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

Example of Reset Synchronous PWM Control Using Timer Z

Introduction

The reset synchronous PWM mode of timer Z is used in producing a PWM waveform in normal and inverse phases.

Target Device

H8/300H Tiny Series H8/36049

Contents

1. Specifications

- The reset synchronous PWM mode of timer Z is used to produce a PWM waveform (normal and inverse phases) output.
- Pins FTI0B0 to FTI0D0 and FTI0A1 to FTI0D1 are set up as PWM output pins by default. TCNT_0 functions as an up-counter.
- By changing the value of GRB_0 on every interrupt generated by a match with GRA_0, the duty cycles of FTI0B0 and FTI0D0 can be changed.
- Buffered operation is not used.
- Output is toggled when TCNT_0 and GRC_0 match.
- TCNT_0 is cleared on a compare-match with GRA_0.
- TCNT_1 is independent of TCNT_0; its output is toggled when it matches GRC_1.
- TCNT_1 is used as a free-running counter.

2. Description of Functions

In this sample task, the reset-synchronous PWM mode of timer Z is used to output a pulse with a controlled duty cycle on the PWM output pin.

- System clock (ϕ)
  20-MHz reference clock that operates the CPU and peripheral functions
- Timer Start Register (TSTR)
  Selects operation or stoppage of the TCNT_0 and TCNT_1 counters. In this sample task, both counters are set to count.
- Timer Mode Register (TMDR)
  Selects timer synchronization/independence of counters TCNT_0 and TCNT_1. In this sample task, the two counters are set to operate independently. Normal rather than buffered operation is selected.
- Timer Function Control Register (TFCR)
  Selects settings and output levels for the various operating modes. In this sample task, reset-synchronous PWM mode operation is selected for channels 0 and 1. The initial output level is set to low, and the active level is set to high.
- Timer Output Master Enable Register (TOER)
  Enables/disables outputs on channels 0 and 1. In this sample task, all of the outputs are enabled.
- Timer Output Control Register (TOCR)
  Initial outputs, i.e. the outputs before the first occurrence of a compare-match, are set here. In this sample task, the initial outputs are all set to 0.
- Timer Control Register_0 (TCR_0)
  Selects the input clock and trigger for clearing of TCNT_0. In this sample task, TCNT_0 counts rising edges of ϕ and is cleared on matches with GRA_0.
- Timer Control Register_1 (TCR_1)
  Selects the input clock and trigger for clearing of TCNT_1. In this sample task, TCNT_1 counts rising edges of ϕ and clearing of TCNT_1 is disabled.
- Timer Counter_0 (TCNT_0)
  16-bit readable/writable up-counter which is incremented by cycles of an input internal/external clock signal. In this sample task, TCNT_0 counts rising edges of ϕ.
- Timer Counter_1 (TCNT_1)
  16-bit readable/writable up-counter which is incremented by cycles of an input internal/external clock signal. In this sample task, TCNT_1 counts rising edges of ϕ.
- General Registers (GRA_0, GRA_1, GRB_0, GRB_1, GRC_0, GRC_1)
  16-bit readable/writable registers, the contents of which are always compared with the counter of TCNT_0.

- Input Capture/Output Compare Pin C0 (FTI0C0)
  Toggled output that synchronizes the PWM cycle period.
- Input Capture/Output Compare Pin B0 (FTI0B0)
  PWM output 1
- Input Capture/Output Compare Pin D0 (FTI0D0)
  PWM output 1 (inverse of PWM output 1)
- Input Capture/Output Compare Pin A1 (FTI0A1)
  PWM output 2
- Input Capture/Output Compare Pin C1 (FTI0C1)
  PWM output 2 (inverse of PWM output 2)
- Input Capture/Output Compare Pin B1 (FTI0B1)
  PWM output 3
- Input Capture/Output Compare Pin D1 (FTI0D1)
  PWM output 3 (inverse of PWM output 3)
- Channel 0 Interrupt (ITMZ0)
  In this sample task, interrupts on matches between TCNT_0 and GRC_0 are used.
- Channel 1 Interrupt (ITMZ1)
  In this sample task, interrupts on matches between TCNT_1 and GRC_1 are used.
- P30 and P31 Terminals of I/O Port (P30, P31)
  Output toggling by GRC_0 interrupts is selected for P30, and output toggling by GRC_1 interrupts is selected for P31.

