_cod C 00D4 1202              ROTXL  R2H
150  FMUL_cod C 00D6 1209              ROTXL  R1L
151  FMUL_cod C 00D8 1201              ROTXL  R1H

```

```

152 FMUL_cod C 00DA 44EC          BCC    LBL12          ;Branch if C = 0
153 FMUL_cod C 00DC 1301          ROTXR  R1H            ;Rotate mantissa 1 bit right
154 FMUL_cod C 00DE 1309          ROTXR  R1L
155 FMUL_cod C 00E0 1302          ROTXR  R2H
156 FMUL_cod C 00E2                LBL13
157 FMUL_cod C 00E2 0B04          ADDS.W #1,R4
158                               ;
159 FMUL_cod C 00E4 790500FF      MOV.W  #H'00FF,R5    ;
160 FMUL_cod C 00E8 1D45          CMP.W  R4,R5
161 FMUL_cod C 00EA 4418          BCC    LBL15          ;Branch if R5>R4
162 FMUL_cod C 00EC 770E          BLD    #0,R6L        ;Load sign bit
163 FMUL_cod C 00EE 450A          BCS    LBL14          ;Branch if C = 1
164 FMUL_cod C 00F0 79007F80      MOV.W  #H'7F80,R0    ;Set H'7F800000 to result
165 FMUL_cod C 00F4 79010000      MOV.W  #H'0000,R1
166 FMUL_cod C 00F8 5470          RTS
167                               ;
168 FMUL_cod C 00FA                LBL14
169 FMUL_cod C 00FA 7900FF80      MOV.W  #H'FF80,R0    ;Set H'FF800000 to product
170 FMUL_cod C 00FE 79010000      MOV.W  #H'0000,R1
171 FMUL_cod C 0102 5470          RTS
172                               ;
173 FMUL_cod C 0104                LBL15
174 FMUL_cod C 0104 0D11          MOV.W  R1,R1
175 FMUL_cod C 0106 4628          BNE    LBL19          ;Branch if not R1=0
176 FMUL_cod C 0108 0C22          MOV.B  R2H,R2H
177 FMUL_cod C 010A 4624          BNE    LBL19          ;Branch if not R2H=0
178 FMUL_cod C 010C 0D10          MOV.W  R1,R0
179 FMUL_cod C 010E 5470          RTS
180                               ;
181 FMUL_cod C 0110                LBL16
182 FMUL_cod C 0110 79050001      MOV.W  #H'0001,R5    ;Set #H'0001 to R5
183 FMUL_cod C 0114 F618          MOV.B  #D'24,R6H     ;Se bit counter
184 FMUL_cod C 0116                LBL17
185 FMUL_cod C 0116 1101          SHLR   R1H            ;Shift mantissa 1 bit right
186 FMUL_cod C 0118 1309          ROTXR  R1L
187 FMUL_cod C 011A 1302          ROTXR  R2H
188 FMUL_cod C 011C 0B04          ADDS.W #1,R4          ;Increment exponent
189 FMUL_cod C 011E 1A06          DEC.B  R6H            ;Decrement bit counter
190 FMUL_cod C 0120 4706          BEQ    LBL18          ;Branch if Z=1
191 FMUL_cod C 0122 1D54          CMP.W  R5,R4
192 FMUL_cod C 0124 47DE          BEQ    LBL15          ;Branch if R5=R4
193 FMUL_cod C 0126 40EE          BRA    LBL17          ;Branch always
194 FMUL_cod C 0128                LBL18
195 FMUL_cod C 0128 79000000      MOV.W  #H'0000,R0    ;Clear result
196 FMUL_cod C 012C 0D01          MOV.W  R0,R1
197 FMUL_cod C 012E 5470          RTS
198                               ;
199 FMUL_cod C 0130                LBL19
200 FMUL_cod C 0130 0C18          MOV.B  R1H,R0L
201 FMUL_cod C 0132 0C91          MOV.B  R1L,R1H
202 FMUL_cod C 0134 0C29          MOV.B  R2H,R1L

```

