RL78/G14 Group

Timer RD Push-Pull Output

R01AN1946EC0101
Rev. 1.01
Feb 25, 2014

Introduction

This application note describes how to use RL78 series MCU, Timer RD's PWM3 mode to implement push-pull output signal or waveform.

Target Device

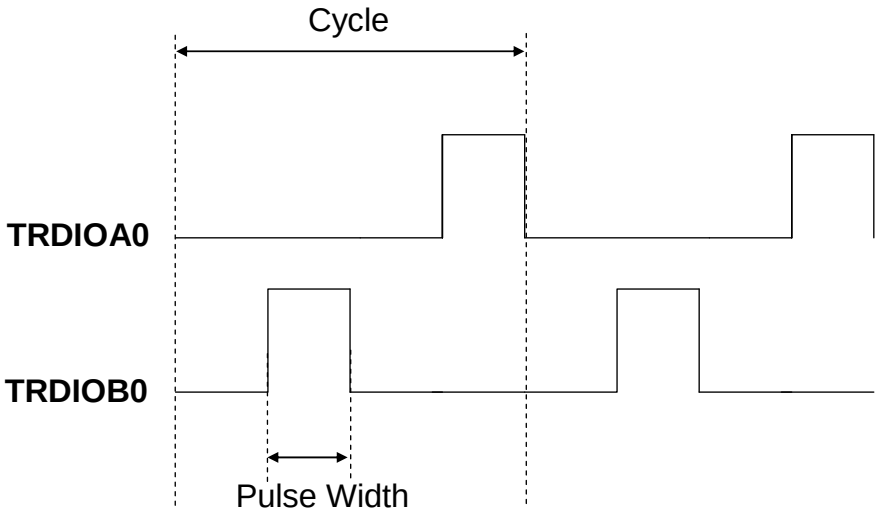
RL78/G14

Content

1. Overview

Push-pull circuit is connected to two different polar transistor output circuit. The push-pull circuit uses two power transistors with same parameters, each transistor responsible for the positive and negative half cycles of waveform, by a push-pull manner presence in the circuit. In operation, only one transistor switch on in two symmetrical transistors, have less conduction loss and high efficiency advantages. Push-pull output either supply current to the load, it can also sink current from the load.

In RL78/G14 MCU, There is a Timer RD, it has an operating mode PWM3. By setting Timer RD's register, the timer output pin TRDIOA0 and TRDIOB0 able to output waveform below. These two waveforms can be used to drive positive and negative half-cycle of transistors.

