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.

H8/300L SLP Series

Control of a Brushed Motor

Introduction

A brushed DC motor is controlled by using the 10-bit PWM function of the H8/38024. Each time the IRQ1 switch is pressed, the motor operation changes in the following sequence: stop → 75% output rotation → 87.3% output rotation → 100% output rotation → 87.3% output rotation → 75% output rotation → stop.

Target Device

H8/38024

Contents

1. Specifications 2
2. Concepts 3
3. Description of Functions 4
4. Principle of Operation 6
5. Description of Software 8
6. Flowchart 10
7. Program Listing 12

1. Specifications

1. A brushed DC motor is controlled by using the 10-bit PWM function of the H8/38024.
2. In this sample task, the motor operation changes in the following sequence each time the IRQ1 switch is pressed: stop → 75% output rotation → 87.3% output rotation → 100% output rotation → 87.3% output rotation → 75% output rotation → stop.
3. Figure 1.1 illustrates how to connect a brushed DC motor to be controlled.

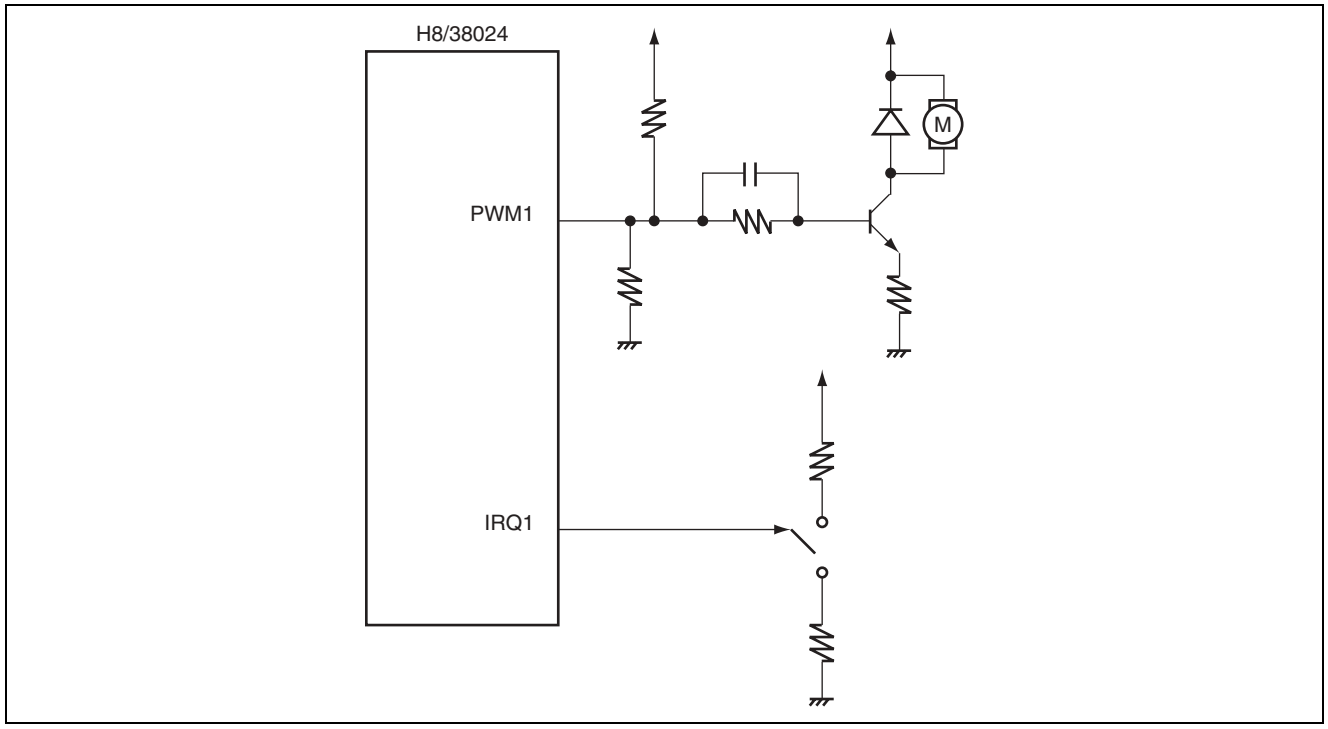


Figure 1.1 Configuration for Sample Task

2. Concepts

1. Example of brushed DC motor operation
   Figure 2.1 shows an example of speed control of the brushed DC motor by PWM waveforms. The following gives an operation overview.
   A. If the duty cycle has changed, the transistor's on/off interval also changes, resulting in changes in the average current supplied to the motor and, thus, in the motor's rpm.
   B. If the duty cycle of the PWM waveform is 75%, the motor runs at 75% of full RPM output.
   C. If the duty cycle of the PWM waveform is 87.5%, the motor runs at 87.5% of full RPM output.
   D. If the duty cycle of the PWM waveform is 100%, the motor runs at 100% full RPM output.

