[Notes]
Smart Configurator for RH850

R20TS0723EC0100
Rev.1.00
Jul. 01, 2021

Outline

When using Smart Configurator for RH850, note the following points.

1. Notes on using One-Shot Pulse Output, PWM Output, Triangle PWM Output, Triangle PWM Output with Dead Time functions with TAUD1, TAUD2

1. Notes on using One-Shot Pulse Output, PWM Output, Triangle PWM Output, Triangle PWM Output with Dead Time functions with TAUD1, TAUD2

1.1 Applicable Products

Smart Configurator for RH850 V1.2.0 or later version

1.2 Applicable Devices

RH850 family: RH850/U2A group

- RH850/U2A16 (516-pin product)
- RH850/U2A16 (292-pin product)
- RH850/U2A8 (292-pin product)

1.3 Details

When using One-Shot Pulse Output, PWM Output, Triangle PWM Output, Triangle PWM Output with Dead Time functions on the following peripherals, the interrupt cannot stop even if the timer stop function has been called.

- RH850/U2A16 (516-pin product)

  TAUD1, TAUD2
- RH850/U2A16 (292-pin product)

  TAUD1, TAUD2
- RH850/U2A8 (292-pin product)

  TAUD1, TAUD2

1.4 Workaround

User can manually modify the code in the following source file

- Source file: “<*Configuration-name*>.c”.
- Function: “void R_<*Configuration-name*>_disable_interrupt (void)”

Note: If code is generated again, the previous state is restored. Modification is necessary each time you perform code generation.

This is an example of the required modification when <*Configuration-name*> is Config_TAUD1 in the RH850/U2A16 group. User should manually swap the bit of EIMK*m* and EIRF*m*. In the following example, the code in red color is wrong code before modification, while the code in blue color is correct code after modification.

Before modification:

```
void r_Config_TAUD1_disable_interrupt(void)
{
    /* Clear INTTAUD1I0 request and enable operation */
    INTC2.EIC384.BIT.EIRF384 = _INT_PROCESSING_DISABLED;
    INTC2.EIC384.BIT.EIMK384 = _INT_REQUEST_NOT_OCCUR;
    /* Clear INTTAUD1I1 request and enable operation */
    INTC2.EIC385.BIT.EIRF385 = _INT_PROCESSING_DISABLED;
    INTC2.EIC385.BIT.EIMK385 = _INT_REQUEST_NOT_OCCUR;
}
```

After modification:

```
void r_Config_TAUD1_disable_interrupt(void)
{
    /* Clear INTTAUD1I0 request and disable operation */
    INTC2.EIC384.BIT.EIMK384 = _INT_PROCESSING_DISABLED;
    INTC2.EIC384.BIT.EIRF384 = _INT_REQUEST_NOT_OCCUR;
    /* Clear INTTAUD1I1 request and disable operation */
    INTC2.EIC385.BIT.EIMK385 = _INT_PROCESSING_DISABLED;
    INTC2.EIC385.BIT.EIRF385 = _INT_REQUEST_NOT_OCCUR;
}
```

1.5 Schedule for Fixing the Problem

This problem will be fixed in the next version. (Scheduled to be released in Nov 2021.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jul.01.21	-	First edition issued

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

TS Colophon 4.2