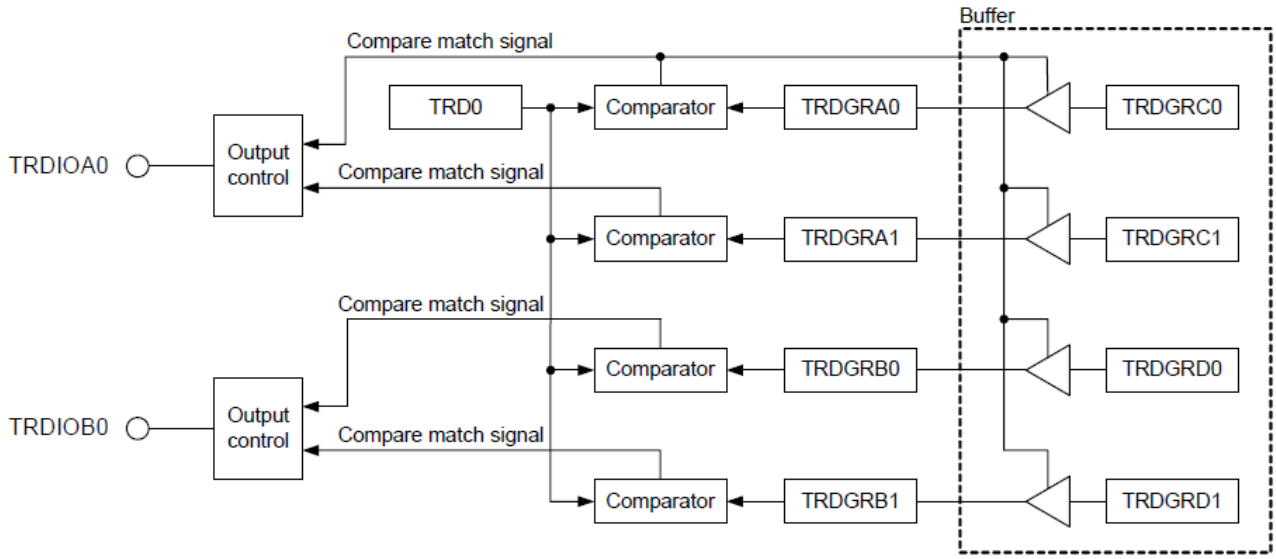


2. Timer RD PWM3 Mode

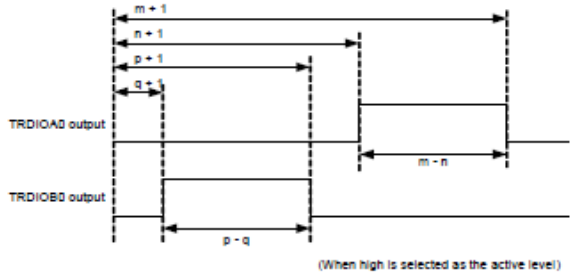
By setting TRDGRA0, TRDGRA1, TRDGRB0 and TRDGRB1, PWM output waveform can be set.

TRDGRC0, TRDGRC1, TRDGRD0 and TRDGRD1 are the buffer registers of TRDGRA0, TRDGRA1, TRDGRB0 and TRDGRB1 respectively. After the value of buffer register is set, corresponding waveform is output in next PWM cycle.

Below is PWM3 mode block diagram.

