To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 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 1st, 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.

H8/300H Tiny Series

Four-Digit BCD Multiplication (MULD)

Introduction

Multiplies two four-digit BCD (binary-coded-decimal) numbers and places the result (eight-digit BCD) in general registers.

Target Device

H8/300H Tiny Series

Contents

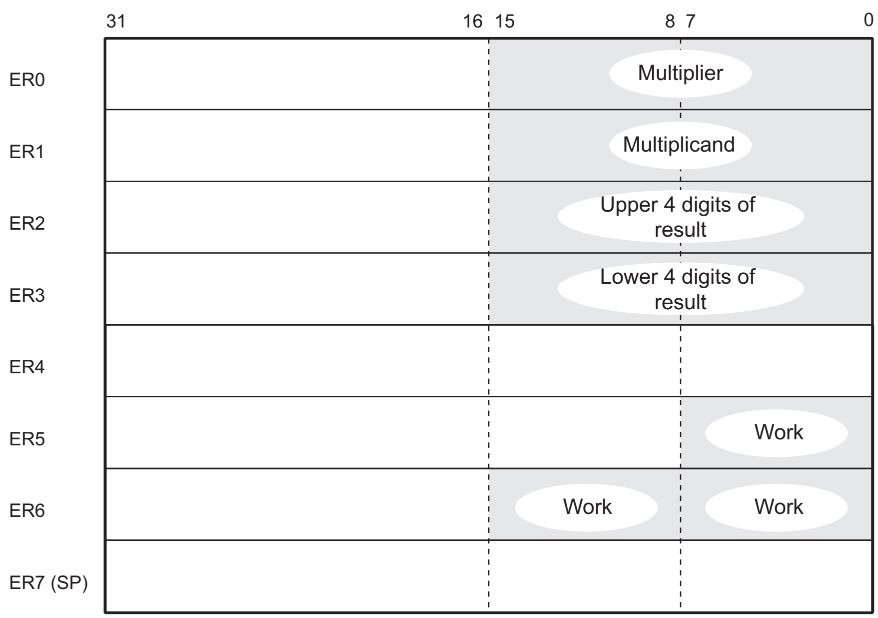
1. Function

1. Multiplies two four-digit BCD (binary-coded-decimal) numbers, and sets the result (eight-digit BCD) in general registers.
2. The arguments are all unsigned integers.
3. Data operations are entirely within the general registers.

2. Arguments

Contents		Storage Location	Data Length (Bytes)
Input	Multiplicand	R1	2
	Multiplier	R0	2
Output	Result	R2, R3	4

3. Changes to Internal Registers and Flags



I	UI	H	U	N	Z	V	C
–	–	↕	–	↕	↕	↕	↕

–: No change
↕: Varies
0: Fixed to 0
1: Fixed to 1

4. Programming Specifications

Program memory (bytes)	62
Data memory (bytes)	0
Stack (bytes)	0
Number of cycles	1192
Re-entrant	Yes
Relocatable	Yes
Interrupts during execution	Yes

5. Note

The number of cycles in the programming specifications is the value for calculating 9999×9999.

6. Description

6.1 Description of Functions

1. The arguments are as follows.
   R0: Set the 4-digit BCD multiplier (16 bits) as an input argument.
   R1: Set the 4-digit BCD multiplicand (16 bits) as an input argument.
   R2: The higher-order four digits of the 8-BCD-digit (32 bit) product are placed here as an output argument.
   R3: The lower-order four digits of the result are placed here.
2. The following figure illustrates the execution of the MULD subroutine. When the input arguments are set as shown below, the product is placed in R2 and R3.

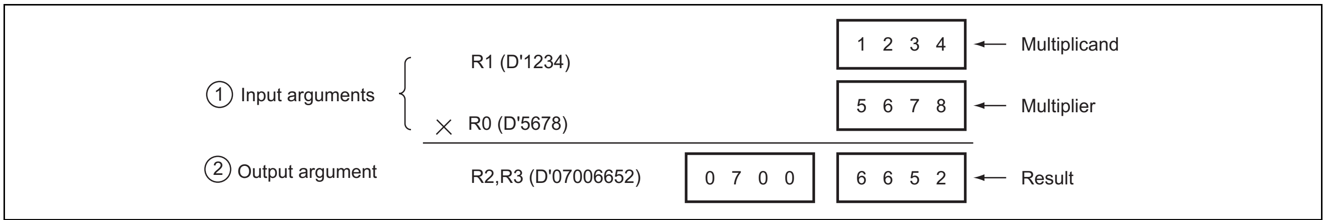


Figure 1 Example of MULD Execution

3. The subroutine always returns zero if either or both of the inputs are zero.

Table 1 Results When "0" Is Set As the Input Argument

Input Argument		Output Argument
Multiplicand (R1)	**Multiplier (R0)**	**Product (R2, R3)**
H'****	H'0000	H'00000000
H'0000	H'****	H'00000000
H'0000	H'0000	H'00000000

Note: H'**** refers to a hexadecimal number.