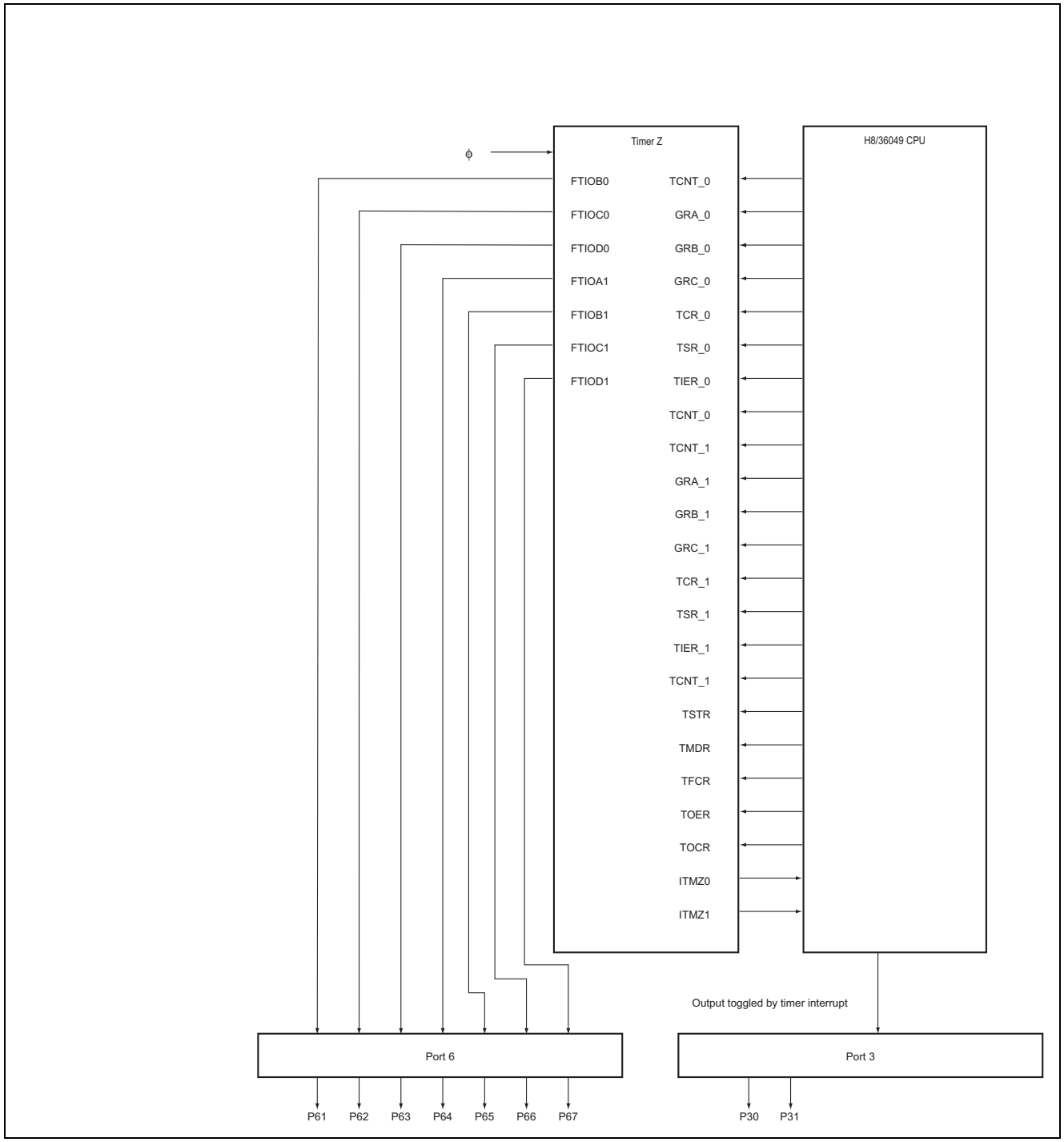


Figure 1 Block Diagram of Reset-Synchronous PWM Output Using Timer Z

The assignment of functions and operation of reset-synchronous PWM output in this sample task are as described in table 1.

Table 1 Assignment of Functions

Element	Description
TSTR	Selects operation and stoppage of TCNT_0 and TCNT_1
TMDR	Sets independent operation of TCNT_0 and TCNT_1
TFCR	Sets reset-synchronous PWM mode operation for channels 0 and 1
TOER	Enables all outputs
TOCR	Sets 0 as the initial value for all outputs
TCR_0	Selects input clock and trigger for clearing of TCNT_0
TCR_1	Selects input clock and trigger for clearing of TCNT_1
TCNT_0	16-bit readable and writable counter that is incremented by cycles of the input clock
GRA_0	Constantly compared with TCNT_0
GRA_1	Constantly compared with TCNT_0
GRB_0	Constantly compared with TCNT_0
GRB_1	Constantly compared with TCNT_0
GRC_0	Constantly compared with TCNT_0
GRC_1	Constantly compared with TCNT_0
FTIOC0 pin	PWM cycle period in synchronization with the output toggled
FTIOB0 pin	PWM output 1
FTIOD0 pin	PWM output 1 (inverse of PWM output 1)
FTIOA1 pin	PWM output 2
FTIOC1 pin	PWM output 2 (inverse of PWM output 2)
FTIOB1 pin	PWM output 3
FTIOD1 pin	PWM output 3 (inverse of PWM output 3)
ITMZ0	Channel 0 interrupt generated by matches with GRC_0
ITMZ1	Channel 1 interrupt generated by matches with GRC_1
Port 3	Operation as interrupt-toggled outputs

3. Principles of Operation

Figure 2 shows the principles of operation for this task. The figure describes how reset-synchronous PWM output operation is obtained through a combination of hardware and software processing.

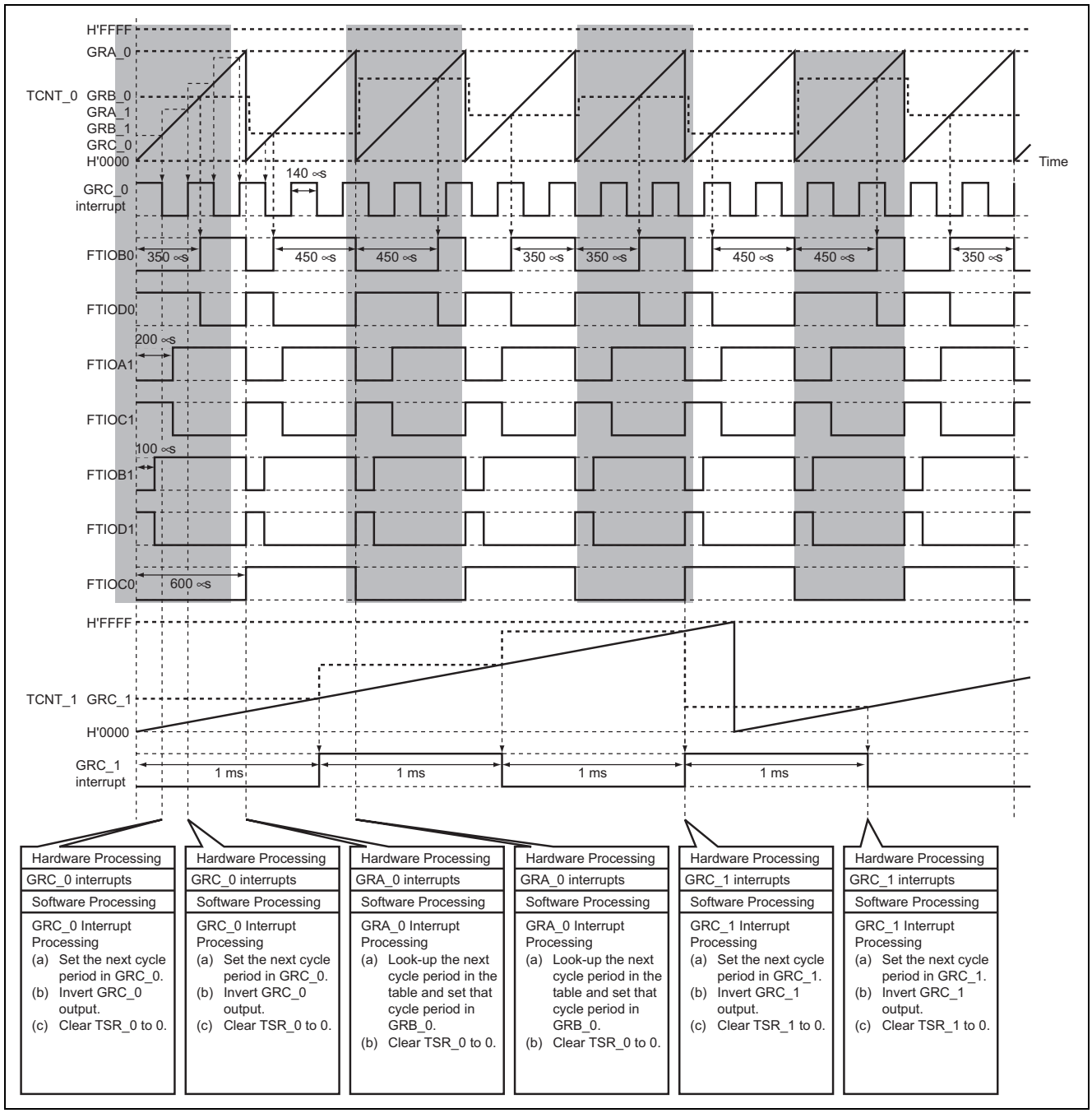


Figure 2 Using Timer Z to Produce a Reset-Synchronous PWM Output

4. Description of Software

4.1 Modules

Table 2 shows the modules used in this sample task.