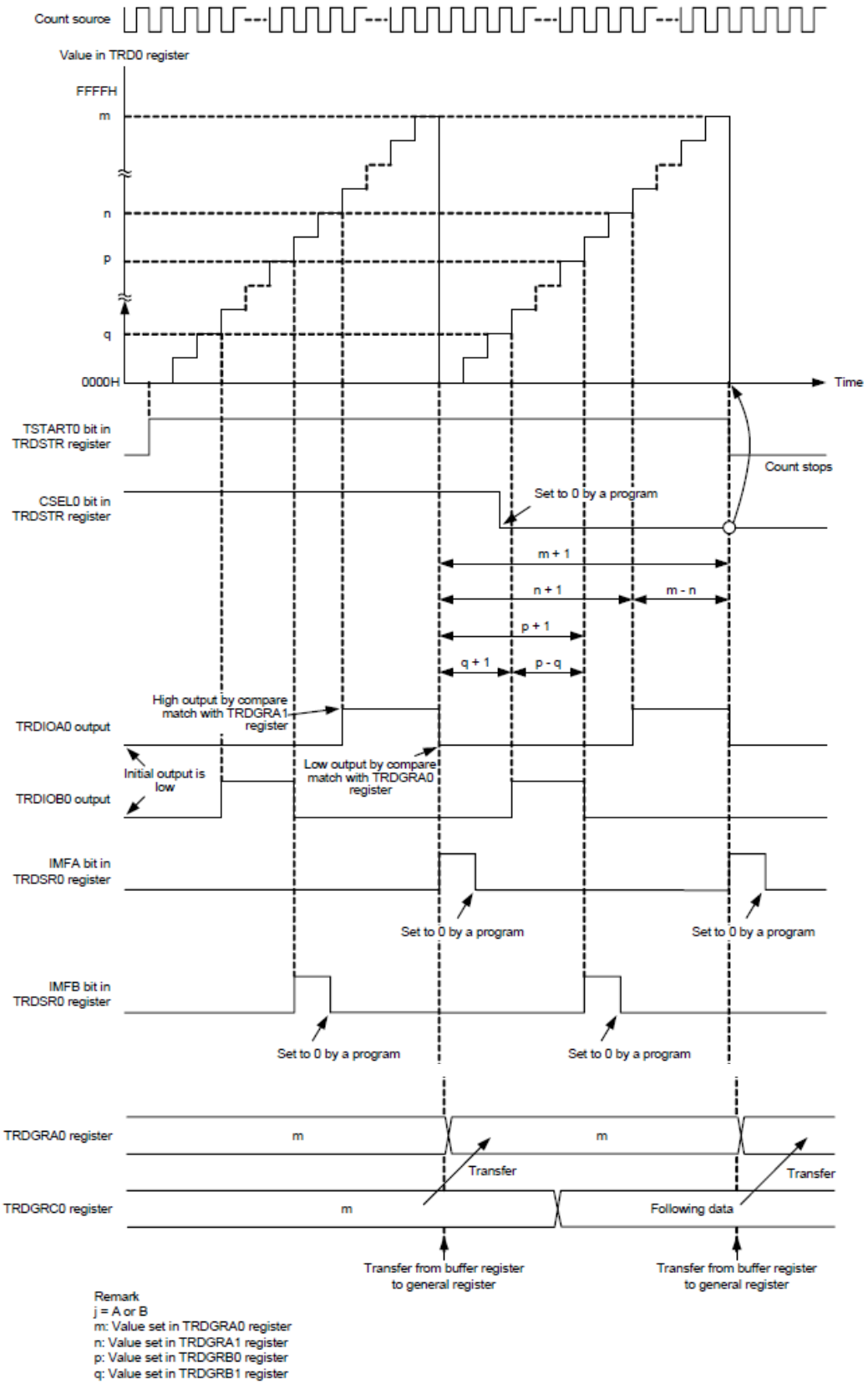


PWM3 Mode Operating specification

Item	Specification
Count sources	f_{HOCO} [Note], f_{CLK}, $f_{CLK}/2$, $f_{CLK}/4$, $f_{CLK}/8$, $f_{CLK}/32$
Count operations	The TRD0 register is incremented (the TRD1 register is not used).
PWM waveform	PWM period: 1/fk × (m + 1) Active level width of TRDIOA0 output: 1/fk × (m - n) Active level width of TRDIOB0 output: 1/fk × (p - q) fk: Frequency of count source m: Value set in the TRDGRA0 register n: Value set in the TRDGRA1 register p: Value set in the TRDGRB0 register q: Value set in the TRDGRB1 register  (When high is selected as the active level)
Count start condition	1 (count starts) is written to the TSTART0 bit in the TRDSTR register.
Count stop conditions	• 0 (count stops) is written to the TSTART0 bit in the TRDSTR register when the CSEL0 bit in the TRDSTR register is set to 1. The PWM output pin holds the output level before the count stops. • When the CSEL0 bit in the TRDSTR register is set to 0, the count stops at compare match with the TRDGRA0 register. The PWM output pin holds the level after output change by compare match.
Interrupt request generation timing	• Compare match (content of the TRDi register matches content of the TRDGRji register) • TRD0 register overflow
TRDIOA0, TRDIOB0 pin function	PWM output
TRDIOA0, TRDIOD0, and TRDIOA1 to TRDIOD1 pin function	I/O port
INTP0 pin function	Pulse output forced cutoff signal input (I/O port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRD0 register.
Write to timer	The value can be written to the TRD0 register.
Selectable functions	• Pulse output forced cutoff signal input (see **8. 3. 1 (4) Pulse Output Forced Cutoff**) • Active level selectable for each pin. • Buffer operation (see **8. 3. 1 (2) Buffer Operation**)

