[Notes] CS+ Code Generator for RX, e^2 studio Code Generator Plug-in, AP4 Coding Assistance Tool for RX

R20TS0283EJ0100
Rev.1.00
Mar. 1, 2018

Outline

When using the products in the title, note the following point.

1. Technical update " Usage Note on the Option Function Select Register 1 (OFS1)"

1. Technical Update " Usage Note on the Option Function Select Register 1 (OFS1)"

1.1 Applicable Products

- V1.11.00 and later versions of CS+ Code Generator for RX
- V5.2.0 of e^2 studio (V2.5.0 of the Code Generator plug-in) and later versions
- V1.10.00 and later versions of AP4 Coding Assistance Tool for RX

1.2 Applicable MCUs

- RX family: RX110, RX111, and RX113 groups

1.3 Details

The technical update "Usage Note on the Option Function Select Register 1 (OFS1)(Note)" indicates notes concerning the voltage detection circuit (LVD). If the startup power monitor 1 reset enable bit in the option function select register 1 (OFS1.STUPLVD1REN bit) is set to 0, and the voltage monitoring 1 reset is enabled after release, voltage monitoring 1 reset may not occur after a reset by the independent watchdog timer (IWDT).
As a countermeasure against this precaution, the technical update has noted that "When the voltage monitoring 1 reset by the option-setting memory is automatically started with the use of the IWDT reset, enable the voltage monitoring 1 reset by a program immediately after a reset." However, when it takes a long time to enable the voltage monitoring 1 reset, voltage monitoring may not operate properly. In this case, the source code generated by the coding assistance tool described in the workaround below must be used.

Note: Technical Update "Usage Note on the Option Function Select Register 1 (OFS1)"
https://www.renesas.com/search/keyword-search.html#genre=document&q=tnrxa153

1.4 Workaround

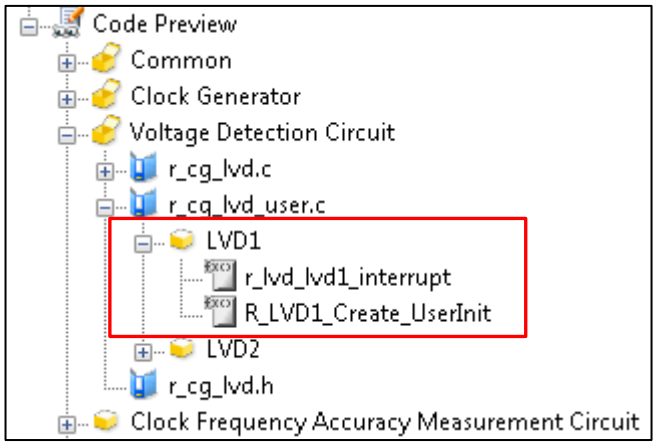
Change the execution sequence of the void R_LVDx_Start(void) function in r_cg_lvd.c and modify the function so that the voltage monitoring 1 reset is enabled immediately after a reset by method A or B, as follows.

A. Adding the user initialization function

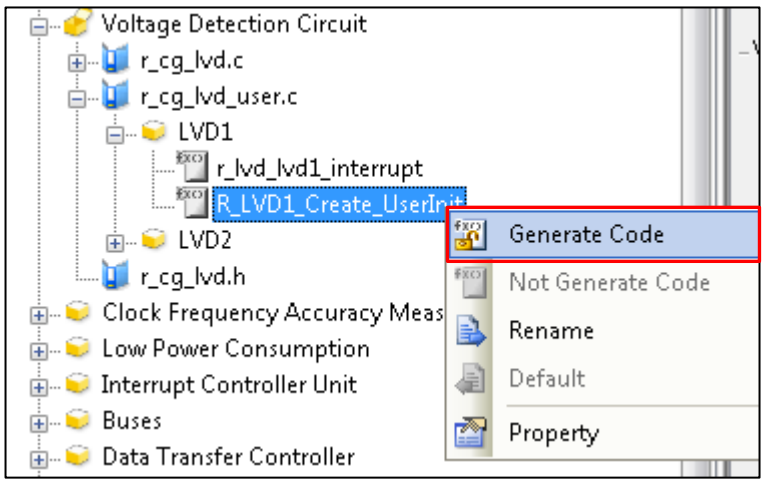
(1) Configure the voltage detection circuit (LVDn)(Note). Note: n = 1 or 2

The following explains a case where LVD1 is set:

(2) Select a code preview from the project tree and open LVD1 in r_cg_lvd_user.c.

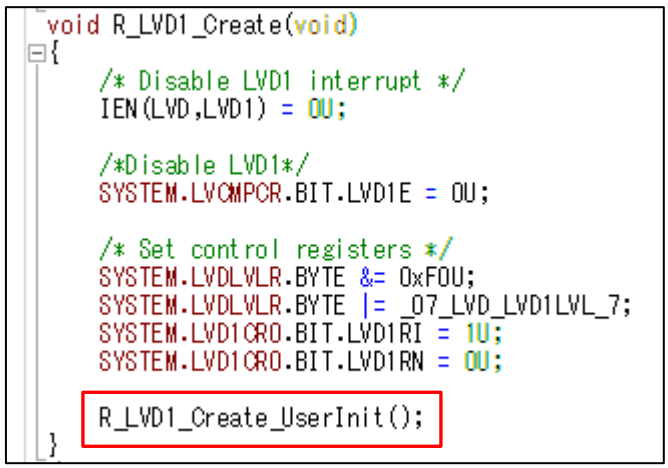


(3) R_LVD1_Create_UserInit is not generated by default. Therefore, set it to be generated.