Table 2 Description of Modules

Module	Label Name	Description
Main routine	main	Selects the compare-match function of Timer Z, starts the counters, and selects the compare-match output pins.
Timer Z0 interrupt handler	tmrz0	On interrupts generated by matches with GRC_0: Sets the point where the next Timer Z0 interrupt will be triggered and drives output toggling. On interrupts generated by matches GRA_0: Sets the next interrupt point.
Timer Z1 interrupt handler	tmrz1	Sets the point where the next Timer Z1 interrupt will be triggered.

4.2 Arguments

No arguments are used in this sample task.

4.3 Internal Registers

The following describes the internal registers used in this sample task.

- TSTR Timer Start Register Address: 0xFFF720

Bit	Bit Name	Setting	Function
1	STR1	0/1	Channel 1 Counter Start 0: TCNT_1 is stopped. 1: TCNT_1 counts.
0	STR0	0/1	Channel 0 Counter Start 0: TCNT_0 is stopped. 1: TCNT_0 counts.

- TMDR Timer Mode Register Address: 0xFFF721

Bit	Bit Name	Setting	Function
7	BFD1	0	Buffer Operation D1 0: GRD_1 operates normally.
6	BFC1	0	Buffer Operation C1 0: GRC_1 operates normally.
5	BFD0	0	Buffer Operation D0 0: GRD_0 operates normally.
4	BFC0	0	Buffer Operation C0 0: GRC_0 operates normally.
0	SYNC	0	Timer Synchronization 0: TCNT_1 and TCNT_0 operate independently.

- TFCR Timer Function Control Register Address: 0xFFF723

Bit	Bit Name	Setting	Function
3	OLS1	1	Output Level Select 1 1: Initial output is low, and the active level is high.
2	OLS0	1	Output Level Select 0 1: Initial output is low, and the active level is high.
1	CMD1	CMD1 = 0	Combination Mode 1 and 0
0	CMD0	CMD0 = 1	CMD1 = 0, CMD0 = 1: Channels 1 and 0 are used together in reset-synchronous PWM mode operation.

- TOER Timer Output Master Enable Register Address: 0xFFF724

Bit	Bit Name	Setting	Function
7	ED1	0	Master Enable D1 0: Output on pin FTIOD1 is enabled.
6	EC1	0	Master Enable C1 0: Output on pin FTIOC1 is enabled.
5	EB1	0	Master Enable B1 0: Output on pin FTIOB1 is enabled.
4	EA1	0	Master Enable A1 0: Output on pin FTIOA1 is enabled.
3	ED0	0	Master Enable D0 0: Output on pin FTIOD0 is enabled.
2	EC0	0	Master Enable C0 0: Output on pin FTIOC0 is enabled.
1	EB0	0	Master Enable B0 0: Output on pin FTIOB0 is enabled.
0	EA0	0	Master Enable A0 0: Output on pin FTIOA0 is enabled.

- TOCR Timer Output Control Register Address: 0xFFF725

Bit	Bit Name	Setting	Function
7	TOD1	0	Output Level Select D1 TOD1 = 0: Sets 0 as the initial output of pin FTIOD1.
6	TOC1	0	Output Level Select C1 TOC1 = 0: Sets 0 as the initial output of pin FTIOC1.
5	TOB1	0	Output Level Select B1 TOB1 = 0: Sets 0 as the initial output of pin FTIOB1.
4	TOA1	0	Output Level Select A1 TOA1 = 0: Sets 0 as the initial output of pin FTIOA1.
3	TOD0	0	Output Level Select D0 TOD0 = 0: Sets 0 as the initial output of pin FTIOD0.
2	TOC0	0	Output Level Select C0 TOC0 = 0: Sets 0 as the initial output of pin FTIOC0.
1	TOB0	0	Output Level Select B0 TOB0 = 0: Sets 0 as the initial output of pin FTIOB0.
0	TOA0	0	Output Level Select A0 TOA0 = 0: Sets 0 as the initial output of pin FTIOA0.

- TCR_0 Timer Control Register_0 Address: 0xFFF700

Bit	Bit Name	Setting	Function
7	CCLR2	CCLR2 = 0	Counter Clear 2 to 0
6	CCLR1	CCLR1 = 0	CCLR2 = 0, CCLR1 = 0, CCLR0 = 1:
5	CCLR0	CCLR0 = 1	Clears TCNT0 on compare-matches with GRA0.
4	CKEG1	CKEG1 = 0	Clock Edge 1 to 0
3	CKEG0	CKEG0 = 0	CKEG1 = 0, CKEG0 = 0: Rising edges are counted.
2	TPSC2	TPSC2 = 0	Timer Prescaler 2 to 0
1	TPSC1	TPSC1 = 0	TPSC2 = 0, TPSC1 = 0, TPSC0 = 0:
0	TPSC0	TPSC0 = 0	Counting is driven by ϕ .

- TCR_1 Timer Control Register_1 Address: 0xFFF710

Bit	Bit Name	Setting	Function
7	CCLR2	CCLR2 = 0	Counter Clear 2 to 0
6	CCLR1	CCLR1 = 0	CCLR2 = 0, CCLR1 = 0, CCLR0 = 0:
5	CCLR0	CCLR0 = 1	Disable Clearing TCNT1.
4	CKEG1	CKEG1 = 0	Clock Edge 1 to 0
3	CKEG0	CKEG0 = 0	CKEG1 = 0, CKEG0 = 0: Rising edges are counted.
2	TPSC2	TPSC2 = 0	Timer Prescaler 2 to 0
1	TPSC1	TPSC1 = 0	TPSC2 = 0, TPSC1 = 0, TPSC0 = 0:
0	TPSC0	TPSC0 = 0	Counting is driven by ϕ .