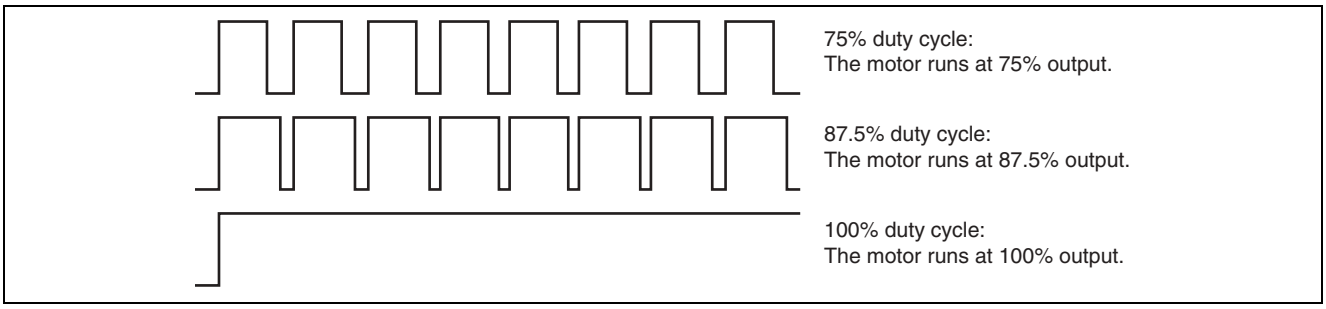


Figure 2.1 Example of Brushed DC Motor Operation

3. Description of Functions

1. The sample task uses a brushed DC motor (FA-130 manufactured by Mabuchi Motor Co., Ltd.) for use in a model. Table 3.1 lists its specifications.

Table 3.1 FA-130 Specifications

Item	Value
Type	FA-130
Limit voltage [V]	1.5 to 3.0
Recommended voltage [V]	1.5
Recommended load [g·cm]	5.4
RPM under no load [rpm]	8500
RPM under recommended load [rpm]	5900
Current consumption under recommended load [mA]	640

2. The following explains the H8/38024's functions used to control the brushed DC motor. Figure 3.1 shows the block diagram of the functions involved in this sample task.

