SH7000 Series

Quotient of 32 Bit ÷ 32 Bit (Signed)

Label: DIVS32Q

Functions Used: DIV0S Instruction
DIV1 Instruction

Contents

1. Function

Divides the dividend (signed 32 bits) by the divisor (signed 32 bits), and determines the quotient (signed 32 bits). Also indicates errors (division by 0) in the T bit.

2. Arguments

Description		Storage Location	Data Length (Bytes)
Input	Dividend (signed 32 bits)	R1	4
	Divisor (signed 32 bits)	R0	4
Output	Quotient (signed 32 bits)	R1	4
	Error (division by 0) generated/not generated (generated: T = 1, not generated: T = 0)	T bit (SR)	4

3. Internal Register Changes and Flag Changes

	(Before Execution) → (After Execution)
R0	Divisor (signed 32 bits) → No change
R1	Dividend (signed 32 bits) → Quotient (signed 32 bits)
R2	Work
R3	Work
R4	
R5	
R6	
R7	
R8	
R9	
R10	
R11	
R12	
R13	
R14	
R15	(SP)

T bit | * |

— : No change
* : Change
0 : Fixed 0
1 : Fixed 1

4. Programming Specifications

Program memory (bytes)
166
Data memory (bytes)
0
Stack (bytes)
8
Number of states
80
Reentrant
Yes
Relocation
Yes
Intermediate interrupt
Yes

5. Notes

The number of states indicated in the programming specifications is the value when H'80000000 ÷ H'7FFFFFFF is calculated.

6. Description

(1) Function

Details of the arguments are as follows.

R0: Set the divisor (signed 32 bits) as the input argument.

R1: Set the dividend (signed 32 bits) as the input argument.
Holds the quotient (signed 32 bits) as the output argument.

T bit (SR): Indicates whether an error (division by 0) has occurred.
T bit = 1: Indicates an error (division by 0) has occurred.
T bit = 0: Indicates no error (division by 0) has occurred.

Figure 1 shows a software DIVS32Q execution example.

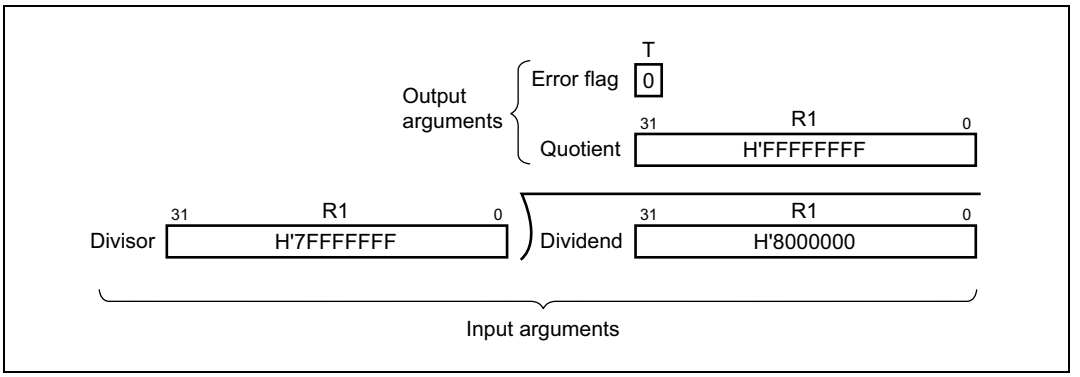


Figure 1 Software DIVS32Q Execution Example

(2) Usage Notes

After execution of software instruction DIVS32Q, the quotient is set in R1, which previously contained the dividend, and the dividend is destroyed. If the value for the dividend will be needed after the software DIVS32Q instruction is executed, it should be saved beforehand.

In addition, although H'80000000 ÷ H'FFFFFFFF results in an overflow, this overflow is not detected by software instruction DIVS32Q.

(3) RAM Used

No RAM is used by the software DIVS32Q instruction.

(4) Usage Example

After the dividend and divisor are set in the input arguments, the software instruction DIVS32Q is executed by a subroutine call.

```
        MOV.L   DATA1,R1     ... Sets dividend (signed 32 bits) in input argument (R1)
        BSR     DIVS32Q      ... Subroutine call to software instruction DIVS32Q
        MOV.L   DATA2,R0     ... Sets divisor (signed 32 bits) in input argument (R0)
        BT      ERROR        ... Branches to error processing subroutine if error (division by 0) occurs
              .
              .
              .
              .
      .align    4
DATA1 .data.l   H'80000000
DATA2 .data.l   H'7FFFFFFF
```

(5) Operating Principle

(a) Before division, the following initial settings are carried out.