- TCNT_0 Timer Counter _0 Address: 0xFFF706
  Function: 16-bit up-counter that counts rising edges of ϕ.
  Set value: 0
- TCNT_1 Timer Counter _1 Address: 0xFFF716
  Function: 16-bit up-counter that counts rising edges of ϕ.
  Set value: 0
- GRA_0 General Register A _0 Address: 0xFFF708
  Function: When the value set in GRA_0 matches the value counted by TCNT_0, a compare-match occurs.
  Set value: 12000
- GRA_1 General Register A _1 Address: 0xFFF718
  Function: When the value set in GRA_1 matches the value counted by TCNT_0, a compare-match occurs.
  Set value: 4000
- GRB_0 General Register B _0 Address: 0xFFF70A
  Function: When the value set in GRB_0 matches the value counted by TCNT_0, a compare-match occurs.
  Set value: Sets one data in GRB_DATA alignment at every GRA_0 interrupt

  (In this sample task, four data are repeatedly set.)
- GRB_1 General Register B _1 Address: 0xFFF71A
  Function: When the value set in GRB_1 matches the value counted by TCNT_0, a compare-match occurs.
  Set value: 2000
- GRC_0 General Register C _0 Address: 0xFFF70C
  Function: When the value set in GRC_0 matches the value counted by TCNT_0, a compare-match occurs.
  Set value: 2800
- GRC_1 General Register C _1 Address: 0xFFF71C
  Function: When the value set in GRC_1 matches the value counted by TCNT_1, a compare-match occurs.
  Set value: 20000

4.4 RAM Usage

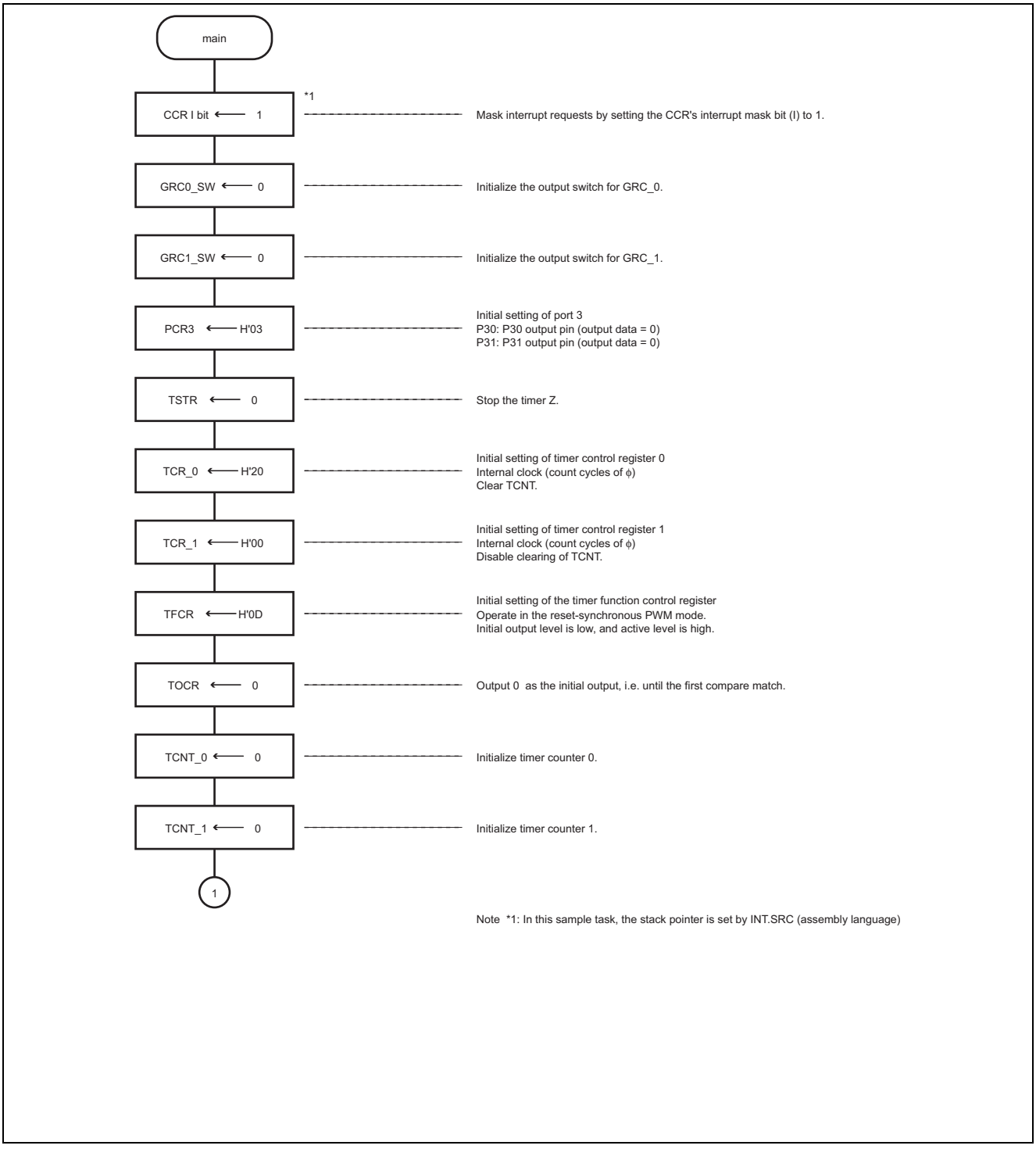
Table 3 describes the RAM usage in this sample task.