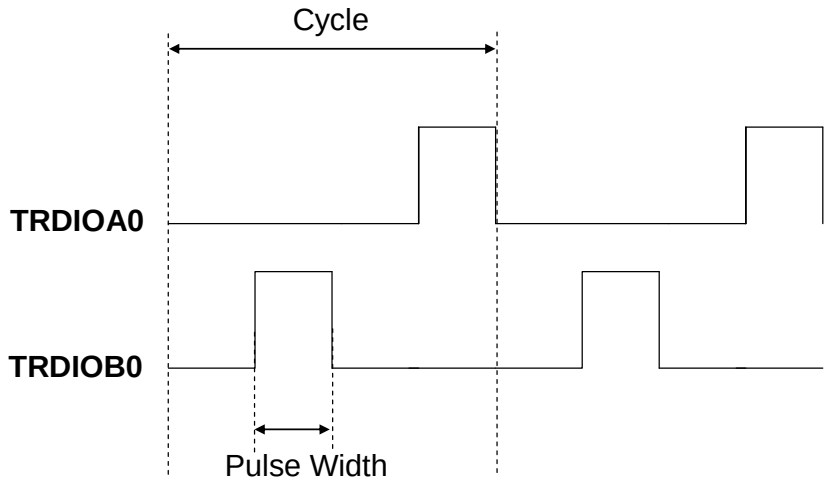
3. Use PWM3 mode to implement push-pull control

PWM3 mode able to output 2-ch PWM waveform at the same PWM frequency period, below is PWM3 operation example.

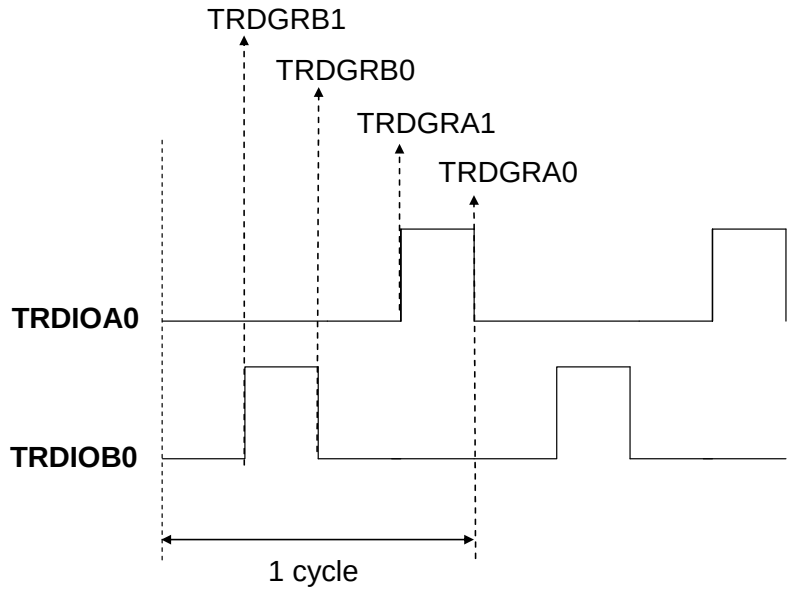


Theory of using PWM3 mode to implement push-pull output

Assume TRDIOA0 and TRDIOB0 output pulse width are the same, in this case, only PWM cycle and pulse width are required to be set.



Through the setting of Timer RD I/O registers TRDGRA0、TRDGRA1、TRDGRB0 and TRDGRB1, PWM cycle and pulse width can be outputted. When Timer RD's counter and Timer RD I/O counter are matched, TRDIOA0 and TRDIOB0 output will be changed accordingly.



PWM Frequency Setting:

Assume MCU internal HOCO is 48MHz，PWM frequency is 33kHz，push-pull pulse width (duty cycle) is 70% on. Frequency and pulse width calculation by below formula.

Cycle = 48000000/33000 = 1454 CPU clocks

Pulse width = 1454/2 *0.7 = 509 CPU clocks

According to PWM Cycle and Pulse Width, use below formula to set Timer RD register value.

TRDGRA0 = Cycle

TRDGRA1 = Cycle – Pulse Width

TRDGRB0 = Cycle/2

TRDGRB1 = Cycle /2 – Pulse Width

Before PWM start to output 在 PWM, buffer register also be set.

