

SH7000 Series

Remainder of 32 Bit ÷ 32 Bit (Signed)

Label: DIVS32R

Functions Used: DIV0S Instruction
DIV1 Instruction

Contents

1. Function	2
2. Arguments.....	2
3. Internal Register Changes and Flag Changes.....	3
4. Programming Specifications	4
5. Notes	4
6. Description	5
7. Flowchart.....	10
8. Program Listing	12

1. Function

Divides the dividend (signed 32 bits) by the divisor (signed 32 bits), and determines the remainder (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	Remainder (signed 32 bits)	R2	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) → Change
R2	Undefined → Remainder (signed 32 bits)
R3	Work
R4	Work
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)
182
Data memory (bytes)
0
Stack (bytes)
8
Number of states
87
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.
- R2: Holds the remainder (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 DIVS32R execution example.

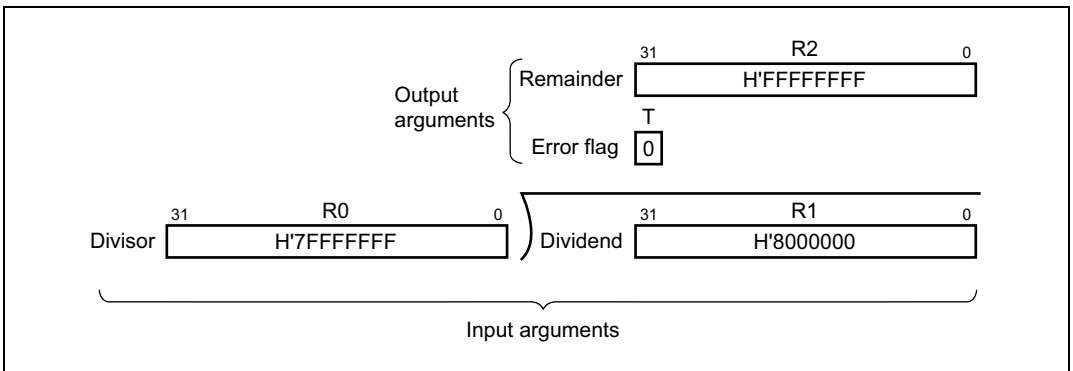


Figure 1 Software DIVS32R Execution Example

(2) Usage Notes

The value of R1, which is set to the dividend, is changed when software instruction DIVS32R is executed. If the value for the dividend will be needed after the software DIVS32R instruction is executed, it should be saved beforehand.

(3) RAM Used

No RAM is used by the software DIVS32R instruction.

