[Notes]

AP4 Coding Assistance Tool for RH850

Outline

When using the AP4 coding assistance tool for RH850, note the following point.

1. Interrupt priority settings

1. Interrupt Priority Settings

1.1 Applicable Products

➤ V1.02.00 and later versions of the AP4 coding assistance tool for RH850

1.2 Applicable MCUs

➤ RH850 family: RH850/F1K group

1.3 Details

The code generated when the priority of the interrupt setting of each peripheral function is changed has an error and is always set to "lowest". Therefore, the peripheral functions cannot be executed with the correct interrupt priority.

1.4 Conditions

This problem arises when EWRH (compiler from IAR) or GHSRH (compiler from Green Hills Software) is selected for the compiler to be used.

1.5 Workaround

Modify the interrupt priority in the initialization function void R_xxx_Create(void)[(Note)] of each peripheral function. This modification is required every time code is generated.

Note: xxx represents abbreviation of each peripheral function.

The following is an example of modifying code to set the priority of the communication interrupt (ICCSIG0IC) of the clock synchronous serial interface to "highest".

Modify the priority of ICCSIG0IC in the void R_CSIG0_Create(void) function in the way shown below. The function is in the r_cg_scig.c file. The modification is shown in red.

Before modification:

```
/***********************************************************************
* Function Name: R_CSIG0_Create
* Description  : This function initializes the CSIG0 module.
* Arguments    : None
* Return Value : None
***********************************************************************/
void R_CSIG0_Create(void)
{
    uint32_t tmp_port;

    /* Disable CSIG0 operation */
    CSIG0.CTL0.UINT8 = _CSIG_OPERATION_CLOCK_STOP;
    /* Disable INTCSIG0IC operation and clear request */
    INTC1.ICCSIG0IC.BIT.MKCSIG0IC = _INT_PROCESSING_DISABLED;
    INTC1.ICCSIG0IC.BIT.RFCSIG0IC = _INT_REQUEST_NOT_OCCUR;
    /* Set CSIG0 interrupt(INTCSIG0IC) setting */
    INTC1.ICCSIG0IC.BIT.TBCSIG0IC = _INT_TABLE_VECTOR;
    INTC1.ICCSIG0IC.UINT16 &= _INT_PRIORITY_LOWEST;
..............
}
```

After modification:

```
/***********************************************************************
* Function Name: R_CSIG0_Create
* Description  : This function initializes the CSIG0 module.
* Arguments    : None
* Return Value : None
***********************************************************************/
void R_CSIG0_Create(void)
{
    uint32_t tmp_port;

    /* Disable CSIG0 operation */
    CSIG0.CTL0.UINT8 = _CSIG_OPERATION_CLOCK_STOP;
    /* Disable INTCSIG0IC operation and clear request */
    INTC1.ICCSIG0IC.BIT.MKCSIG0IC = _INT_PROCESSING_DISABLED;
    INTC1.ICCSIG0IC.BIT.RFCSIG0IC = _INT_REQUEST_NOT_OCCUR;
    /* Set CSIG0 interrupt(INTCSIG0IC) setting */
    INTC1.ICCSIG0IC.BIT.TBCSIG0IC = _INT_TABLE_VECTOR;
    INTC1.ICCSIG0IC.UINT16 &= _INT_PRIORITY_HIGHEST;
..............
}
```

1.6   Schedule for Fixing the Problem

This problem will be fixed in the next version. The next version will be available in July 2018.

Revision History

Rev.	Date	Page	Summary
			Description
1.00	Feb. 16, 2018	-	First edition issued

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061 Japan

Renesas Electronics Corporation

■Inquiry

https://www.renesas.com/contact/

All trademarks and registered trademarks are the property of their respective owners.