(i) R2 is used for the upper 32 bits to sign extend the dividend to 64 bits.
(Figure 2-(1))

(ii) If the dividend is negative, it is converted to a complement of 1 for handling by the one-step division instruction.
(Figure 2-(2))

(iii) The M, Q, and T bits used in one-step division are set to signed division values (M = divisor sign, Q = dividend sign, T = quotient sign).
(Figure 2-(3))

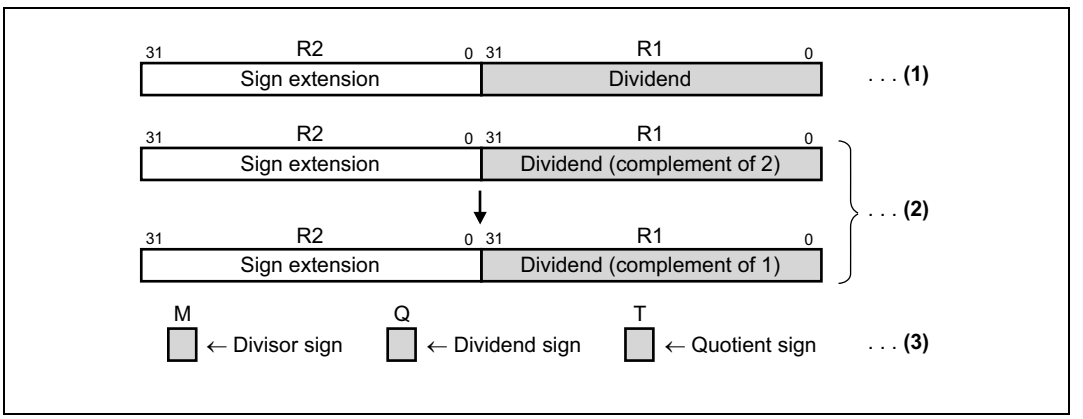


Figure 2 Initial Settings

(b) As shown in figure 3, the division operation is repeated through the number of divisor bits (32 times) using the ROCTL and DIV1 instructions.