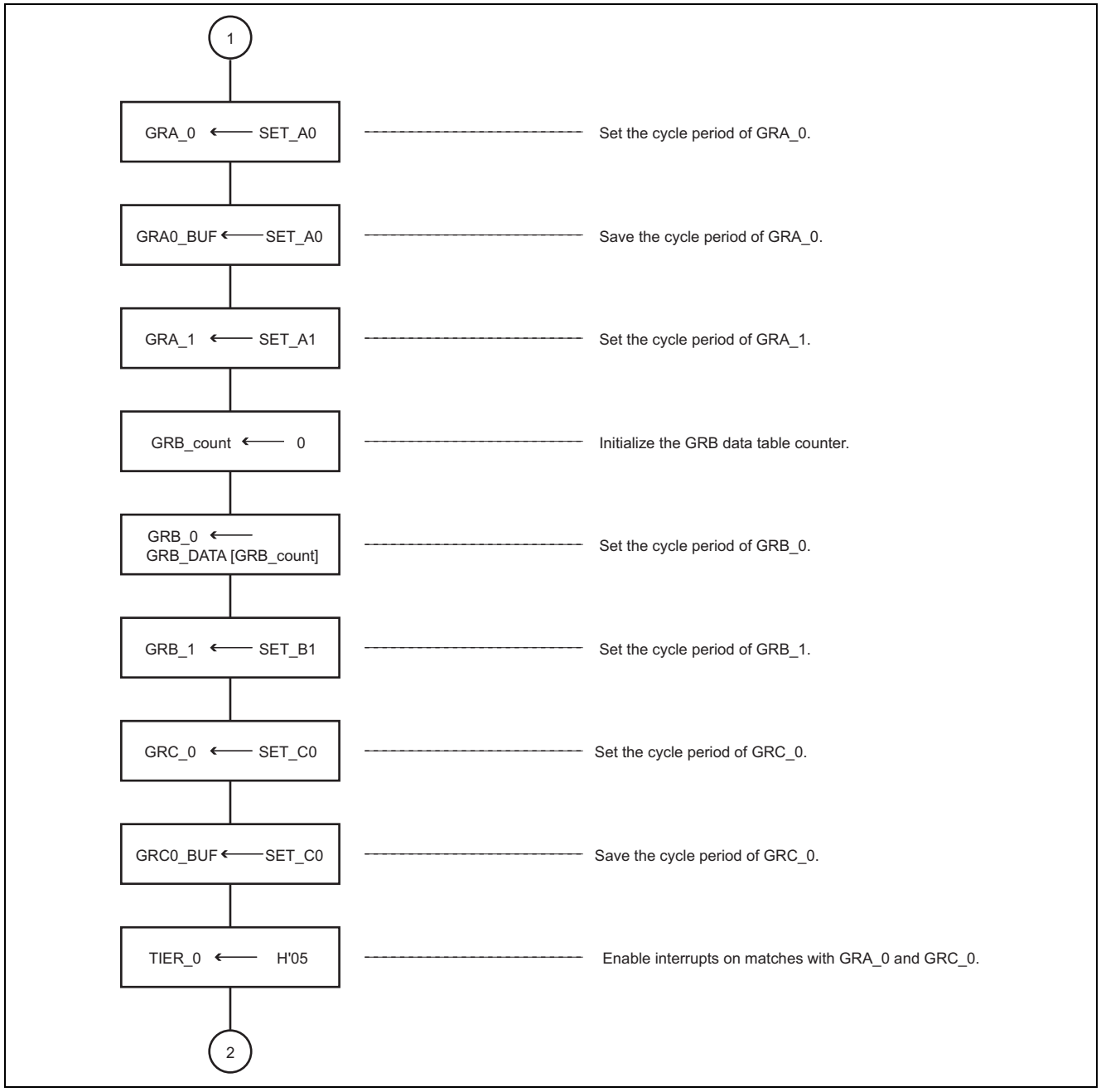
Table 3 Description of RAM Used

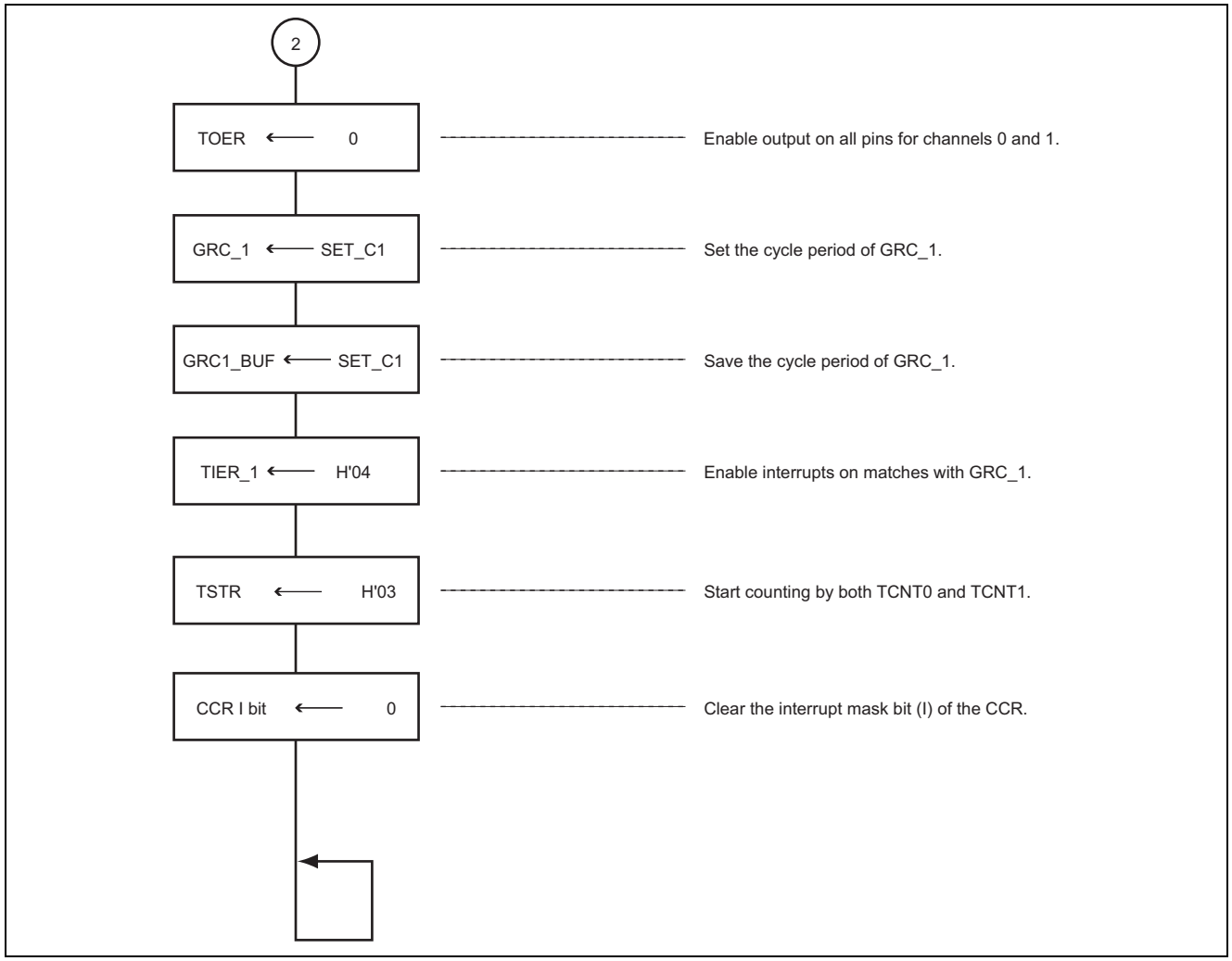
Label Name	Function	Address	Label Name of Module Used
TSR_SV	Saves the value of TSR_0	H'FFE812	tmrz0
GRB_count	Index counter for data to be placed in GRB_0	H'FFE800	main, tmrz0
GRA0_BUF	Saves the initially set data for GRA_0	H'FFE802	main, tmrz0
GRC0_BUF	Holds the data for output driven by matches with GRC_0	H'FFE806	main, tmrz0
GRC1_BUF	Saves the data for output driven by matches with GRC_1	H'FFE80A	main, tmrz1
GRC0_SW	Switch for toggling output driven by GRC_0	H'FFE80E	main, tmrz0
GRC1_SW	Switch for toggling output driven by GRC_1	H'FFE810	main, tmrz1