TRDGRC0 = Cycle

TRDGRC1 = Cycle – Pulse Width

TRDGRD0 = Cycle/2

TRDGRD1 = Cycle /2 – Pulse Width

4. Sample Program

```
/***********************************************************************************************
* File Name    : r_main.c
* Version      : CodeGenerator for RL78/G14 V2.01.00.02 [09 Aug 2013]
* Device(s)    : R5F104FA
* Tool-Chain   : CA78K0R
* Description  : This file implements main function.
* Creation Date: 6/11/2013
***********************************************************************************************/

/***********************************************************************************************
Pragma directive
***********************************************************************************************/
/* Start user code for pragma. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/***********************************************************************************************
Includes
***********************************************************************************************/
#include "r_cg_macrodriver.h"
#include "r_cg_cgc.h"
#include "r_cg_timer.h"
/* Start user code for include. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
#include "r_cg_userdefine.h"

/***********************************************************************************************
Global variables and functions
***********************************************************************************************/
/* Start user code for global. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
void R_MAIN_UserInit(void);

/***********************************************************************************************
**********************
* Function Name: main
* Description  : This function implements main function.  PWM3 mode is initialized,and output the
PWM waveform for
*                pushpull control
* Arguments    : None
* Return Value : None
***********************************************************************************************/
void main(void)
{
    R_MAIN_UserInit();
    /* Start user code. Do not edit comment generated here */
```

```
    R_TMR_RD0_Start();
    while (1U)
    {
        ;
    }
    /* End user code. Do not edit comment generated here */
}
```

```
/***********************************************************************************************
* File Name    : r_cg_timer.c
* Version      : CodeGenerator for RL78/G14 V2.01.00.02 [09 Aug 2013]
* Device(s)    : R5F104FA
* Tool-Chain   : CA78K0R
* Description  : This file implements device driver for TAU module.
* Creation Date: 6/11/2013
***********************************************************************************************/

/***********************************************************************************************
Pragma directive
***********************************************************************************************/
/* Start user code for pragma. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/***********************************************************************************************
**********************
Includes
***********************************************************************************************
**********************/
#include "r_cg_macrodriver.h"
#include "r_cg_timer.h"
/* Start user code for include. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
#include "r_cg_userdefine.h"

/***********************************************************************************************
Global variables and functions
***********************************************************************************************/
/* Start user code for global. Do not edit comment generated here */
uint16_t g_cycle = 0x5AD;     //PWM cycle
uint16_t g_width = 0x200;     //PWM width
/* End user code. Do not edit comment generated here */

/***********************************************************************************************
* Function Name: R_TMR_RD0_Create
* Description  : This function initializes the TMRD0 and TMRD1 module.
* Arguments    : None
* Return Value : None