6.2 Usage Notes

1. When any upper digit of an input argument is not used, the "0" must be explicitly set in the digit, as is shown in figure 2. Otherwise, the correct result might not be obtained because the undefined data in the upper digits is included in the multiplication.

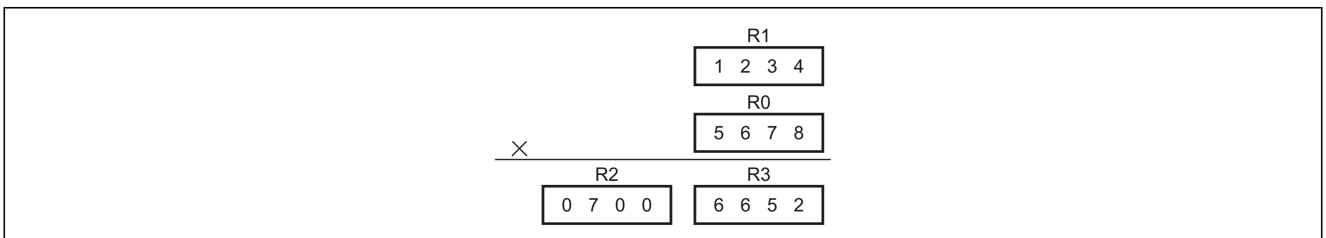


Figure 2 Multiplication When the Upper Digits are not Used

2. The multiplier placed in R0 will be lost in the execution of MULD. When you will still require the multiplier, save it elsewhere in memory beforehand.

6.3 Description of Data Memory

No data memory is used by MULD.

6.4 Example of Usage

After setting the multiplicand and multiplier, call the MULD subroutine.

```
WORK1   . RES. W 1           ......... Reservation of the data memory area for setting of the 4-digit BCD multiplicand by the user
                                       program.
WORK2   . RES. W 1           ......... Reservation of the data memory area for setting of the 4-digit BCD multiplier by the user
                                       program.
WORK3   . RES. W 2           ......... Reservation of the data memory area to hold the 8-digit BCD multiplication result for the user
                                       program.
            .
            .
            .
        MOV. W @WORK1, R1    ......... Sets, as an input argument for MULD, the 4-digit BCD multiplicand specified by the user
                                       program.
        MOV. W @WORK2, R0    ......... Sets the 4-digit BCD multiplier specified by the user program.
        JSR    @MULD         ......... Subroutine call of MULD.
        MOV. W R2, @WORK3    ......... Transfers the 8-digit BCD result from the output argument to the data memory of the user
                                       program.
        MOV. W R3, @WORK3+2
            .
            .
            .
```

6.5 Principles of Operation

1. Decimal multiplication is done by repeated addition. The following figure shows the multiplication 5678 × 1234 as an example.

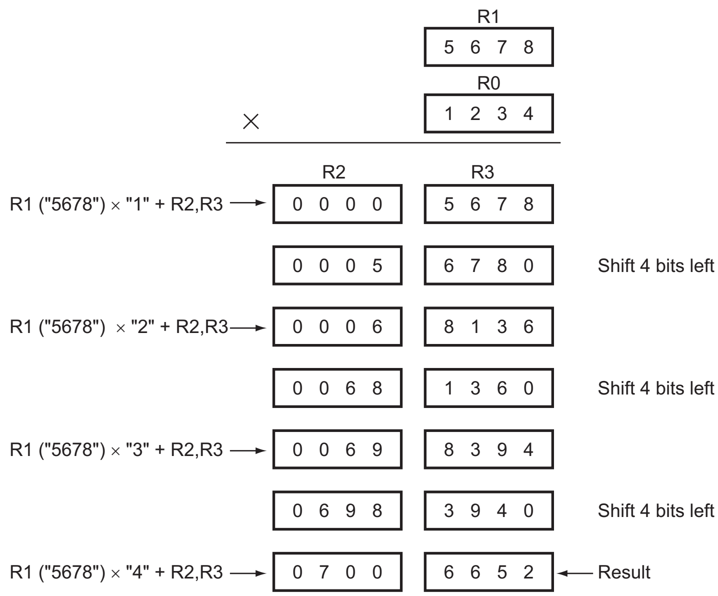


Figure 3 Multiplication (5678 × 1234)

Figure 6.3 illustrates the method used to find the product by repeatedly adding the multiplicand, multiplied by the respective digits of the multiplier from the leftmost digit, to the result and then shifting the result.