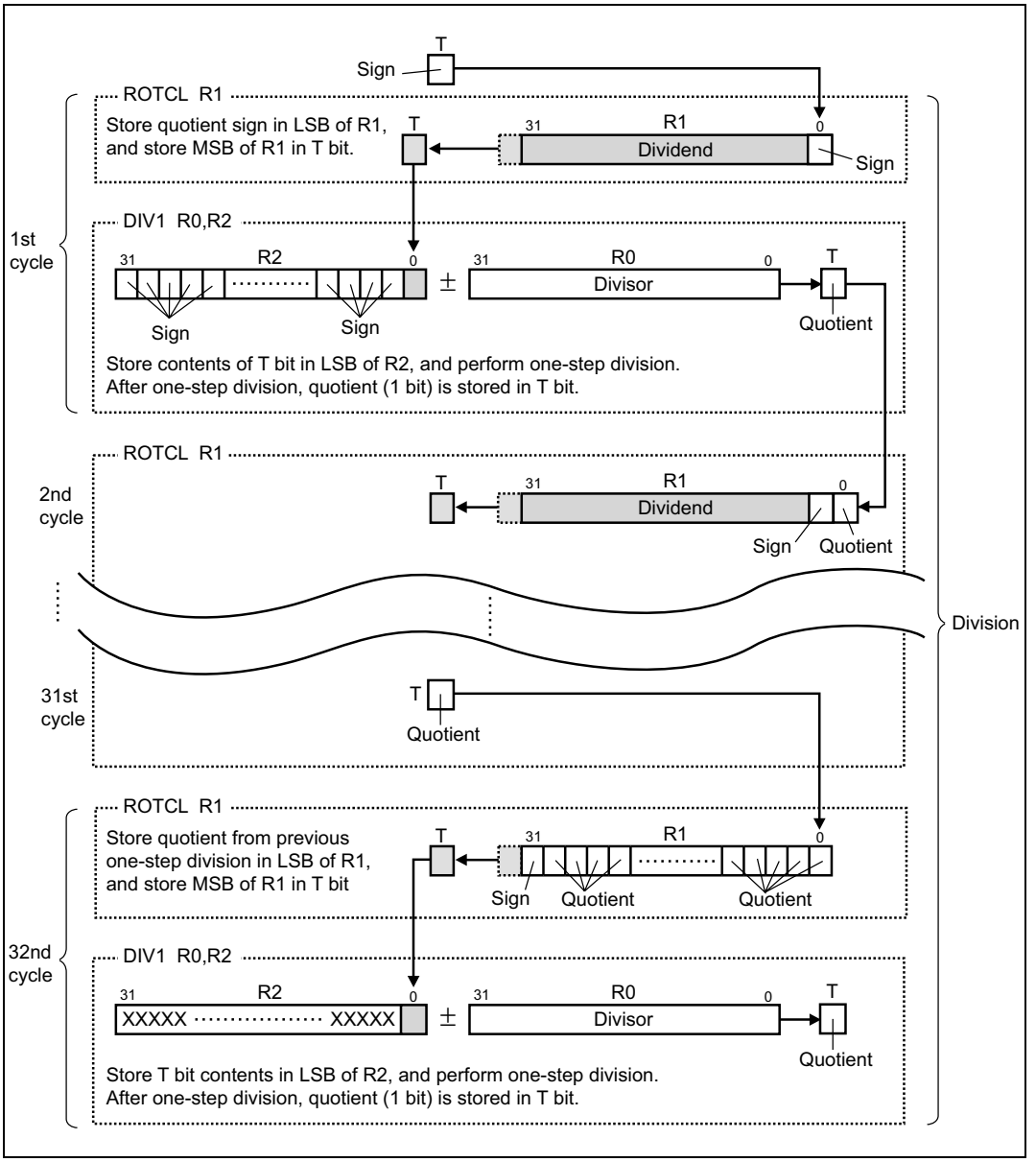


Figure 3 Operation Example

(c) - (i) As shown in figure 4, the 32nd quotient of one-step division is stored in the T bit, and the quotient sign in the MSB of R1, at the end of division. If the quotient is positive, it becomes the contents of R1, which stores the T bit (32nd quotient of one-step division) in the LSB. If the quotient is negative, it becomes a complement of 1 of the T bit (32nd quotient of one-step division) stored in the LSB of R1, which in turn is converted into a complement of 2.