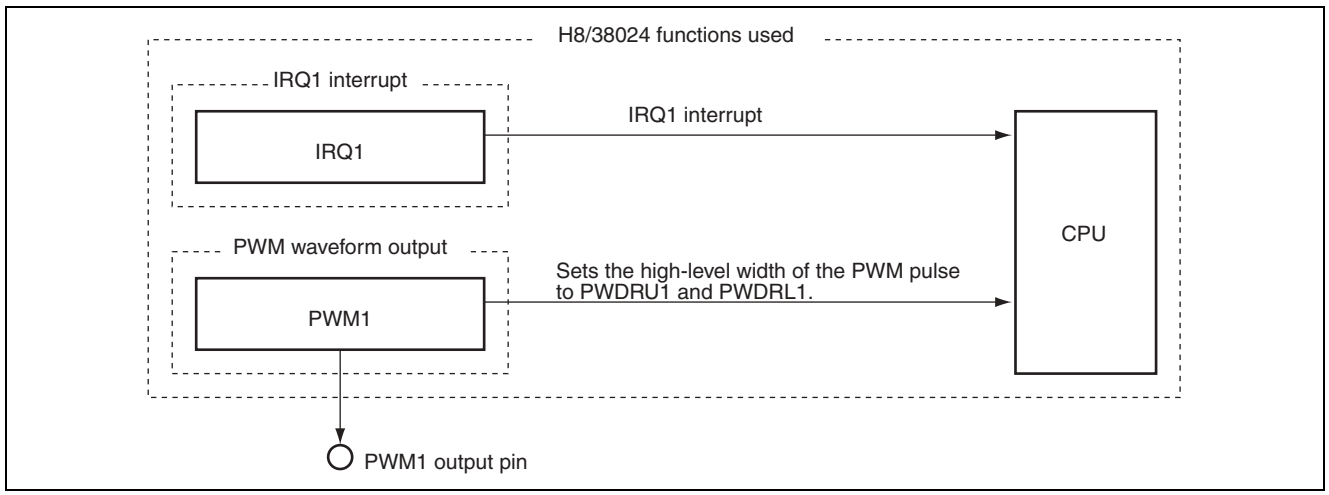


Figure 3.1 H8/38024 Functions Used

3. This section describes the functions of the PWM1.
   A. Figure 3.2 shows the block diagram of the 10-bit PWM function. The following explains the block diagram of this function.
      - The system clock (ϕ), which is rated at 5 MHz, is used as a reference clock to operate the CPU and its peripheral functions.
      - The PWM1 control register (PWCR1) is an eight-bit write-only register that selects an input clock.
      - A pulse-division method is used to reduce ripples.
      - The PWM1 data registers U and L (PWDRU1 and PWDRL1) are 10-bit write-only registers. The upper two bits provide PWDRU1, while the lower eight bits provide PWDRL1. The contents written to PWDRU1 and PWDRL1 correspond to the total of high level width over one PWM waveform cycle. Writing 10-bit data to PWDRU1 and PWDRL1 moves their contents to the PWM waveform generator, and updates the data for PWM waveform generation. To set 10-bit data, the lower eight bits must be written to PWDRL1 first, and then the upper two bits to PWDRU1.
      - The port mode register 9 (PMR9) is an eight-bit readable/writable register that controls selection of the pin functions of port 9. Use the P90/PWM1 pin function select bit (PWM1), PMR9's bit 0, to set the P90/PWM1 pin as the PWM1 output pin.
      - The PWM1 output pin (PWM1) outputs a PWM waveform based on the pulse division method.

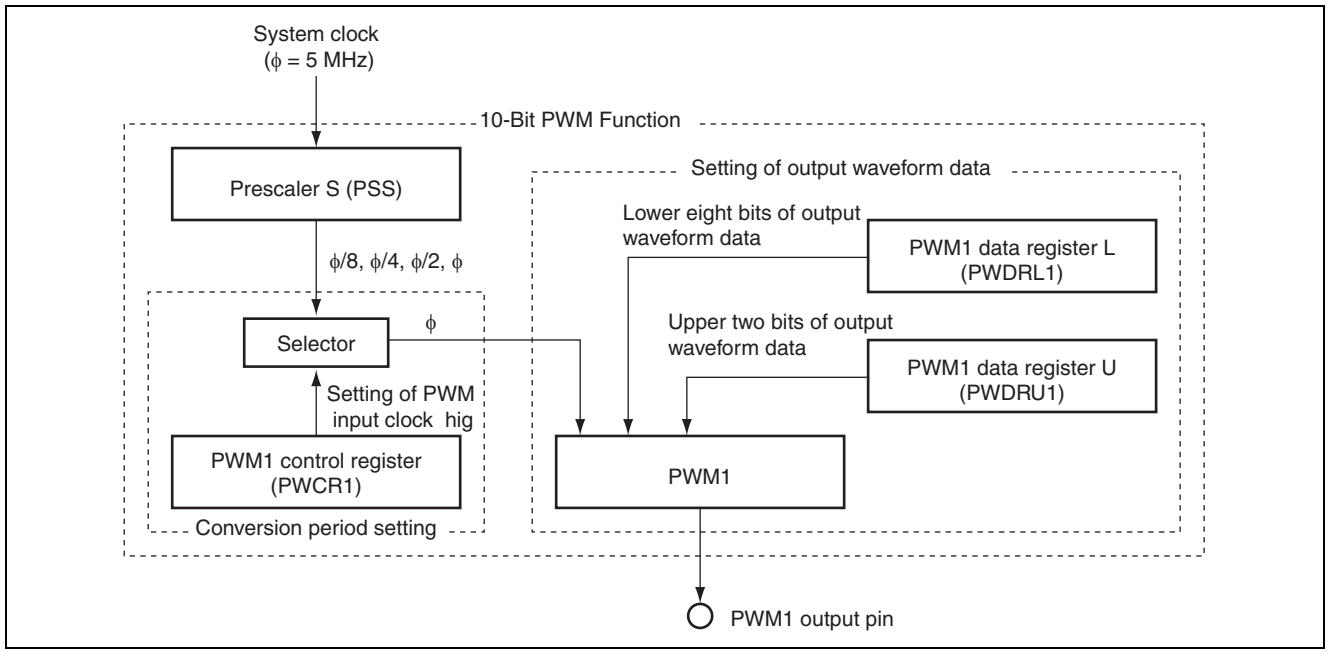


Figure 3.2 Block Diagram of PWM1

4. Table 3.2 lists the functions assigned for the sample task. With these function assignments, control of a brushed DC motor by PWM waveform is implemented.