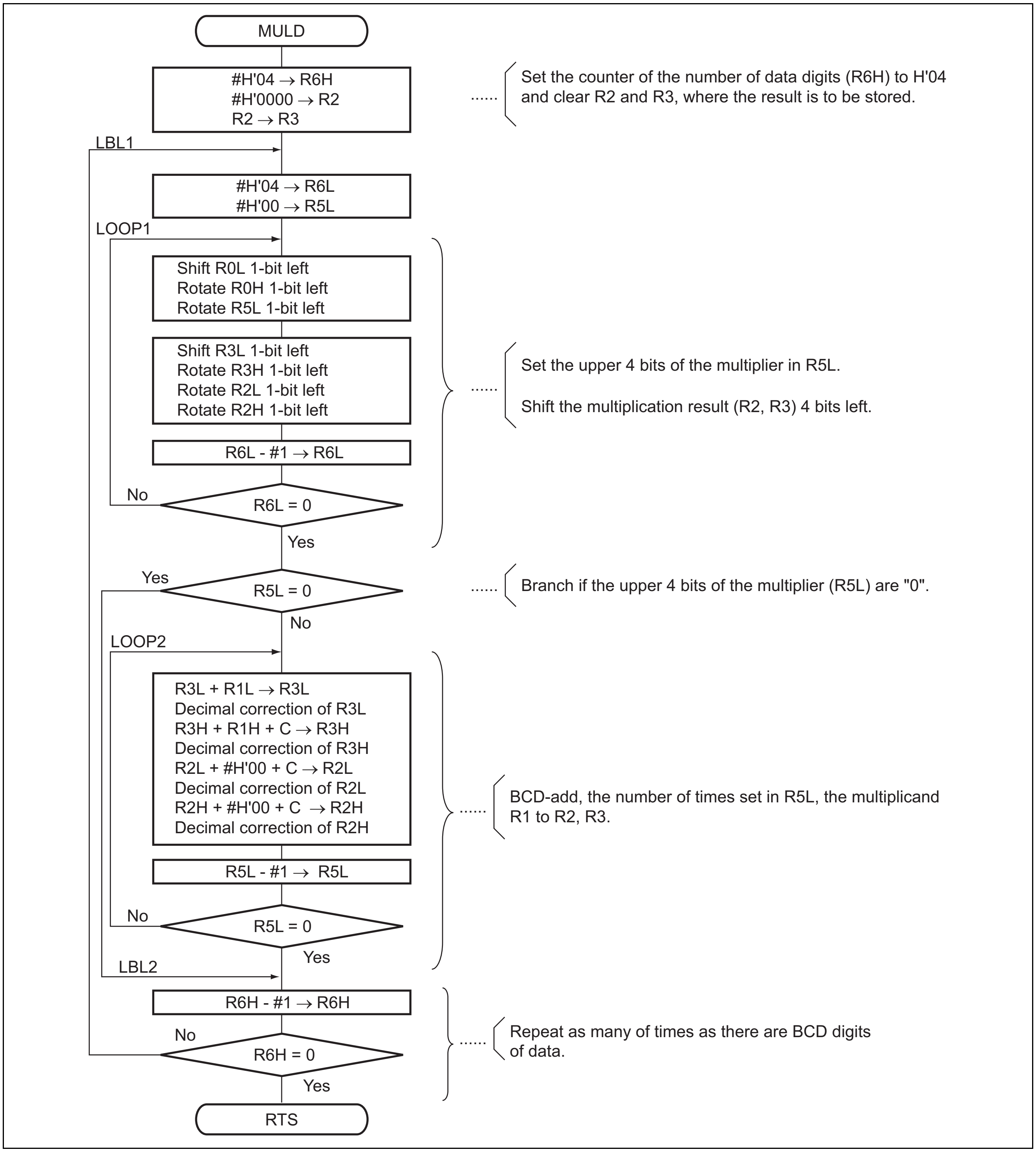
Firstly, the highest-order BCD digit (four bits) of the multiplier is taken, and the multiplicand is added this number of times. The result is then shifted four bits to the left (one BCD digit). Next, the second highest digit is taken from the multiplier, the multiplicand is added to the preceding result this number of times, and so on. The final result is found by repeating the processes as many times as there are BCD digits (four).

2. Details of the program are given below.

   The program:

   1) places H'04 in R6H as a counter that indicates the number of BCD digits in the data;
   2) clears R2 and R3, where the result of multiplication is to be stored;
   3) shifts R2 and R3 four bits (one BCD digit) left;
   4) loads one BCD digit from the higher-order end of the multiplier to R5L, and branches to step 6) if R5L is "0";
   5) BCD-adds the multiplicand to R2, R3 the same number of times as the value in R5L;
   6) decrements R6H; and
   7) repeats steps 3) to 6) above until R6H has become "0".