5. Flowcharts

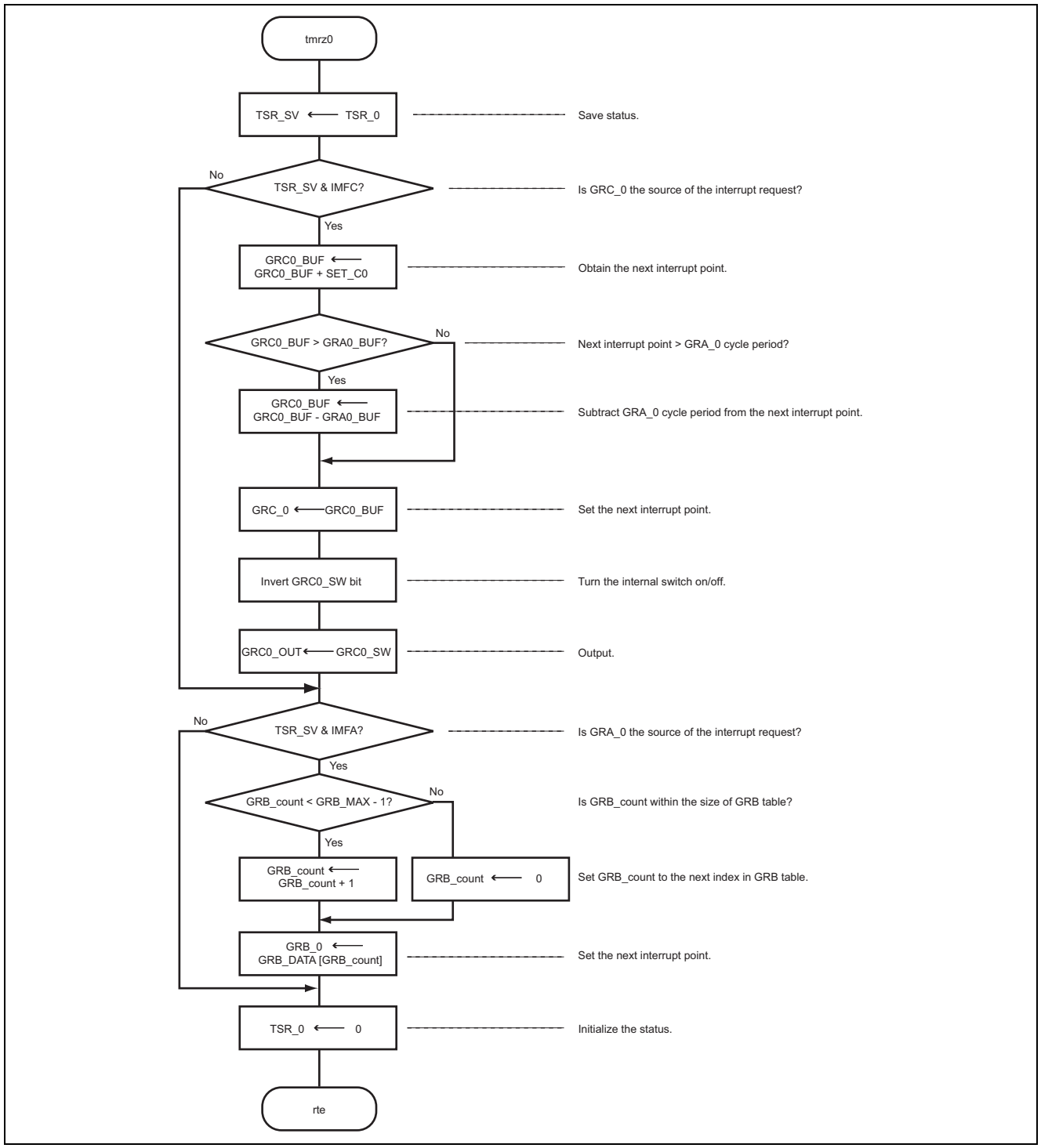
5.1 Main Routine (main)