```

203                                     ;
204 FMUL_cod C 0136 0CC0                MOV.B   R4L,R0H
205 FMUL_cod C 0138 7778                BLD    #7,R0L
206 FMUL_cod C 013A 4502                BCS    LBL20                ;Branch if C = 1
207 FMUL_cod C 013C F000                MOV.B   #H'00,R0H
208 FMUL_cod C 013E                    LBL20                ;Change floating point format
209 FMUL_cod C 013E 1100                SHLR   R0H
210 FMUL_cod C 0140 6778                BST    #7,R0L
211 FMUL_cod C 0142 770E                BLD    #0,R6L
212 FMUL_cod C 0144 6770                BST    #7,R0H
213 FMUL_cod C 0146 5470                RTS
214                                     ;
215                                     ;-----
216                                     ;
217 FMUL_cod C 0148                    MULA                ;R2H * (R0L:R1) -> (R4:R5)
218 FMUL_cod C 0148 0D04                MOV.W   R0,R4                ;R0 -> R4
219 FMUL_cod C 014A 0D15                MOV.W   R1,R5                ;R1 -> R5
220 FMUL_cod C 014C 0C1E                MOV.B   R1H,R6L                ;R1H -> R6L
221                                     ;
222 FMUL_cod C 014E 5025                MULXU  R2H,R5                ;R2H * R5L -> R5
223 FMUL_cod C 0150 5026                MULXU  R2H,R6                ;R2H * R6L -> R6
224 FMUL_cod C 0152 5024                MULXU  R2H,R4                ;R2H * R4L -> R4
225                                     ;
226 FMUL_cod C 0154 08E5                ADD.B   R6L,R5H                ;R5H + R6L -> R5H
227 FMUL_cod C 0156 0E6C                ADDX.B  R6H,R4L                ;R4L + R6H + C -> R4L
228 FMUL_cod C 0158 9400                ADDX.B  #H'00,R4H                ;R4H + #H'00 + C -> R4H
*** H8/300 ASSEMBLER VER 1.0B ** 08/18/92 10:22:23 PAGE
5
PROGRAM NAME =
229 FMUL_cod C 015A 5470                RTS
230                                     ;
231                                     .END
*****TOTAL ERRORS 0
*****TOTAL WARNINGS 0

```

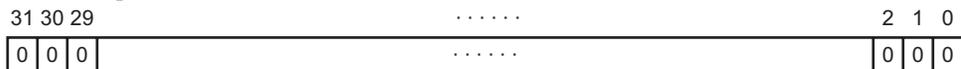
## About Single-Precision Floating-Point Numbers <Reference>

### Single-Precision Floating-Point Formats:

1. Internal representation of single-precision floating-point numbers

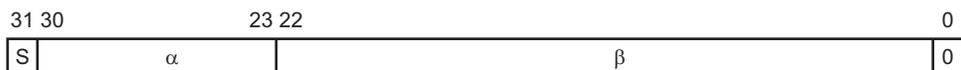
In this Application Note, the following formats are applied to single-precision floating-point numbers depending on their values (R = real number):

A. Internal representation for R = 0



All of the 32 bits are 0's.

B. Normalized format



$\alpha$  is an exponent whose field is 8 bits long.  $\beta$  is a mantissa whose field is 23 bits long. The value of R can be represented by the following equation (on conditions that  $1 \leq \alpha \leq 254$ ):

$$R = 2^S \times 2^{\alpha-127} \times (1 + 2^{-1} \times \beta_{22} + 2^{-2} \times \beta_{21} + \dots + 2^{-23} \times \beta_0)$$

where  $\beta_i$  is the value of the i-th bit ( $0 \leq i \leq 22$ ) and S is the sign bit.

C. Denormalized format



where  $\beta$  is a mantissa whose field is 23 bits long. This format is used to represent a real number too small to be represented in the normal format. In this format, R can be represented by the following equation:

$$R = 2^S \times 2^{-126} \times (2^{-1} \times \beta_{22} + 2^{-2} \times \beta_{21} + \dots + 2^{-23} \times \beta_0)$$

D. Infinity



where  $\beta$  is a mantissa whose field is 23 bits long. In this Application Note, however, the following rules apply if all exponents are 1's;

Positive infinity when S = 0

$$R = +\infty$$

Negative infinity when S = 1

$$R = -\infty$$

2. Example of internal representation

If  $S = B'0$  (binary)  
 $\alpha = B'10000011$  (binary)  
 $\beta = B'1011100\dots\dots 0$  (binary)

Then the corresponding real number is as follows:

$$R = 2^0 \times 2^{131-127} \times (1 + 2^{-1} + 2^{-3} + 2^{-4} + 2^{-5})$$

$$= 16 + 8 + 2 + 1 + 0.5 = 27.5$$

A. Maximum and minimum values

The maximum value ( $R_{MAX}$ ) and minimum value ( $R_{MIN}$ ), in terms of the absolute value, are as follows:

$$R_{MAX} = 2^{254 - 127} \times (1 + 2^{-1} + 2^{-2} + 2^{-3} \dots\dots + 2^{-23})$$

$$= 3.37 \times 10^{38}$$

$$R_{MIN} = 2^{-126} \times 2^{-23} = 2^{-140} = 1.40 \times 10^{-45}$$

The absolute values within the above range can be represented.

## Website and Support

Renesas Technology Website

<http://www.renesas.com/>

Inquiries

<http://www.renesas.com/inquiry>

[csc@renesas.com](mailto:csc@renesas.com)

## Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Sep.18.03	—	First edition issued
2.00	Nov.30.06	All pages	Content correction

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
  - (1) artificial life support devices or systems
  - (2) surgical implantations
  - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
  - (4) any other purposes that pose a direct threat to human life
 Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.