Table 3.2 Function Assignments

Function	Function Allocation
PSS	13-bit up-counter that uses the system clock (5 MHz) as input
PWCR1	Selects a clock supplied to the 10-bit PWM.
PWDRU1	Sets the upper two bits of PWM output waveform data.
PWDRL1	Sets the lower eight bits of PWM output waveform data.
PWM1	PWM1 waveform output pin
IEGR	Enables an $\overline{\text{IRQ1}}$ pin interrupt request.
IENR1	Selects the edge of $\overline{\text{IRQ1}}$ pin input.
IRR1	Indicates the presence or absence of an IRQ1 interrupt.
IRQ1	Motor rotation speed selector

4. Principle of Operation

1. Figure 4.1 is the flowchart for brushed DC motor control.

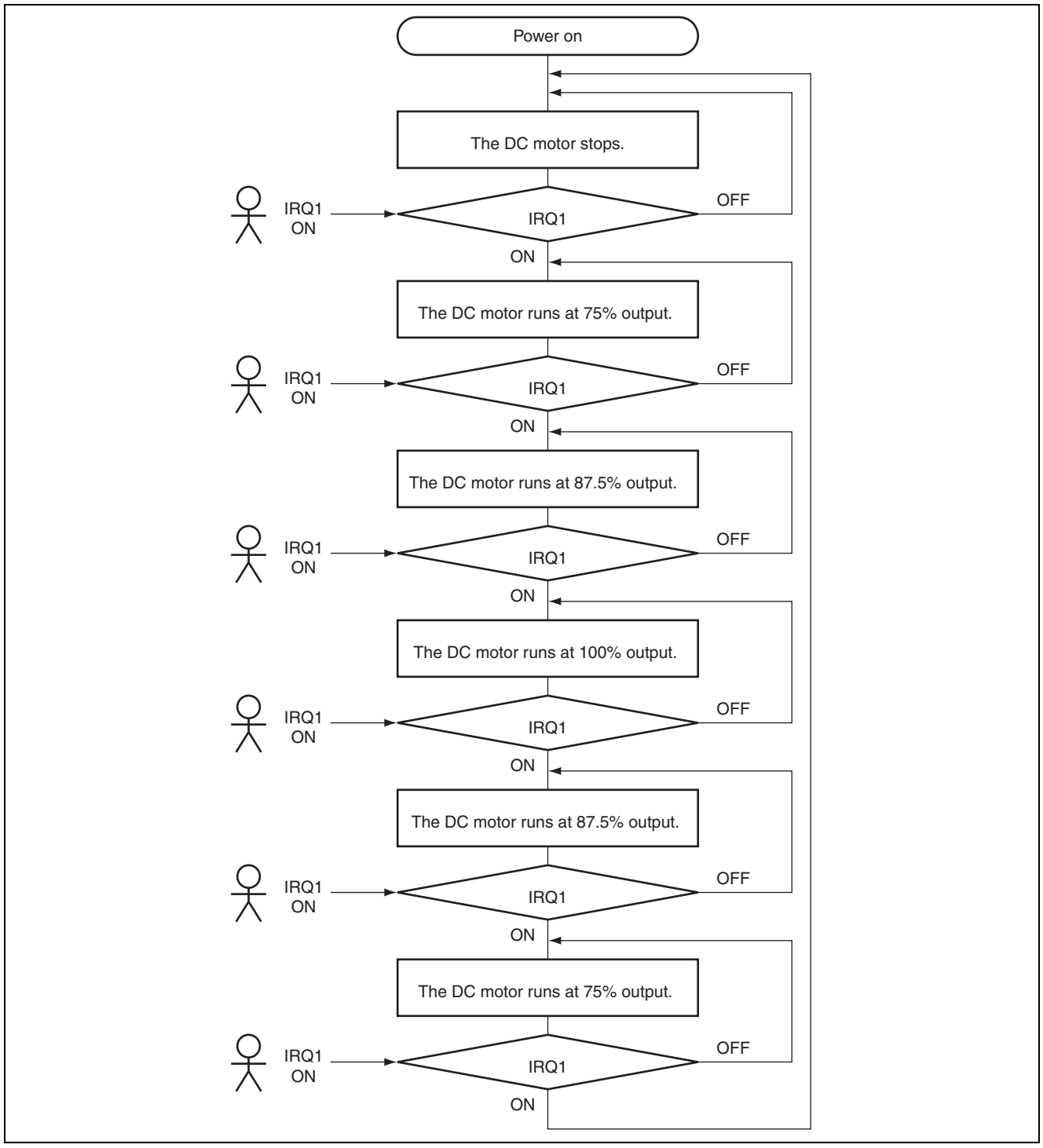


Figure 4.1 Flowchart for Brushed DC Motor Control

2. Motor output and PWM waveforms

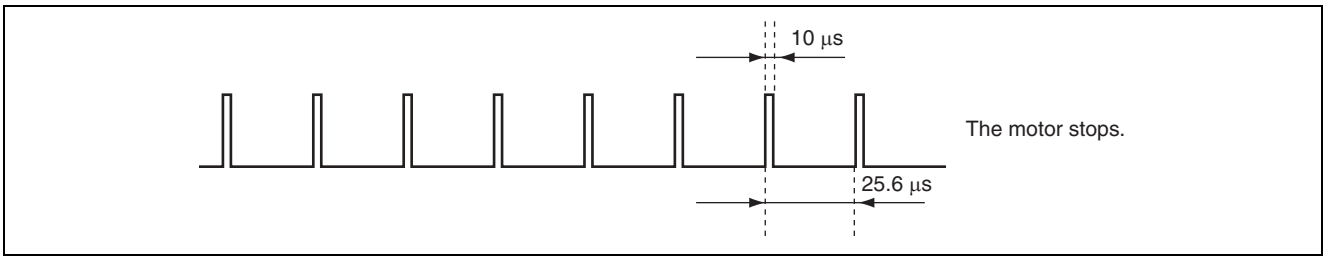