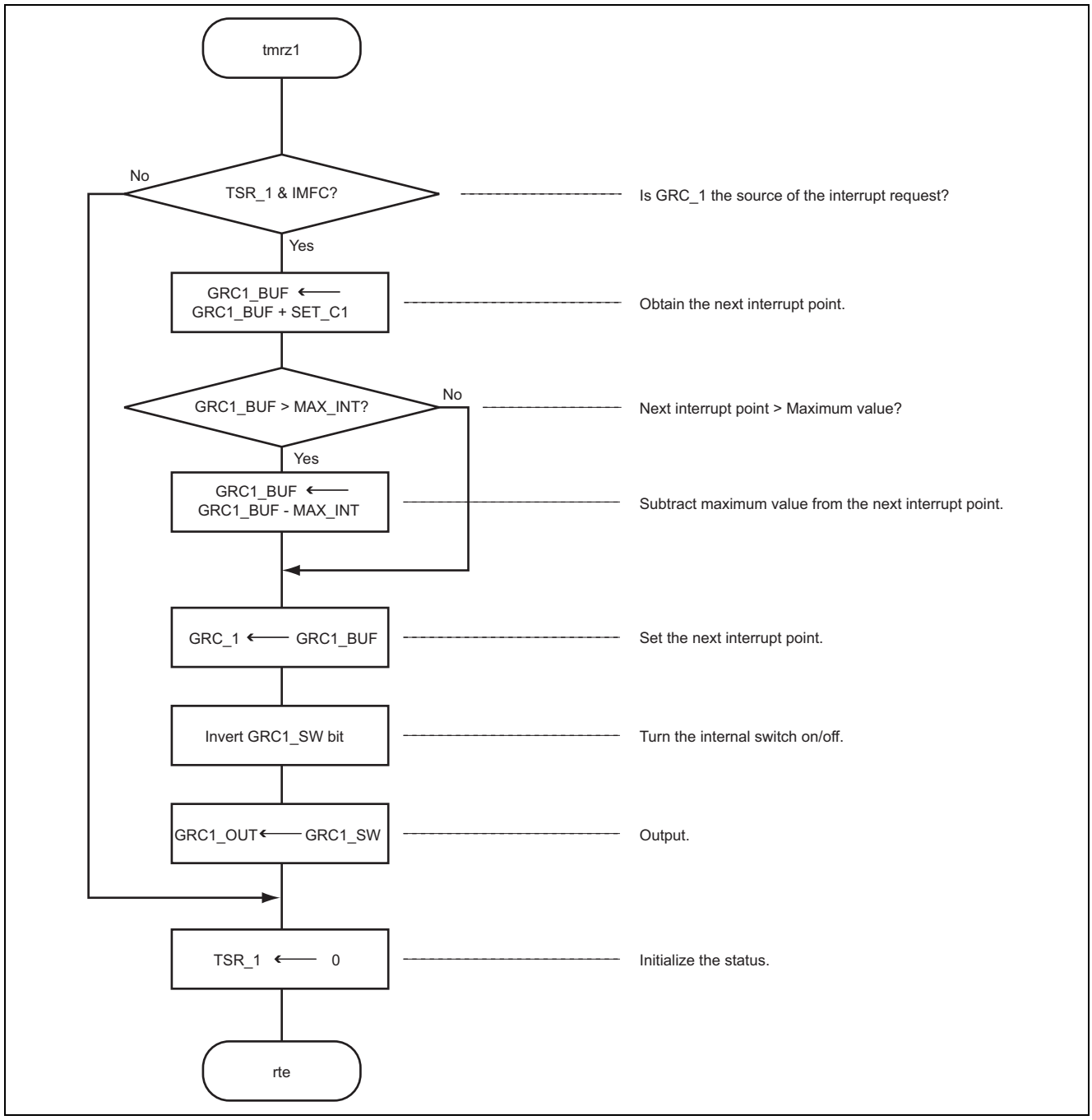

1
GRA_0 ← SET_A0
Set the cycle period of GRA_0.
GRA0_BUF ← SET_A0
Save the cycle period of GRA_0.
GRA_1 ← SET_A1
Set the cycle period of GRA_1.
GRB_count ← 0
Initialize the GRB data table counter.
GRB_0 ← GRB_DATA [GRB_count]
Set the cycle period of GRB_0.
GRB_1 ← SET_B1
Set the cycle period of GRB_1.
GRC_0 ← SET_C0
Set the cycle period of GRC_0.
GRC0_BUF ← SET_C0
Save the cycle period of GRC_0.
TIER_0 ← H'05
Enable interrupts on matches with GRA_0 and GRC_0.
2


2
TOER ← 0
Enable output on all pins for channels 0 and 1.
GRC_1 ← SET_C1
Set the cycle period of GRC_1.
GRC1_BUF ← SET_C1
Save the cycle period of GRC_1.
TIER_1 ← H'04
Enable interrupts on matches with GRC_1.
TSTR ← H'03
Start counting by both TCNT0 and TCNT1.
CCR I bit ← 0
Clear the interrupt mask bit (I) of the CCR.