(4) Usage Example

After the dividend and divisor are set in the input arguments, the software instruction DIVS32R is executed by a subroutine call.

```

MOV.L DATA1,R1    ... Sets dividend (unsigned 32 bits) in input argument (R1)
BSR   DIVS32R      ... Subroutine call to software instruction DIVS32R
MOV.L DATA2,R0    ... Sets divisor (unsigned 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 the 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))

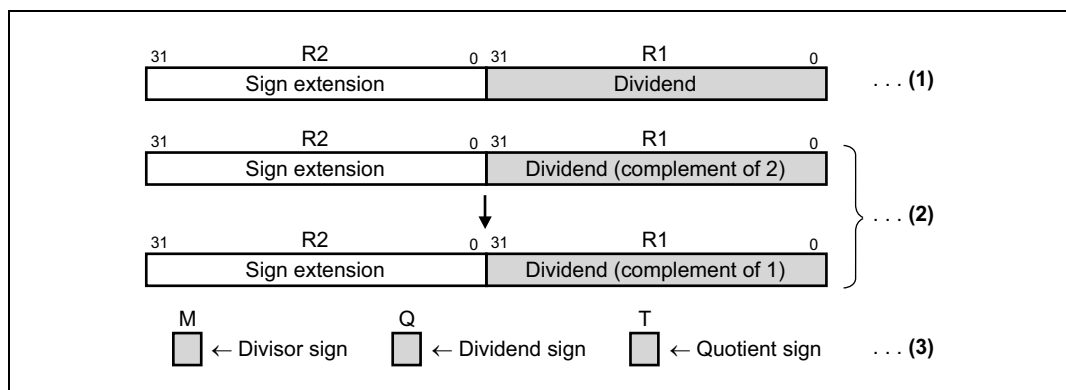


Figure 2 Initial Settings

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

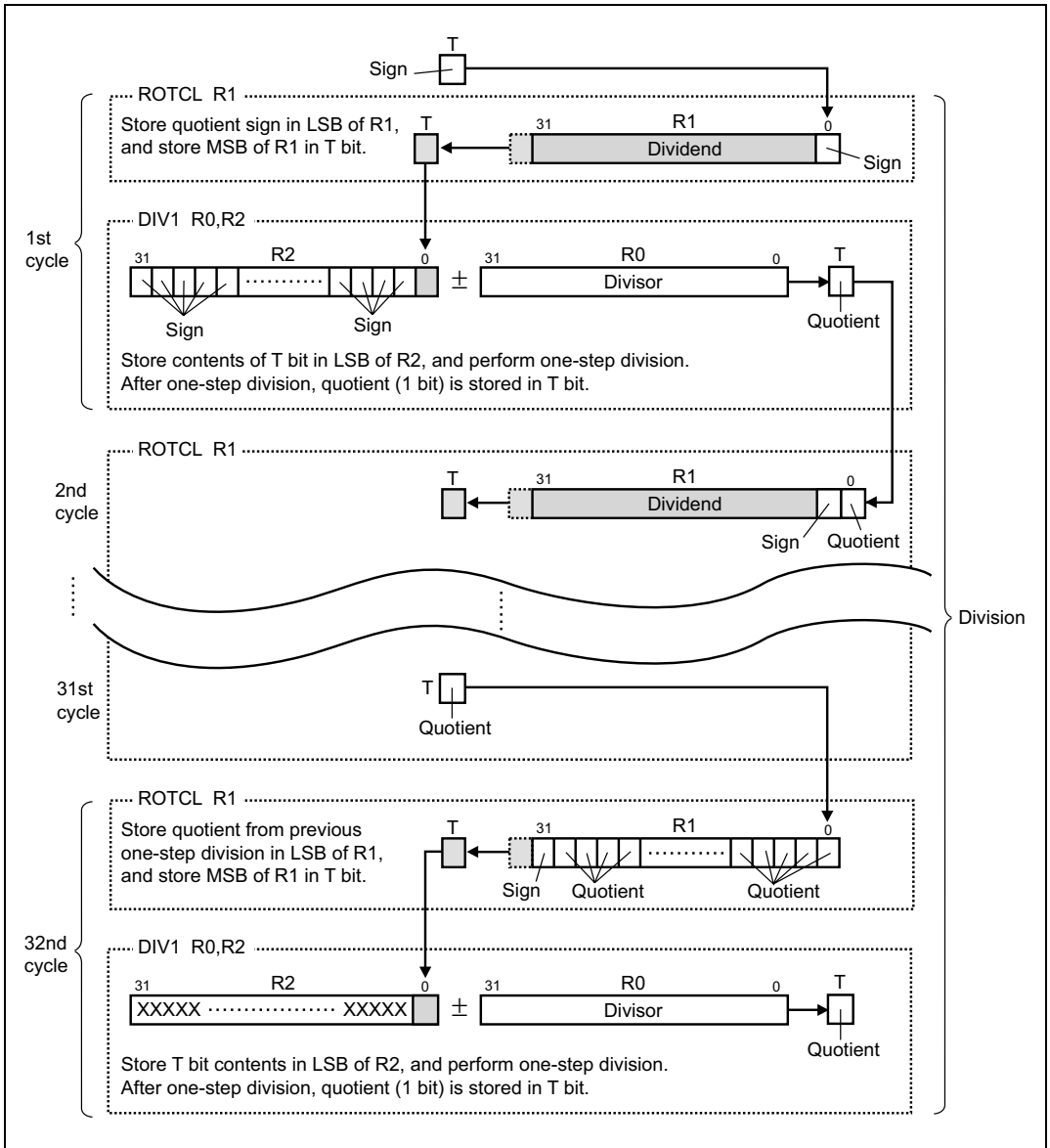


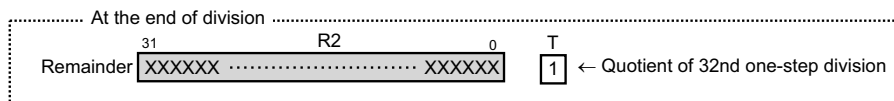
Figure 3 Operation Example

- (c) - (i) As shown in figure 4, the way of determining the remainder differs depending on the dividend sign and the value of the T bit (quotient of 32nd one-step division).

• If dividend is positive

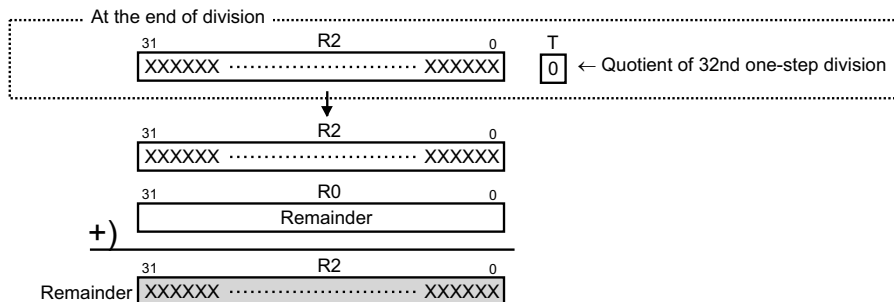
T = 0

The contents of R2 at the end of the division are the remainder.



T = 1

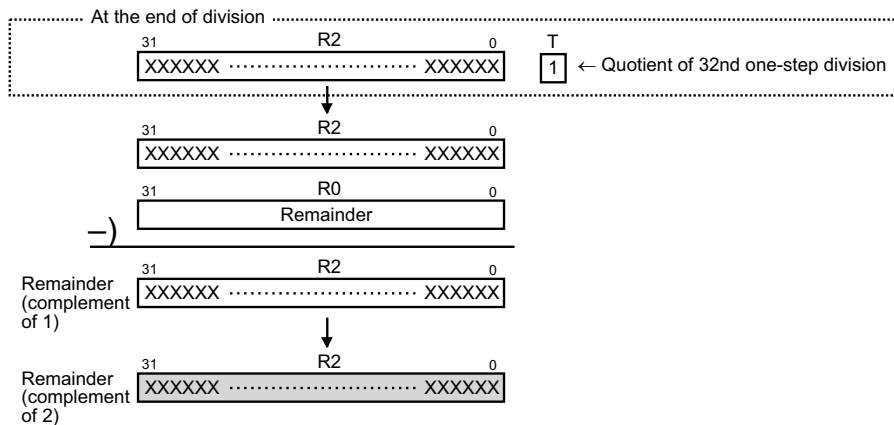