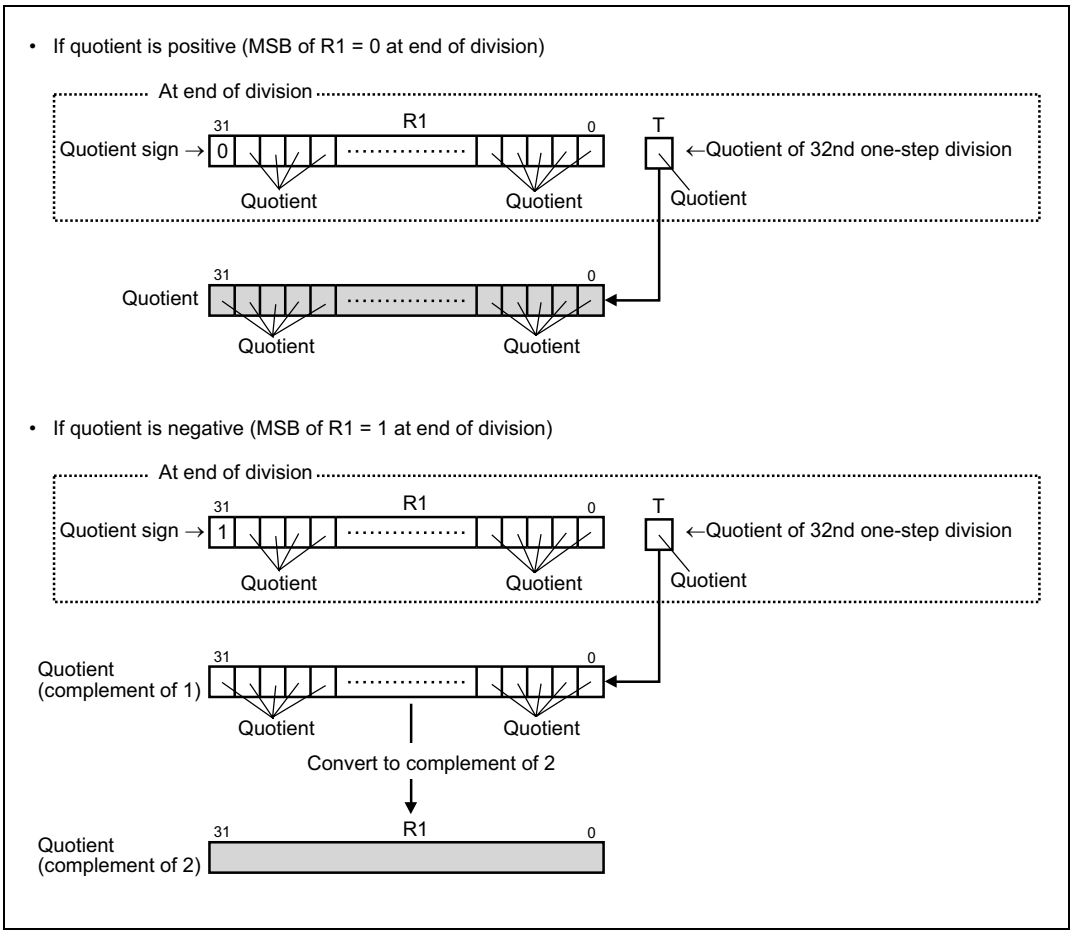


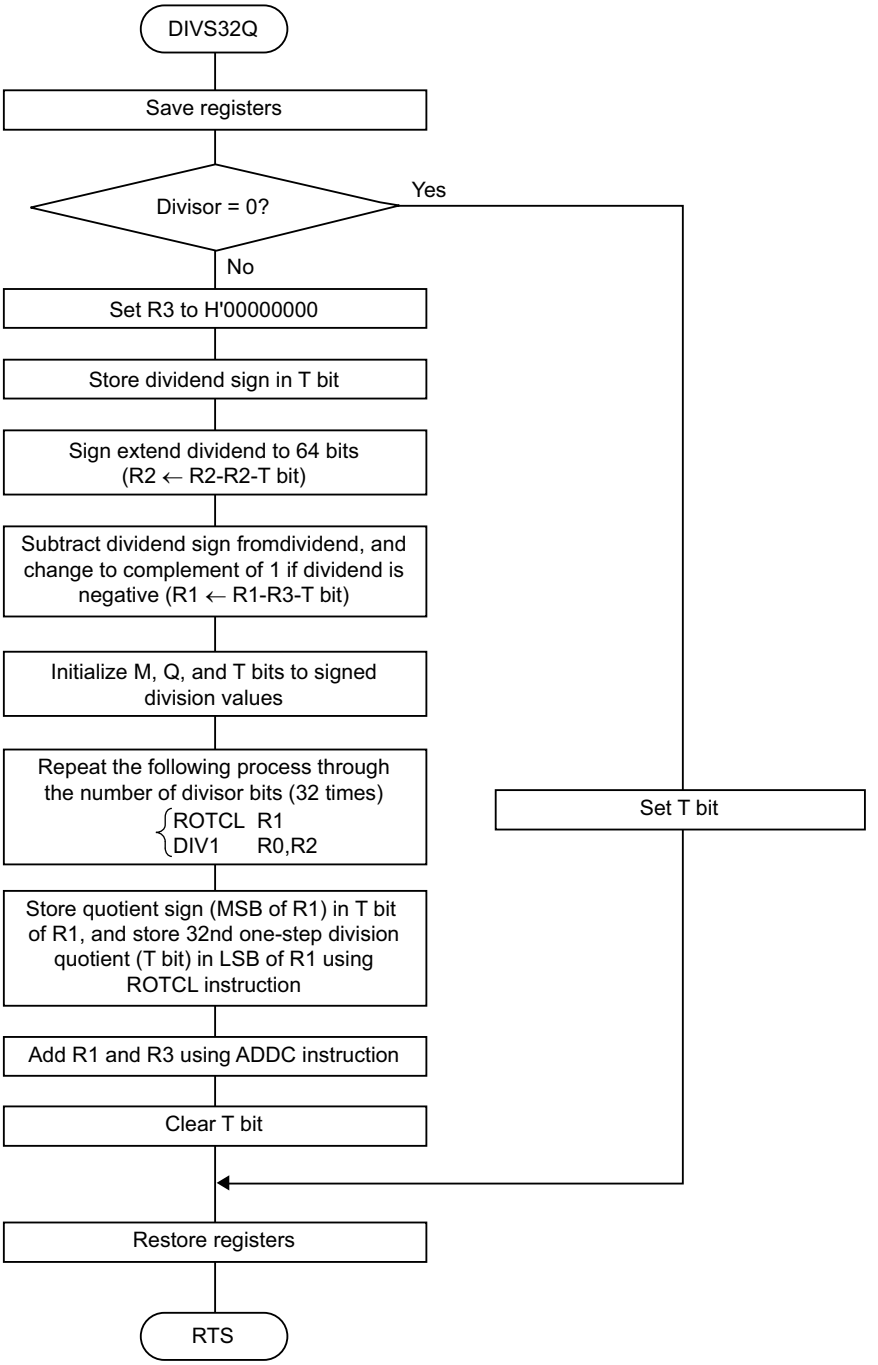
Figure 4 Quotient

(ii) The software instruction DIVS32Q performs the processing described in (i) as follows. Note that R3 stores H'00000000.

`ROTCL R1` : Stores quotient sign in T bit, and saves T bit quotient to LSB of R1.

`ADDC R3,R1` : If quotient is positive, T bit = 0, so there is no change of value. If quotient is negative, T bit = 1, so 1 is added to make it complement of 2.

7. Flowchart


DIVS32Q
Save registers
Divisor = 0?
Yes
No
Set R3 to H'00000000
Store dividend sign in T bit
Sign extend dividend to 64 bits
(R2 ← R2-R2-T bit)
Subtract dividend sign fromdividend, and
change to complement of 1 if dividend is
negative (R1 ← R1-R3-T bit)
Initialize M, Q, and T bits to signed
division values
Repeat the following process through
the number of divisor bits (32 times)
ROTCL R1
DIV1 R0,R2
Store quotient sign (MSB of R1) in T bit
of R1, and store 32nd one-step division
quotient (T bit) in LSB of R1 using
ROTCL instruction
Add R1 and R3 using ADDC instruction
Clear T bit
Set T bit
Restore registers
RTS