Figure 4.2 PWM Waveform for Stop

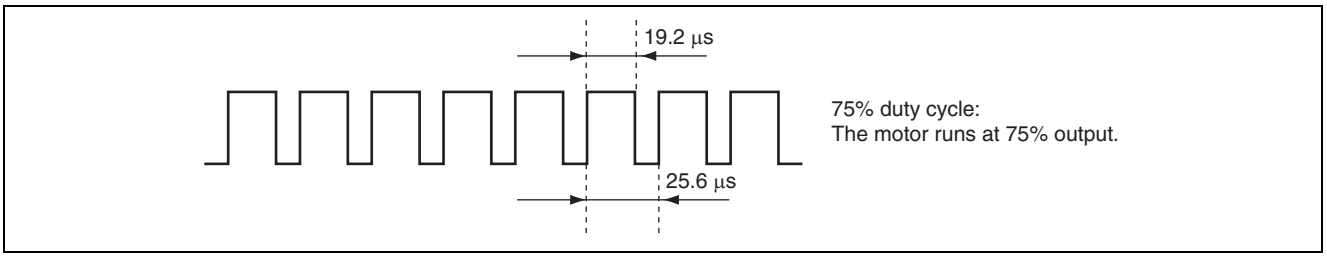


Figure 4.3 PWM Waveform for 75% Output

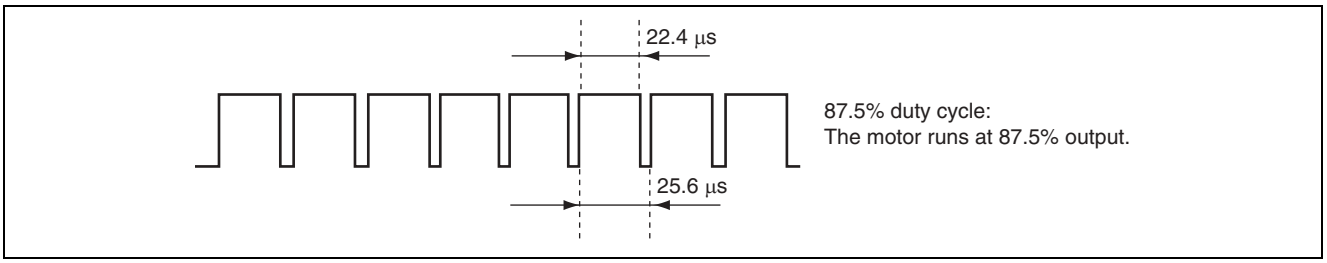


Figure 4.4 PWM Waveform for 87.5% Output

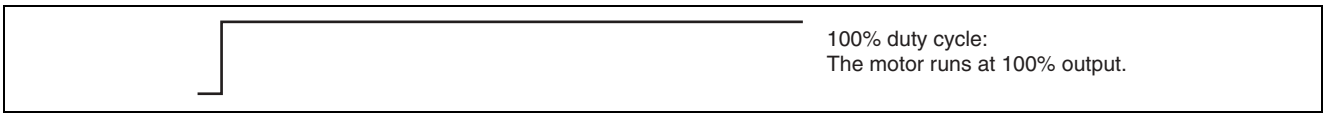


Figure 4.5 PWM Waveform for 100% Output

5. Description of Software

5.1 Modules

Table 5.1 lists the modules of this sample task.

Table 5.1 Description of Modules

Module	Label	Function
Main routine	main	Sets global variables and PWM1, and enables the interrupt.
IRQ1 interrupt processing	irq1int	Routine to set a duty cycle for PWM1. Changes the speed of rotation according to the number of IRQ1 interrupts.

5.2 Arguments

The sample task uses no arguments.

5.3 Internal Registers

Table 5.2 lists the internal registers used for the sample task.