The contents of R2 at the end of division, due to internal processing of the 32nd one-step division, are a value reflecting a one-time oversubtraction of the divisor from the remainder. Therefore, the divisor is added to R2



• If dividend is negative

T = 1

The contents of R2 at the end of division, due to internal processing of the 32nd one-step division, is the remainder plus one divisor over. Therefore, the divisor is subtracted from R2. Since the content of R2 becomes a complement of 1, it must then be changed to a complement of 2.



T = 0

The contents of R2 at the end of division is a remainder that is a complement of 1, so it is changed to a complement of 2.

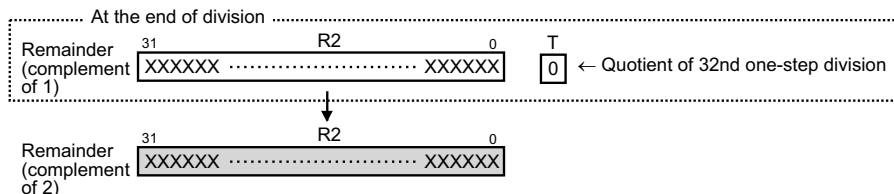


Figure 4 Remainder

(ii) The software instruction DIVS32R performs the processing described in (i) as follows.
Note that R3 stores H'00000000, and the sign bit of the dividend is stored in the LSB of R4.

- The initialization instruction (DIV0S) of unsigned division is used to store the remainder's sign bit in the T bit and to store the remainder's sign bit in the T bit to R3.