8. Program Listing

```
 1                          1  ;*************************************************************
 2                          2  ;*                                                           *
 3                          3  ;*      NAME ; QUOTIENT OF 32 BIT SIGNED DIVISION (DIVS32Q)  *
 4                          4  ;*                                                           *
 5                          5  ;*************************************************************
 6                          6  ;*                                                           *
 7                          7  ;*      ENTRY : R1 (DIVIDEND)                                *
 8                          8  ;*              R0 (DIVISOR)                                 *
 9                          9  ;*    RETURNS : R1 (QUOTIENT)                                *
10                         10  ;*              T BIT (ERROR -> TRUE;T=1,FALSE;T=0           *
11                         11  ;*                                                           *
12                         12  ;*************************************************************
13 00001000                13          .SECTION A,CODE,LOCATE=H'1000
14          00001000       14  DIVS32Q .EQU    $           : Entry point
15 00001000 2F26           15          MOV.L   R2,@-R15    ; Escape register
16 00001002 2F36           16          MOV.L   R3,@-R15    ;
17 00001004 2008           17          TST     R0,R0       ; Divisor = 0 ?
18 00001006 894A           18          BT      DIVS32Q1    ; Yes
19 00001008 233A           19          XOR     R3,R3       ; R3 <- H'00000000
20 0000100A 2137           20          DIV0S   R3,R1       ; T bit <- Sign of dividend
21 0000100C 322A           21          SUBC    R2,R2       ; R2 sign extend
22 0000100E 313A           22          SUBC    R3,R1       ;
23 00001010 2207           23          DIV0S   R0,R2       ; Divide as signed
24                         24                              ;
25 00001012 4124           25          ROTCL   R1          ;
26 00001014 3204           26          DIV1    R0,R2       ;
27 00001016 4124           27          ROTCL   R1          ;
28 00001018 3204           28          DIV1    R0,R2       ;
29 0000101A 4124           29          ROTCL   R1          ;
30 0000101C 3204           30          DIV1    R0,R2       ;
31 0000101E 4124           31          ROTCL   R1          ;
32 00001020 3204           32          DIV1    R0,R2       ;
33 00001022 4124           33          ROTCL   R1          ;
34 00001024 3204           34          DIV1    R0,R2       ;
35 00001026 4124           35          ROTCL   R1          ;
36 00001028 3204           36          DIV1    R0,R2       ;
37 0000102A 4124           37          ROTCL   R1          ;
38 0000102C 3204           38          DIV1    R0,R2       ;
39 0000102E 4124           39          ROTCL   R1          ;
40 00001030 3204           40          DIV1    R0,R2       ;
41                         41                              ;
42 00001032 4124           42          ROTCL   R1          ;
43 00001034 3204           43          DIV1    R0,R2       ;
44 00001036 4124           44          ROTCL   R1          ;
45 00001038 3204           45          DIV1    R0,R2       ;
46 0000103A 4124           46          ROTCL   R1          ;
47 0000103C 3204           47          DIV1    R0,R2       ;
48 0000103E 4124           48          ROTCL   R1          ;
49 00001040 3204           49          DIV1    R0,R2       ;
```

```
 50 00001042 4124          50          ROTCL   R1              ;
 51 00001044 3204          51          DIV1    R0,R2           ;
 52 00001046 4124          52          ROTCL   R1              ;
 53 00001048 3204          53          DIV1    R0,R2           ;
 54 0000104A 4124          54          ROTCL   R1              ;
 55 0000104C 3204          55          DIV1    R0,R2           ;
 56 0000104E 4124          56          ROTCL   R1              ;
 57 00001050 3204          57          DIV1    R0,R2           ;
 58                        58                                  ;
 59 00001052 4124          59          ROTCL   R1              ;
 60 00001054 3204          60          DIV1    R0,R2           ;
 61 00001056 4124          61          ROTCL   R1              ;
 62 00001058 3204          62          DIV1    R0,R2           ;
 63 0000105A 4124          63          ROTCL   R1              ;
 64 0000105C 3204          64          DIV1    R0,R2           ;
 65 0000105E 4124          65          ROTCL   R1              ;
 66 00001060 3204          66          DIV1    R0,R2           ;
 67 00001062 4124          67          ROTCL   R1              ;
 68 00001064 3204          68          DIV1    R0,R2           ;
 69 00001066 4124          69          ROTCL   R1              ;
 70 00001066 3204          70          DIV1    R0,R2           ;
 71 0000106A 4124          71          ROTCL   R1              ;
 72 0000106C 3204          72          DIV1    R0,R2           ;
 73 0000106E 4124          73          ROTCL   R1              ;
 74 00001070 3204          74          DIV1    R0,R2           ;
 75                        75                                  ;
 76 00001072 4124          76          ROTCL   R1              ;
 77 00001074 3204          77          DIV1    R0,R2           ;
 78 00001076 4124          78          ROTCL   R1              ;
 79 00001078 3204          79          DIV1    R0,R2           ;
 80 0000107A 4124          80          ROTCL   R1              ;
 81 0000107C 3204          81          DIV1    R0,R2           ;
 82 0000107E 4124          82          ROTCL   R1              ;
 83 00001080 3204          83          DIV1    R0,R2           ;
 84 00001082 4124          84          ROTCL   R1              ;
 85 00001084 3204          85          DIV1    R0,R2           ;
 86 00001086 4124          86          ROTCL   R1              ;
 87 00001088 3204          87          DIV1    R0,R2           ;
 88 0000108A 4124          88          ROTCL   R1              ;
 89 0000108C 3204          89          DIV1    R0,R2           ;
 90 0000108E 4124          90          ROTCL   R1              ;
 91 00001090 3204          91          DIV1    R0,R2           ;
 92                        92                                  ;
 93 00001092 4124          93          ROTCL   R1              ;
 94 00001094 313E          94          ADDC    R3,R1           ;
 95 00001096 0008          95          CLRT                    ; T bit <- No error
 96 00001098 63F6          96          MOV.L   @R15+,R3        ; Return register
 97 0000109A 000B          97          RTS                     ;
 98 0000109C 62F6          98          MOV.L   @R15+,R2        ;
 99 0000109E               99    DIVS32Q1                      ;
100 0000109E 0018         100          SETT                    ; T bit <- Error
```

```
101 000010A0 63F6        101          MOV.L   @R15+,R3    ; Return register
102 000010A2 000B        102          RTS                 ;
103 000010A4 62F6        103          MOV.L   @R15+,R2    ;
104                      104          .END
*****TOTAL ERRORS    0
*****TOTAL WARNINGS  0
```

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.