***********************************************************************************************/
void R_TMR_RD0_Create(void)
{
    TRD0EN = 1U;    /* enable input clock supply */
    TRDSTR |= _04_TMRD_TRD0_COUNT_CONTINUES | _08_TMRD_TRD1_COUNT_CONTINUES;
    TRDSTR &= (uint8_t)~_03_TRD_COUNT_STATR_INITIAL_VALUE;  /* disable TMRD operation */
    TRDMK0 = 1U;    /* disable TMRD0 interrupt */
    TRDIF0 = 0U;    /* clear TMRD0 interrupt flag */
```

```
    /* Set INTTRD0 low priority */
    TRDPR10 = 1U;
    TRDPR00 = 1U;
    TRDMR = _10_TMRD_TRDGRC0_BUFFER | _20_TMRD_TRDGRD0_BUFFER | _40_TMRD_TRDGRC1_BUFFER |
_80_TMRD_TRDGRD1_BUFFER;
    TRDFCR = _00_TMRD_PWM3_MODE;
    TRDDF0 = _00_TMRD_TRDIOB_FORCEDCUTOFF_DISABLE | _00_TMRD_TRDIOA_FORCEDCUTOFF_DISABLE;
    TRDOER1 = _00_TMRD_TRDIOA0_OUTPUT_ENABLE | _00_TMRD_TRDIOB0_OUTPUT_ENABLE |
_04_TMRD_TRDIOC0_OUTPUT_DISABLE |
              _08_TMRD_TRDIOD0_OUTPUT_DISABLE | _10_TMRD_TRDIOA1_OUTPUT_DISABLE |
_20_TMRD_TRDIOB1_OUTPUT_DISABLE |
              _40_TMRD_TRDIOC1_OUTPUT_DISABLE | _80_TMRD_TRDIOD1_OUTPUT_DISABLE;
    TRDOCR = _00_TMRD_TRDIOA0_INITIAL_OUTPUT_L | _00_TMRD_TRDIOB0_INITIAL_OUTPUT_L;
    TRDCR0 = _00_TMRD_INETNAL_CLOCK_F1_FIH | _20_TMRD_COUNTER_CLEAR_TRDGRA;
    TRDIER0 = _01_TMRD_IMIA_ENABLE | _02_TMRD_IMIB_ENABLE | _00_TMRD_OVIE_DISABLE;
    TRDGRA0 = g_cycle;
    TRDGRB0 = g_cycle / 2;
    TRDGRC0 = g_cycle;
    TRDGRD0 = g_cycle / 2;
    TRDGRA1 = g_cycle - g_width;
    TRDGRB1 = (g_cycle / 2) - g_width;
    TRDGRC1 = g_cycle - g_width;
    TRDGRD1 = (g_cycle / 2) - g_width;
    /* Set TRDIOA0 pin */
    P1 &= 0x7FU;
    PM1 &= 0x7FU;
    /* Set TRDIOB0 pin */
    P1 &= 0xDFU;
    PM1 &= 0xDFU;
}

/***********************************************************************************************
* Function Name: R_TMR_RD0_Start
* Description  : This function starts TMRD0 and TMRD1 counter.
* Arguments    : None
* Return Value : None
***********************************************************************************************/
void R_TMR_RD0_Start(void)
{
    uint8_t trdsr_dummy;

    trdsr_dummy = TRDSR0; /* read TRDSR0 before write 0 */
    TRDSR0 = 0x00U; /* clear TRD0 each interrupt request */
    trdsr_dummy = TRDSR1; /* read TRDSR1 before write 0 */
    TRDSR1 = 0x00U; /* clear TRD1 each interrupt request */
    TRDIF0 = 0U;    /* clear TMRD0 interrupt flag */
    TRDMK0 = 0U;    /* enable TMRD0 interrupt */
```

```
    TRDSTR |= _04_TMRD_TRD0_COUNT_CONTINUES;
    TRDSTR |= _01_TMRD_TRD0_COUNT_START;    /* start TMRD0 counter */
}

/***********************************************************************************************
* Function Name: R_TMR_RD0_Stop
* Description  : This function stops TMRD0 and TMRD1 counter.
* Arguments    : None
* Return Value : None
***********************************************************************************************/
void R_TMR_RD0_Stop(void)
{
    uint8_t trdsr_dummy;

    TRDSTR |= _04_TMRD_TRD0_COUNT_CONTINUES;
    TRDSTR &= (uint8_t)~_01_TMRD_TRD0_COUNT_START;  /* stop TMRD0 counter */
    TRDMK0 = 1U;    /* disable TMRD0 interrupt */
    TRDIF0 = 0U;    /* clear TMRD0 interrupt flag */
    trdsr_dummy = TRDSR0; /* read TRDSR0 before write 0 */
    TRDSR0 = 0x00U; /* clear TRD0 each interrupt request */
    trdsr_dummy = TRDSR1; /* read TRDSR1 before write 0 */
    TRDSR1 = 0x00U; /* clear TRD1 each interrupt request */
}
```

Website and Support

Renesas Electronics Website

http://www.renesas.com/

Inquiries

http://www.renesas.com/contact/

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Jan 25, 2014	—	First release version
1.01	Feb 25, 2014	8~12	Sample code modified

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "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 etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
6. 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.
7. 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 systems manufactured by you.
8. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 LanGao Rd., Putuo District, Shanghai, China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 3.0