```
DIV0S  R3, R2 } Remainder sign bit → R3
MOVT   R3
```

- Since the remainder sign bit and dividend sign bit become the same sign, the dividend sign (R4) and remainder sign (R3) are exclusively ORed to determine whether the two signs match. If the signs are different, the remainder is the value with the divisor added or subtracted one time too many.

Table 1 Remainder Signs

Dividend Sign	Remainder Sign
Positive	Positive
Negative	Negative

```
XOR    R4, R3 } Dividend sign = remainder sign?
ROTCR  R3
BF     DIVS32R1
```

- The DIV1 instruction is used to correct for oversubtraction or overaddition.

```
DIV0S  R0, R2 : Initializes signed division
```

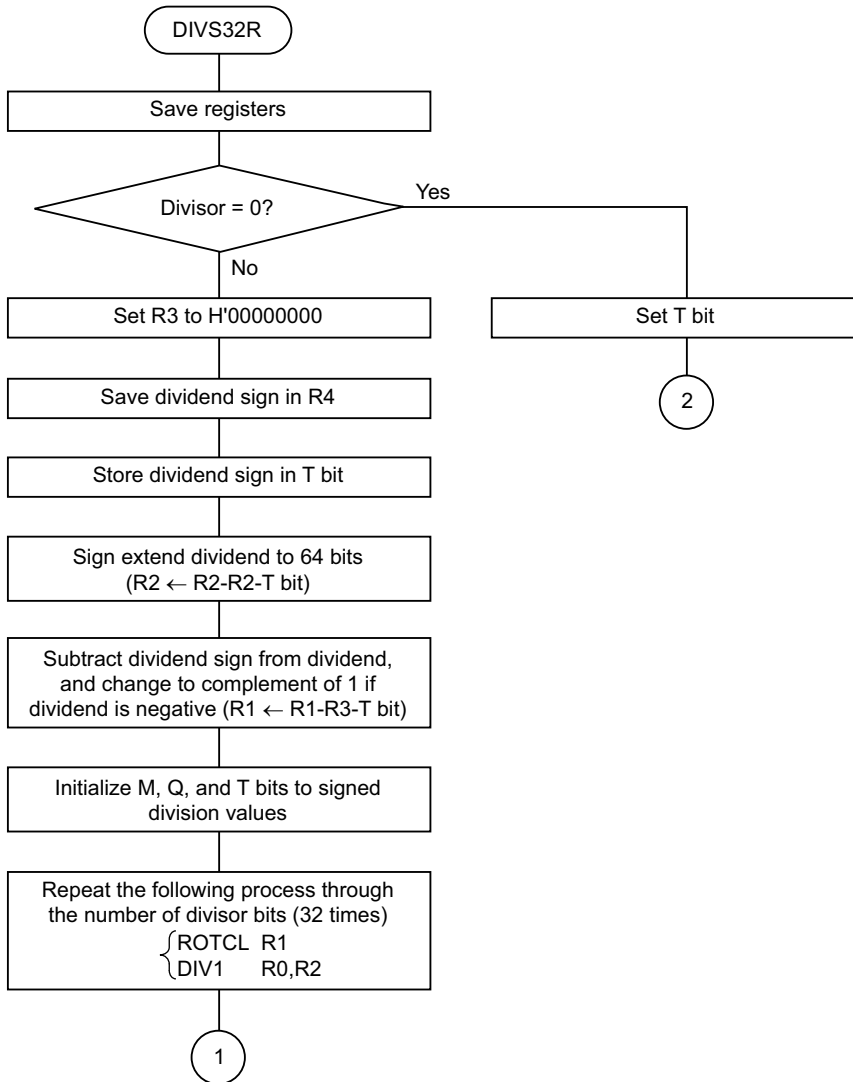
```
SHAR   R2      : Remainder (R2) is halved due to execution of next DIV1
               : instruction, so R2 is doubled
```

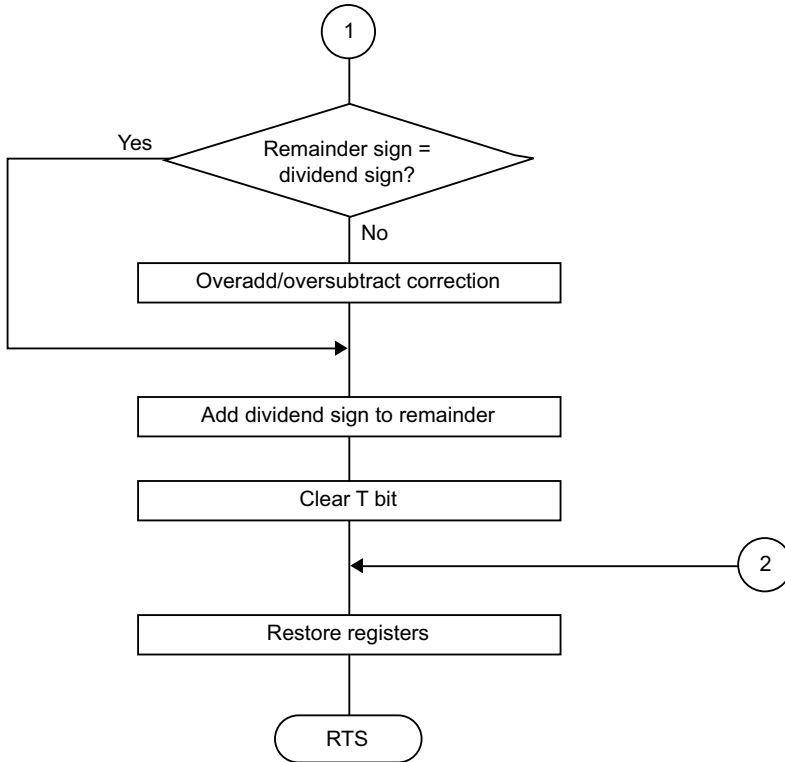
```
DIV1   R0, R2 : Remainder is added when oversubtraction occurs due to DIV1
               : internal processing; remainder is subtracted when overaddition
               : occurs
```

- R4, which contains the remainder and dividend sign bit in the LSB, is added to the remainder (R2). The remainder and dividend sign bits indicate the same sign, so if the remainder is positive, 0 is added and the remainder does not change. If the remainder is negative, 1 is added and it becomes a complement of 2.