Table 5.2 Description of Internal Registers

Register		Function	Address	Setting
PWCR1	PWCR10	PWM1 control register (clock selection 1 and 0)	0xFFD0	PWCR10 = 0
	PWCR11	When PWCR10 = 0 and PWCR11 = 0, sets the clock supplied to the 10-bit PWM as ϕ.	Bit 1 Bit 0	PWCR11 = 0
PWDRU1		PWM1 data register U Sets the upper two bits of PWM output waveform data.	0xFFD1	0x00
PWDRL1		PWM1 data register L Sets the lower eight bits of PWM output waveform data.	0xFFD2	0x00
PMR9	PWM1	Port mode register 9 (P90/PWM1 pin function selection) When PWM1 = 0, sets the P90/PWM1 pin for the P90 output pin function. When PWM1 = 1, sets the P90/PWM1 pin for the PWM1 output pin function.	0xFFEC Bit 0	1
PMRB	IRQ1	Port mode register B (PB3/AN3/$\overline{\text{IRQ1}}$ pin function selection) When IRQ1 = 0, functions as the PB3/AN3 input pin. When IRQ1 = 1, functions as the $\overline{\text{IRQ1}}$/TMIC input pin.	0xFFEE Bit 3	1
IEGR	IEG1	IRQ edge select register (IRQ1 edge selection) When IEG1 = 0, selects the falling edge detection for the $\overline{\text{IRQ1}}$ pin input. When IEG1 = 1, selects the rising edge detection for the $\overline{\text{IRQ1}}$ pin input.	0xFFF2 Bit 1	1
IENR1	IEN1	Interrupt enable register 1 (IRQ1 interrupt enable) When IEN1 = 0, disables the $\overline{\text{IRQ1}}$ pin interrupt request. When IEN1 = 1, enables the $\overline{\text{IRQ1}}$ pin interrupt request.	0xFFF3 Bit 1	1
IRR1	IRRI1	Interrupt request register 1 (IRQ1 interrupt request flag) When IRRI1 = 0, an $\overline{\text{IRQ1}}$ pin interrupt has not been requested. When IRRI1 = 1, an $\overline{\text{IRQ1}}$ pin interrupt has been requested.	0xFFF6 Bit 1	0

5.4 Description of RAM

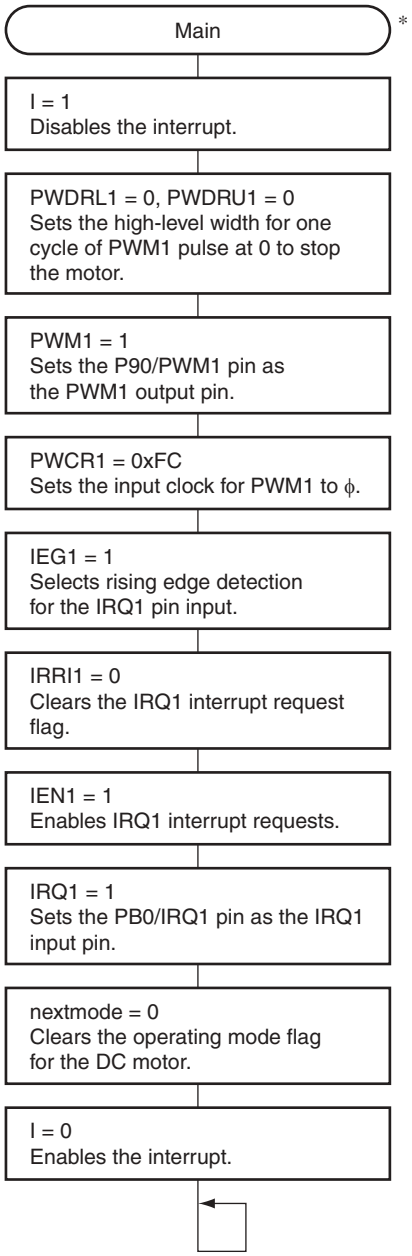
Table 5.3 lists the RAM used for the sample task.

Table 5.3 Description of RAM

Label	Function	Address	Used in
nextmode	Changes the duty cycle for PWM1 to obtain appropriate motor rpm.	char/1 byte	main, irq1int