7. Flowchart


MULD
#H'04 → R6H
#H'0000 → R2
R2 → R3
Set the counter of the number of data digits (R6H) to H'04 and clear R2 and R3, where the result is to be stored.
LBL1
#H'04 → R6L
#H'00 → R5L
LOOP1
Shift R0L 1-bit left
Rotate R0H 1-bit left
Rotate R5L 1-bit left
Shift R3L 1-bit left
Rotate R3H 1-bit left
Rotate R2L 1-bit left
Rotate R2H 1-bit left
R6L - #1 → R6L
No
R6L = 0
Yes
Set the upper 4 bits of the multiplier in R5L.
Shift the multiplication result (R2, R3) 4 bits left.
Yes
R5L = 0
No
Branch if the upper 4 bits of the multiplier (R5L) are "0".
LOOP2
R3L + R1L → R3L
Decimal correction of R3L
R3H + R1H + C → R3H
Decimal correction of R3H
R2L + #H'00 + C → R2L
Decimal correction of R2L
R2H + #H'00 + C → R2H
Decimal correction of R2H
R5L - #1 → R5L
No
R5L = 0
Yes
BCD-add, the number of times set in R5L, the multiplicand R1 to R2, R3.
LBL2
R6H - #1 → R6H
No
R6H = 0
Yes
Repeat as many of times as there are BCD digits of data.
RTS

8. Program Listing

```
                        ;*************************************************************
                        ;*                                                           *
                        ;*          NAME : DECIMAL MULTIPLICATION                   *
                        ;*                 (MULD)                                    *
                        ;*                                                           *
                        ;*************************************************************
                        ;*                                                           *
                        ;*         ENTRY:  R1           (MULTIPLICAND)              *
                        ;*                 R0           (MULTIPLIER)                 *
                        ;*                                                           *
                        ;*        RETURN:  R2           (UPPER WORD OF RESULT)      *
                        ;*                 R3           (LOWER WORD OF RESULT)      *
                        ;*                                                           *
                        ;*************************************************************
                        ;
                                .CPU        300HN
0000                            .SECTION    MULD_code,CODE,ALIGN=2
                                .EXPORT     MULD
                        ;
     00000000           MULD    .EQU        $           ;Entry point
0000 F604                       MOV.B       #H'04,R6H   ;Set the bcd digit counter
0002 79020000                   MOV.W       #H'0000,R2  ;Clear R2
0006 0D23                       MOV.W       R2,R3       ;Clear R3
                        ;
0008                    LBL1
0008 FE04                       MOV.B       #H'04,R6L   ;Set bit counter
000A FD00                       MOV.B       #H'00,R5L   ;Clear R5L
000C                    LOOP1
000C 1008                       SHLL.B      R0L         ;Shift multiplier 1 bit left
000E 1200                       ROTXL.B     R0H
0010 120D                       ROTXL.B     R5L
0012 100B                       SHLL.B      R3L         ;Shift result 1 bit left
0014 1203                       ROTXL.B     R3H
0016 120A                       ROTXL.B     R2L
0018 1202                       ROTXL.B     R2H
001A 1A0E                       DEC.B       R6L         ;Decrement bit counter
001C 46EE                       BNE         LOOP1       ;Branch if Z=0
001E AD00                       CMP.B       #H'00,R5L
0020 4714                       BEQ         LBL2        ;Branch if Z=1
0022                    LOOP2
0022 089B                       ADD.B       R1L,R3L     ;R1L + R3L      --> R1L
0024 0F0B                       DAA.B       R3L         ;Decimal adjust R3L
0026 0E13                       ADDX.B      R1H,R3H     ;R1H + R3H + C  --> R1H
0028 0F03                       DAA.B       R3H         ;Decimal adjust R3H
002A 9A00                       ADDX.B      #H'00,R2L   ;R2L + #H'00 + C --> R2L
002C 0F0A                       DAA.B       R2L         ;Decimal adjust --> R2L
002E 9200                       ADDX.B      #H'00,R2H   ;R2H + #H'00 + C --> R2H
0030 0F02                       DAA.B       R2H         ;Decimal adjust R2H
0032 1A0D                       DEC.B       R5L         ;Clear bit 0 of R5L
0034 46EC                       BNE         LOOP2       ;Branch if Z=0
0036                    LBL2
0036 1A06                       DEC.B       R6H         ;Decrement bcd digit counter
```

```
 53   0038 46CE              53            BNE        LBL1        ;Branch if Z=0
 54                          54      ;
 55   003A 5470              55            RTS
 56                          56            .END
*****TOTAL ERRORS       0
*****TOTAL WARNINGS     0
```

Revision Record

Rev.	Date	Description	
		Page	Summary
2.00	Feb.28.06	—	Format has been changed from Hitachi version to Renesas version.