```
ADD    R4, R2
```

7. Flowchart





8. Program Listing

```

1          1      ;*****
2          2      ;*
3          3      ;*      NAME ; RESIDUAL OF 32 BIT SIGNED DIVISION (DIVS32R)
4          4      ;*
5          5      ;*****
6          6      ;*
7          7      ;*      ENTRY : R1 (DIVIDEND)
8          8      ;*      R0 (DIVISOR)
9          9      ;*      RETURNS : R2 (RESIDUAL)
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         14     DIVS32R .EQU $      ; Entry point
15 00001000 2F36 15         MOV.L R3,@-R15 ; Escape register
16 00001002 2F46 16         MOV.L R4,@-R15 ;
17 00001004 2008 17         TST R0,R0 ; Divisor = 0?
18 00001006 8952 18         BT DIVS32R2 ; 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 0429 21         MOVT R4 ; R4 <- T bit
22 0000100E 322A 22         SUBC R2,R2 ; R2 sign extend
23 00001010 313A 23         SUBC R3,R1 ;
24 00001012 2207 24         DIV0S R0,R2 ; Divide as signed
25         25         ;
26 00001014 4124 26         ROTCL R1 ; Divide 1 step
27 00001016 3204 27         DIV1 R0,R2 ;
28 00001018 4124 28         ROTCL R1 ;
29 0000101A 3204 29         DIV1 R0,R2 ;
30 0000101C 4124 30         ROTCL R1 ;
31 0000101E 3204 31         DIV1 R0,R2 ;
32 00001020 4124 32         ROTCL R1 ;
33 00001022 3204 33         DIV1 R0,R2 ;
34 00001024 4124 34         ROTCL R1 ;
35 00001026 3204 35         DIV1 R0,R2 ;
36 00001028 4124 36         ROTCL R1 ;
37 0000102A 3204 37         DIV1 R0,R2 ;
38 0000102C 4124 38         ROTCL R1 ;
39 0000102E 3204 39         DIV1 R0,R2 ;
40 00001030 4124 40         ROTCL R1 ;
41 00001032 3204 41         DIV1 R0,R2 ;
42         42         ;
43 00001034 4124 43         ROTCL R1 ;
44 00001036 3204 44         DIV1 R0,R2 ;
45 00001036 4124 45         ROTCL R1 ;
46 0000103A 3204 46         DIV1 R0,R2 ;
47 0000103C 4124 47         ROTCL R1 ;
48 0000103E 3204 48         DIV1 R0,R2 ;
49 00001040 4124 49         ROTCL R1 ;

```

50	00001042	3204	50	DIV1	R0,R2	;
51	00001044	4124	51	ROTCL	R1	;
52	00001046	3204	52	DIV1	R0,R2	;
53	00001048	4124	53	ROTCL	R1	;
54	0000104A	3204	54	DIV1	R0,R2	;
55	0000104C	4124	55	ROTCL	R1	;
56	0000104E	3204	56	DIV1	R0,R2	;
57	00001050	4124	57	ROTCL	R1	;