5.2 Timer Z0 Interrupt Processing Routine (tmrz0)



5.3 Timer Z1 Interrupt Processing Routine (tmrz1)



6. Program Listing

INIT.SRC (Program listing)

```
    .export   _INIT
    .import   _main
;
    .section  P,CODE
_INIT:
    mov.l  #h'fff000,sp
    ldc.b  #b'10000000,ccr
    jmp@_main
;
    .end
```

```
/* H8/300H Tiny Series -H8/36049- Application Note */
/* Application Version */
/* Usage Example of Internal Timer Z */

#include <machine.h>

/* Symbol definitions */
struct BIT {
    unsigned char b7:1;        /* bit 7 */
    unsigned char b6:1;        /* bit 6 */
    unsigned char b5:1;        /* bit 5 */
    unsigned char b4:1;        /* bit 4 */
    unsigned char b3:1;        /* bit 3 */
    unsigned char b2:1;        /* bit 2 */
    unsigned char b1:1;        /* bit 1 */
    unsigned char b0:1;        /* bit 0 */
};

#define PDR3      *(volatile unsigned char *)0xFFFFD6        /* Port Data Register 3 */
#define PDR3_BIT (*(struct BIT *)0xFFFFD6)                   /* Port Data Register 3 */
#define GRC0_OUT  PDR3_BIT.b0
                                                  /* Output it by interrupt from GRC0 */
#define GRC1_OUT  PDR3_BIT.b1
                                                   /* Output it by interrupt from GRC1 */
#define PCR3      *(volatile unsigned char *)0xFFFFE6     /* Port Control Register 3 */

                                                                          /* COMMON */
#define TSTR    *(volatile unsigned char *)0xFFF720          /* Timer Start Register */
#define TMDR    *(volatile unsigned char *)0xFFF721           /* Timer Mode Register */
#define TPMR    *(volatile unsigned char *)0xFFF722       /* Timer PWM Mode Register */
#define TFCR    *(volatile unsigned char *)0xFFF723
                                                    /* Timer Function Control Register */
#define TOER    *(volatile unsigned char *)0xFFF724
                                               /* Timer Output Master Enable Register */
#define TOCR    *(volatile unsigned char *)0xFFF725
```

```
                                                    /* Timer Output Control Register */

                                                                          /* CHANNEL 0 */
#define TCR_0   *(volatile unsigned char *)0xFFF700
                                                         /* Timer Control Register_0 */
#define TIORA_0 *(volatile unsigned char *)0xFFF701
                                                    /* Timer I/O Control RegisterA_0 */
#define TIORC_0 *(volatile unsigned char *)0xFFF702
                                                    /* Timer I/O Control RegisterC_0 */
#define TSR_0   *(volatile unsigned char *)0xFFF703       /* Timer Status Register_0 */
#define TIER_0  *(volatile unsigned char *)0xFFF704
                                               /* Timer Interrupt Enable Register_0 */
#define POCR_0  *(volatile unsigned char *)0xFFF705
                                              /* PWM Mode Output Level Control Register_0 */
#define TCNT_0  *(volatile unsigned int *)0xFFF706                /* Timer Counter_0 */
#define GRA_0   *(volatile unsigned int *)0xFFF708           /* General Register A_0 */
#define GRB_0   *(volatile unsigned int *)0xFFF70A           /* General Register B_0 */
#define GRC_0   *(volatile unsigned int *)0xFFF70C           /* General Register C_0 */
#define GRD_0   *(volatile unsigned int *)0xFFF70E           /* General Register D_0 */

                                                                          /* CHANNEL 1 */
#define TCR_1   *(volatile unsigned char *)0xFFF710      /* Timer Control Register_1 */
#define TIORA_1 *(volatile unsigned char *)0xFFF711
                                                    /* Timer I/O Control RegisterA_1 */
#define TIORC_1 *(volatile unsigned char *)0xFFF712
                                                    /* Timer I/O Control RegisterC_1 */
#define TSR_1   *(volatile unsigned char *)0xFFF713       /* Timer Status Register_1 */
#define TIER_1  *(volatile unsigned char *)0xFFF714
                                               /* Timer Interrupt Enable Register_1 */
#define POCR_1  *(volatile unsigned char *)0xFFF715
                                              /* PWM Mode Output Level Control Register_1 */
#define TCNT_1  *(volatile unsigned int *)0xFFF716                /* Timer Counter_1 */
#define GRA_1   *(volatile unsigned int *)0xFFF718           /* General Register A_1 */
#define GRB_1   *(volatile unsigned int *)0xFFF71A           /* General Register B_1 */
#define GRC_1   *(volatile unsigned int *)0xFFF71C           /* General Register C_1 */
#define GRD_1   *(volatile unsigned int *)0xFFF71E           /* General Register D_1 */

#define IMFA    0x01                                         /* bit position of IMFA */
#define IMFC    0x04                                         /* bit position of IMFC */
#define GRB_MAX    4                                               /* GRB table size */

#define SET_A0  12000                                       /* setting value for GRA_0 */
#define SET_A1  4000                                        /* setting value for GRA_1 */
#define SET_B1  2000                                        /* setting value for GRB_1 */
#define SET_C0  2800                                        /* setting value for GRC_0 */
#define SET_C1  20000                                       /* setting value for GRC_1 */

#define MAX_INT 65535                                             /* integer max value */


#pragma interrupt (tmrz0)
#pragma interrupt (tmrz1)
```

```
/* function definition */
extern void INIT(void);                                    /* Stack pointer set */
void main(void);                                           /* main routine */
void tmrz0(void);
                                                  /* Timer Z0 interrupt routine */
void tmrz1(void);
                                                  /* Timer Z1 interrupt routine */

/* Data table */
const unsigned int GRB_DATA[GRB_MAX] =                          /* GRB table */
{
    7000,
    3000,
    9000,
    5000
};

/* RAM definition */
unsigned char TSR_SV;                                            /* Save TSR */
int GRB_count;                                                  /* GRB counter */
unsigned long GRA0_BUF;                                      /* Buffer for GRA_0 */
unsigned long GRC0_BUF;                                      /* Buffer for GRC_0 */
unsigned long GRC1_BUF;                                      /* Buffer for GRC_1 */
int GRC0_SW;                                           /* Inverse switch for GRC_0 */
int GRC1_SW;                                           /* Inverse switch for GRC_1 */

/* Vector address */
#pragma section V1                                           /* Vector section set */
void (*const VEC_TBL1[])(void) = {
    INIT                                                   /* H'0000 Reset vector */
};
#pragma section V2                                           /* Vector section set */
void (*const VEC_TBL2[])(void) = {
    tmrz0
                                                /* H'0068 Timer Z0 interrupt vector */
};
#pragma section V3                                           /* Vector section set */
void (*const VEC_TBL3[])(void) = {
    tmrz1
                                                /* H'006C Timer Z1 interrupt vector */
};
#pragma section                                                          /* P */

/************************************************************/
/* Main program                                             */
/************************************************************/
void main(void)
{
    set_imask_ccr(1);                                          /* CCR I-bit = 1 */

    GRC0_SW = 0;                                        /* initialize GRC_0 switch */
```

```
    GRC1_SW = 0;                                        /* initialize GRC_1 switch */

    PCR3 = 0x03;                                          /* initialize Port 3 */

    TSTR = 0;                                                 /* stop timer */
    TCR_0 = 0x20;
                                    /* select Internal clock(φ), Clears TCNT by GRA */
    TCR_1 = 0x00;
                                  /* select Internal clock(φ), Disables TCNT clearing */
    TFCR = 0x0D;
                                           /* operate in reset synchronous PWM mode */
                                                  /* initial output is low */
    TOCR = 0;
                                                /* selects the initial outputs */
    TCNT_0 = 0;                                         /* clear timer counter */
    TCNT_1 = 0;                                         /* clear timer counter */
    GRA_0 = SET_A0;                                           /* set period */
    GRA0_BUF = SET_A0;                                      /* keep period */
    GRA_1 = SET_A1;                                           /* set period */
    GRB_count = 0;                                          /* init counter */
    GRB_0 = GRB_DATA[GRB_count];                              /* set period */
    GRB_1 = SET_B1;                                           /* set period */

    GRC_0 = SET_C0;                                           /* set period */
    GRC0_BUF = SET_C0;                                      /* keep period */

    TIER_0 = 0x05;
                                               /* enable GRA_0,GRC_0 interrupt */

    TOER = 0;                                                /* enable output */

    GRC_1 = SET_C1;                                           /* set period */
    GRC1_BUF = SET_C1;                                      /* keep period */
    TIER_1 = 0x04;                                  /* enable GRC_1 interrupt */

    TSTR = 0x03;                                              /* start timer */

    set_imask_ccr(0);                                        /* CCR I-bit = 0 */

    while(1);
}

/*********************************************************/
/* Timer Z0 Interrupt                                    */
/*********************************************************/
void tmrz0(void)
{

    TSR_SV = TSR_0;                                           /* save status */

    /* interrupt by GRC_0 */
    if(TSR_SV & IMFC) {
```

```
        GRC0_BUF += SET_C0;                                     /* get next period */
        if(GRC0_BUF > GRA0_BUF) {
            GRC0_BUF -= GRA0_BUF;
        }
        GRC_0 = GRC0_BUF;                                       /* set next period */

        GRC0_SW ^= 1;                                            /* reverse switch */
        GRC0_OUT = GRC0_SW;                                       /* output signal */
    }

    /* interrupt by GRA_0 */
    if(TSR_SV & IMFA) {
        if(GRB_count < (GRB_MAX - 1))                            /* get next index */
            GRB_count++;
        else
            GRB_count = 0;
        GRB_0 = GRB_DATA[GRB_count];
                                                  /* set next period from table */
    }
    TSR_0 = 0;                                                    /* clear status */
}

/*********************************************************/
/* Timer Z1 Interrupt                                    */
/*********************************************************/
void tmrz1(void)
{

    /* interrupt by GRC_1 */
    if(TSR_1 & IMFC) {
        GRC1_BUF += SET_C1;                                     /*get next period */
        if(GRC1_BUF > MAX_INT) {
            GRC1_BUF -= MAX_INT;
        }
        GRC_1 = GRC1_BUF;                                       /*set next period */

        GRC1_SW ^= 1;                                           /* reverse switch */
        GRC1_OUT = GRC1_SW;                                      /* output signal */
    }
    TSR_1 = 0;                                                    /* clear status */
}
```

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Jun.10.05	—	First edition issued

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.