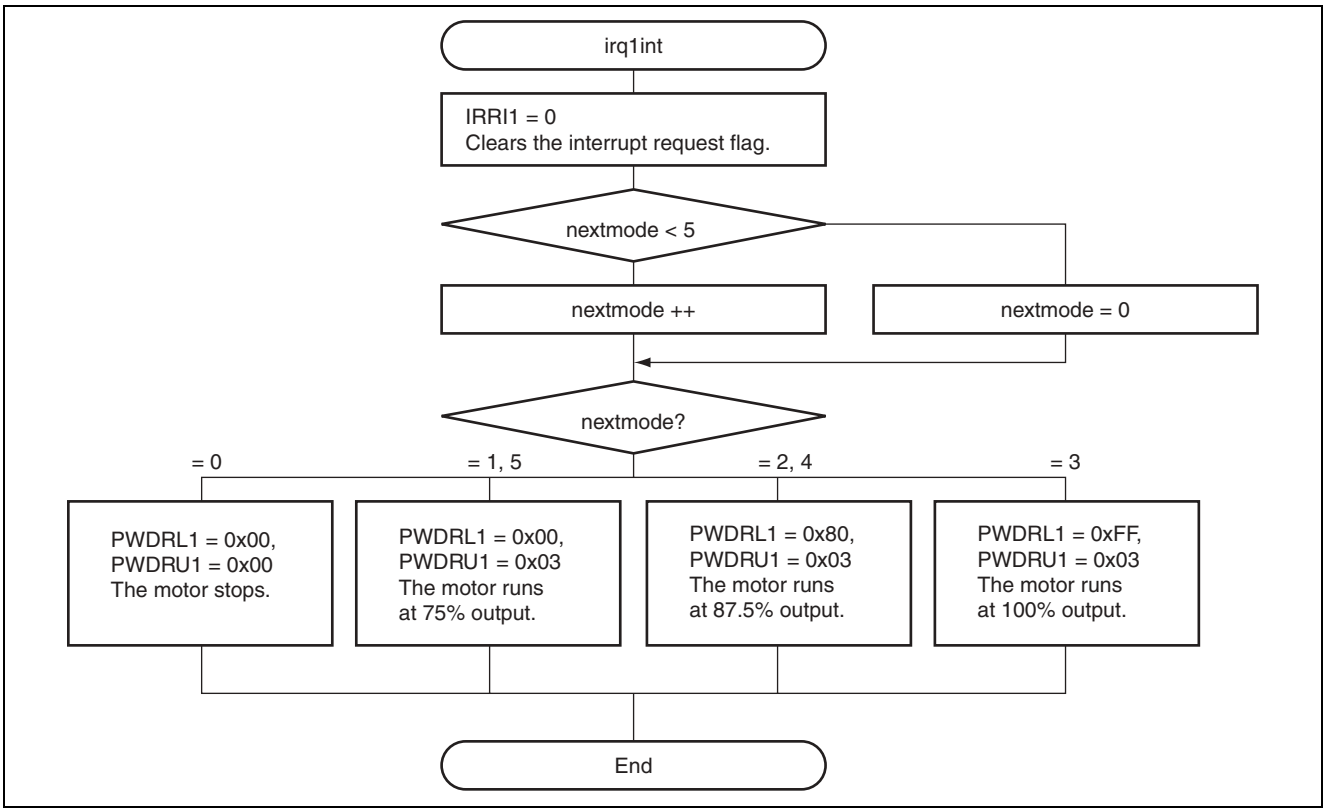
6. Flowchart

1. Main routine



Note: * In this sample task, the stack pointer is set in INIT.SRC (assembly language).