58	00001052	3204	58	DIV1	R0,R2	;
59			59			;
60	00001054	4124	60	ROTCL	R1	;
61	00001056	3204	61	DIV1	R0,R2	;
62	00001058	4124	62	ROTCL	R1	;
63	0000105A	3204	63	DIV1	R0,R2	;
64	0000105C	4124	64	ROTCL	R1	;
65	0000105E	3204	65	DIV1	R0,R2	;
66	00001060	4124	66	ROTCL	R1	;
67	00001062	3204	67	DIV1	R0,R2	;
68	00001064	4124	68	ROTCL	R1	;
69	00001066	3204	69	DIV1	R0,R2	;
70	00001068	4124	70	ROTCL	R1	;
71	0000106A	3204	71	DIV1	R0,R2	;
72	0000106C	4124	72	ROTCL	R1	;
73	0000106E	3204	73	DIV1	R0,R2	;
74	00001070	4124	74	ROTCL	R1	;
75	00001072	3204	75	DIV1	R0,R2	;
76			76			;
77	00001074	4124	77	ROTCL	R1	;
78	00001076	3204	78	DIV1	R0,R2	;
79	00001078	4124	79	ROTCL	R1	;
80	0000107A	3204	80	DIV1	R0,R2	;
81	0000107C	4124	81	ROTCL	R1	;
82	0000107E	3204	82	DIV1	R0,R2	;
83	00001080	4124	83	ROTCL	R1	;
84	00001082	3204	84	DIV1	R0,R2	;
85	00001084	4124	85	ROTCL	R1	;
86	00001086	3204	86	DIV1	R0,R2	;
87	00001088	4124	87	ROTCL	R1	;
88	0000108A	3204	88	DIV1	R0,R2	;
89	0000108C	4124	89	ROTCL	R1	;
90	0000108E	3204	90	DIV1	R0,R2	;
91	00001090	4124	91	ROTCL	R1	;
92	00001092	3204	92	DIV1	R0,R2	;
93			93			;
94	00001094	2237	94	DIV0S	R3,R2	; R2 : keep sign
95	00001096	0329	95	MOVT	R3	;
96	00001098	234A	96	XOR	R4,R3	; (R4 Xor R3)--1? oversub or overadd
97	0000109A	4325	97	ROTCL	R3	;
98	0000109C	8B02	98	BF	DIVS32R1	; T bit = 0?
99	0000109E	2207	99	DIV0S	R0,R2	; Clear oversub or overadd
100	000010A0	4221	100	SHAR	R2	;

```

101 000010A2 3204      101      DIV1      R0,R2      ;
102 000010A4          102      DIVS32R1          ;
103 000010A4 324C      103      ADD       R4,R2      ;
104 000010A6 0008      104      CLRT          ; T bit <- No error
105 000010A8 64F6      105      MOV.L     @R15+,R4    ; Return register
106 000010AA 000B      106      RTS          ;
107 000010AC 63F6      107      MOV.L     @R15+,R3    ;
108 000010AE          108      DIVS32R2          ;
109 000010AE 0018      109      SETT          ; T bit <- Error
110 000010B0 64F6      110      MOV.L     @R15+,R4    ; Return register
111 000010B2 000B      111      RTS          ;
112 000010B4 63F6      112      MOV.L     @R15+,R3    ;
113 000010B4 63F6      113      .END
****TOTAL ERRORS      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.