(4) Execute code generation.

(5) R_LVD1_Create_UserInit() is added to the R_LVD1_Create(void) function in r_cg_lvd.c.



```
void R_LVD1_Create(void)
{
    /* Disable LVD1 interrupt */
    IEN(LVD,LVD1) = 0U;

    /*Disable LVD1*/
    SYSTEM.LVCMPCR.BIT.LVD1E = 0U;

    /* Set control registers */
    SYSTEM.LVDLVLR.BYTE &= 0xF0U;
    SYSTEM.LVDLVLR.BYTE |= _07_LVD_LVD1LVL_7;
    SYSTEM.LVD1CR0.BIT.LVD1RI = 1U;
    SYSTEM.LVD1CR0.BIT.LVD1RN = 0U;

    R_LVD1_Create_UserInit();
}
```

(6) Add R_LVD1_Start() to the R_LVD1_Create_UserInit(void) function in r_cg_lvd_user.c.

r_cg_cmt_user.c

```
void R_LVD1_Create_UserInit(void)
{
    /* Start user code. Do not edit comment generated here */
        R_LVD1_Start();
    /* End user code. Do not edit comment generated here */
}
```

After completing modification (1) to (6), R_LVD1_Create_UserInit() is called when R_LVD1_Create() is executed. R_LVD1_Start() is executed in R_LVD1_Create_UserInit(), and accordingly the voltage detection circuit (LVD1) is started.

Even when code generation is executed after performing modification (1) to (6), the modification is not deleted.

The whole function generated after the settings is shown below:

```
void R_LVD1_Create(void)
{
    /* Disable LVD1 interrupt */
    IEN(LVD,LVD1) = 0U;
    /*Disable LVD1*/
    SYSTEM.LVCMPCR.BIT.LVD1E = 0U;
    /* Set control registers */
    SYSTEM.LVDLVLR.BYTE &= 0xF0U;
    SYSTEM.LVDLVLR.BYTE |= _07_LVD_LVD1LVL_7;
    SYSTEM.LVD1CR0.BIT.LVD1RI = 1U;
    SYSTEM.LVD1CR0.BIT.LVD1RN = 0U;

    R_LVD1_Create_UserInit();
}

void R_LVD1_Create_UserInit(void)
{
    /* Start user code. Do not edit comment generated here */
        R_LVD1_Start();
    /* End user code. Do not edit comment generated here */
}

void R_LVD1_Start(void)
{
    uint16_t protect_dummy = (uint16_t)(SYSTEM.PRCR.WORD & 0x000FU);
    uint16_t w_count;
    /* Disable protect bit */
    SYSTEM.PRCR.WORD = 0xA508U;
    SYSTEM.LVD1CR0.BIT.LVD1RIE = 1U;
    SYSTEM.LVCMPCR.BIT.LVD1E = 1U;
    /* Wait for LVD voltage detection to start */
    for (w_count = 0U; w_count < _F1_LVD1_STABLE_WAIT_TIME; w_count++)
    {
        nop();
    }
    SYSTEM.LVD1CR0.BIT.LVD1CMPE = 1U;
    /* Restore the previous state of the protect register */
    SYSTEM.PRCR.WORD = (uint16_t)(0xA500U | protect_dummy);
}
```

B. Enabling the voltage detection circuit (LVD) before setting the clock of the MCU

(1) Configure the voltage detection circuit (LVDn)(Note). Note: n = 1 or 2

The following explains a case where LVD1 is set:

(2) Execute code generation.

(3) Edit the R_systeminit(void) function in r_cg_hardware_setup.c to change the position where the R_LVD1_Create() function is called. Additionally, start the voltage detection circuit (LVD1).

r_cg_hardware_setup.c

Before modification:

```
void R_Systeminit(void)
{
    /* Enable writing to registers related to operating modes, LPC, CGC and
software reset */
    SYSTEM.PRCR.WORD = 0xA50FU;
    Omitted
    /* Set peripheral settings */
    R_CGC_Create();
    R_LVD1_Create();
    Omitted
    /* Enable protection */
    SYSTEM.PRCR.WORD = 0xA500U;
}
```

After modification: The red text below indicates the modification.

```
void R_Systeminit(void)
{
    /* Enable writing to registers related to operating modes, LPC, CGC and
software reset */
    SYSTEM.PRCR.WORD = 0xA50FU;
    Omitted
    /* Set peripheral settings */
    R_LVD1_Create();
    R_LVD1_Start();
    R_CGC_Create();
    Omitted
    /* Enable protection */
    SYSTEM.PRCR.WORD = 0xA500U;
}
```

(4) Edit the R_LVD1_Start(void) function in r_cg_lvd.c to change the waiting time for starting LVD.

r_cg_hardware_setup.c
Before modification: The value of _781_LVD1_STABLE_WAIT_TIME differs depending on the settings of the clock generation circuit.

```
void R_LVD1_Start(void)
{
    Omitted
    /* Wait for LVD voltage detection to start */
    for (w_count = 0U; w_count < _781_LVD1_STABLE_WAIT_TIME; w_count++)
    {
        nop();
    }
    Omitted
}
```

After modification: The red text below indicates the modification.

```
void R_LVD1_Start(void)
{
    Omitted
    /* Wait for LVD voltage detection to start */
    for (w_count = 0U; w_count < 50; w_count++)
    {
        nop();
    }
    Omitted
 }
```

Modification (3) and (4) are required every time code is generated.

1.5 Schedule for Fixing the Problem

This problem will be fixed in the next version.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Mar. 1, 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/