2. IRQ1 interrupt



7. Program Listing

INIT.SRC (Program list)

```
      .export  _INIT
      .import  _main
  ;
      .section P,CODE
  _INIT:
      mov.w    #h'ff80,r7
      ldc.b    #b'10000000,ccr
      jmp @_main
  ;
      .end
```

```
/************************************************************************************************************************/
/*                                                                                                                      */
/*  H8/300L Super Low Power Series                                                                                      */
/*      -H8/38024 Series-                                                                                               */
/*  Application Note                                                                                                    */
/*                                                                                                                      */
/*  'DC Motor Function'                                                                                                 */
/*                                                                                                                      */
/*  Function                                                                                                            */
/*  : 10bit PWM                                                                                                         */
/*                                                                                                                      */
/*  External Clock : 10MHz                                                                                              */
/*  Internal Clock : 5MHz                                                                                               */
/*  Sub Clock      : 32.768kHz                                                                                          */
/*                                                                                                                      */
/************************************************************************************************************************/

#include    <machine.h>

/************************************************************************************************************************/
/*  Symbol Definition                                                                                                   */
/************************************************************************************************************************/
struct BIT {
    unsigned char   b7:1;       /* bit7 */
    unsigned char   b6:1;       /* bit6 */
    unsigned char   b5:1;       /* bit5 */
    unsigned char   b4:1;       /* bit4 */
    unsigned char   b3:1;       /* bit3 */
    unsigned char   b2:1;       /* bit2 */
    unsigned char   b1:1;       /* bit1 */
    unsigned char   b0:1;       /* bit0 */
};

#define     PWCR1       *(volatile unsigned char *)0xFFD0       /* PWM Control Register                    */
#define     PWDRU1      *(volatile unsigned char *)0xFFD1       /* PWM Data Register U                     */
#define     PWDRL1      *(volatile unsigned char *)0xFFD2       /* PWM Data Register L                     */
#define     PMR9_BIT    (*(struct BIT *)0xFFEC)                 /* Port Mode Register 9                    */
#define     PWM1        PMR9_BIT.b0                             /* P90/PWM1 pin function switches          */
#define     IEGR1_BIT   (*(struct BIT *)0xFFF2)                 /* Interrupt Edge Select Register 1        */
#define     IEG1        IEGR1_BIT.b1                            /* IEG1 Edge Select                        */
#define     IENR1_BIT   (*(struct BIT *)0xFFF3)                 /* Interrupt Enable Register 1             */
#define     IEN1        IENR1_BIT.b1                            /* IEN1 Interrupt Enable                   */
#define     IRR1_BIT    (*(struct BIT *)0xFFF6)                 /* Interrupt Request Register 1            */
#define     IRRI1       IRR1_BIT.b1                             /* IRRI1 Interrupt Request Register        */
```

```
#define     PMRB_BIT    (*(struct BIT *)0xFFEE)                 /* Port mode register B                     */
#define     IRQ1        PMRB_BIT.b3                             /* PB3/AN3/IRQ1 pin function switch         */

#pragma interrupt   (irq1int)
/************************************************************************************************************/
/*  Function define                                                                                         */
/************************************************************************************************************/
extern void INIT( void );                                       /* SP Set                                   */
void main( void );
void irq1int( void );

/************************************************************************************************************/
/*  RAM define                                                                                              */
/************************************************************************************************************/
unsigned int    nextmode;                                       /* Motor mode counter                       */

/************************************************************************************************************/
/*  Vector Address                                                                                          */
/************************************************************************************************************/
#pragma section     V1                                          /* Vector Section Set                       */
void (*const VEC_TBL1[])(void) = {                              /* 0x0000 - 0x000F                          */
    INIT                                                        /* 0x0000 Reset Vector                      */
};
#pragma section     V2                                          /* Vector Section Set                       */
void (*const VEC_TBL2[])(void) = {
    irq1int                                                     /* 0x000A IRQ1  Interrupt Vector            */
};

#pragma section                                                 /* P                                        */
/************************************************************************************************************/
/*  Main Program                                                                                            */
/************************************************************************************************************/
void main( void )
{
    unsigned char   pwmu_data,pwml_data;
    unsigned char   tmp;

    set_imask_ccr(1);                                           /* Interrupt Disable                        */

    PWDRL1 = 0x00;                                              /* Set PWM Output Pulse Data Higher         */
    PWDRU1 = 0x00;                                              /* Set PWM Output Pulse Data Lower          */
    PWM1 = 1;                                                   /* Pin function Select PWM1                 */
    PWCR1 = 0xFC;                                               /* Initialize PWM Input Clock               */

    IEG1 = 1;                                                   /* Rising edge of IRQ1                      */
    IRRI1 = 0;                                                  /* Initialize IRRI1                         */
    IEN1 = 1;                                                   /* IRQ1 Interrupt Request Enable            */
    IRQ1 = 1;                                                   /* Pin function Select IRQ1                 */

    nextmode = 0;                                               /* Motor mode counter Clear                 */
    set_imask_ccr(0);                                           /* Interrupt Disable                        */

    while(1);
}
```

```
/*****************************************************************************************************************/
/*  IRQ1 Interrupt                                                                                               */
/*****************************************************************************************************************/
void irq1int( void )
{
    unsigned char tmp;

    IRRI1 = 0;                                          /* Clear IRRI0                          */

    if(nextmode < 5)                                    /* Motor Mode count over 5?             */
        nextmode++;                                     /* Next mode                            */
    else
        nextmode = 0;                                   /* Mode Clear                           */

    switch(nextmode){                                   /* What Motor Mode?                     */
        case 0:
            PWDRL1 = 0x00;                              /* Set PWDRU1, PWDRL1                   */
            PWDRU1 = 0x00;                              /* DC Motor Stop                        */
            break;

        case 1:
        case 5:
            PWDRL1 = 0x00;                              /* Set PWDRU1, PWDRL1                   */
            PWDRU1 = 0x03;                              /* DC Motor 75%Power revolution         */
            break;

        case 2:
        case 4:
            PWDRL1 = 0x80;                              /* Set PWDRU1, PWDRL1                   */
            PWDRU1 = 0x03;                              /* DC Motor 87.5%Power revolution       */
            break;

        case 3:
            PWDRL1 = 0xFF;                              /* Set PWDRU1, PWDRL1                   */
            PWDRU1 = 0x03;                              /* DC Motor Full Power revolution       */
            break;
    }
}
```

Link address specifications

Section Name	Address
CV1	0x0000
CV2	0x000A
P, C	0x0100
B	0xFB80

Revision Record

		Description	
Rev.	Date	Page	Summary
1.00	Dec.19.03	—	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.