

【注意事項】**CS+用 RX コード生成,
e² studio Code Generator プラグイン,
RX コード生成支援ツール AP4****概要**

タイトルに記載している製品の使用上の注意事項を連絡します。

1. テクニカルアップデート「オプション機能選択レジスタ 1 (OFS1) 使用時の注意事項」に関する注意事項

1. テクニカルアップデート「オプション機能選択レジスタ 1 (OFS1) 使用時の注意事項」に関する注意事項**1.1 該当製品**

- CS+用 RX コード生成 V1.11.00 以降
- e² studio V5.2.0 (Code Generator プラグイン V2.5.0) 以降
- RX コード生成支援ツール AP4 V1.10.00 以降

1.2 該当 MCU

- RX ファミリ : RX110、111 および 113 グループ

1.3 内容

テクニカルアップデート「オプション機能選択レジスタ 1 (OFS1) 使用時の注意事項^(注)」は電圧検出回路 (LVD)に関する注意事項です。オプション機能選択レジスタ 1 の起動時電源監視 1 リセット有効ビット (OFS1.STUPLVD1REN ビット)を“0”にして、リセット解除後から電圧監視 1 リセットを有効にしている場合、独立ウォッチドッグタイマによるリセットが発生した後、電圧監視 1 リセットが動作しないことがあります。

この注意事項の対策には、「オプション設定メモリを使用した電圧監視 1 リセットの自動起動を、IWDT リセットと同時に使用する場合、リセット直後にプログラムでも電圧監視 1 リセットを有効にしてください。」と記載しています。ただし、電圧監視 1 リセットを有効にするまでの時間が長いと、電圧監視が正常に動作しない場合があります。その場合は、以下回避策に記載のコード生成支援ツールで生成したソースコードで対策をとる必要があります。

注 : テクニカルアップデート「オプション機能選択レジスタ 1 (OFS1) 使用時の注意事項」

<https://www.renesas.com/search/keyword-search.html#genre=document&q=tnrxa153>

1.4 回避策

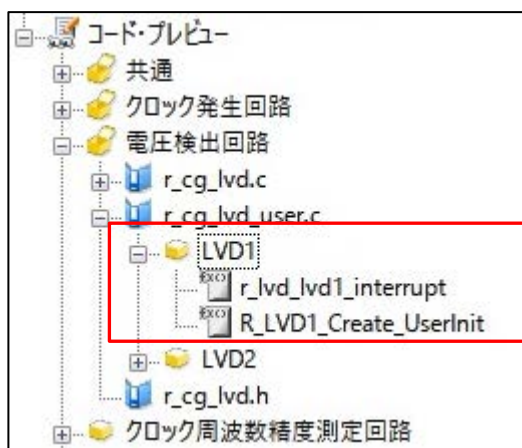
r_cg_lvd.c にある関数 void R_LVDx_Start(void)の実行順序を入れ替え、リセット直後に電圧監視 1 リセットを有効にするように、以下の A、B いずれかの方法で修正してください。

A. ユーザ初期化関数を追加する方法

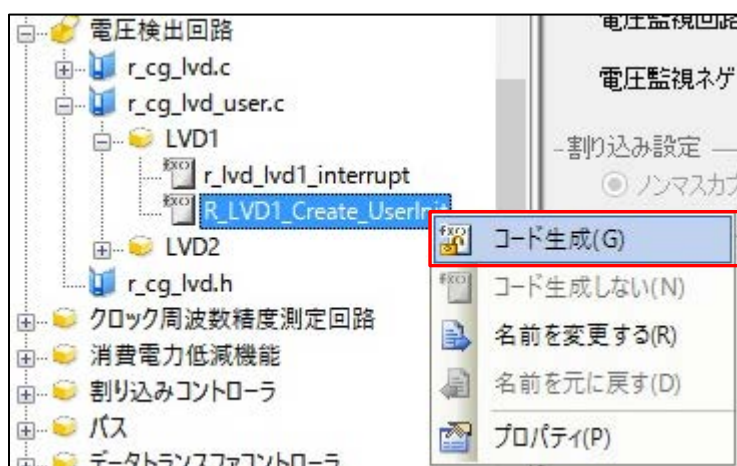
(1) 電圧検出回路(LVDn) ^(注) を設定してください。注：n=1 or 2

以降、LVD1 を設定した場合

(2) プロジェクト・ツリーからコード・プレビューを選び、r_cg_lvd_user.c LVD1 を開いてください。



(3) R_LVD1_Create_UserInit はデフォルトの生成関数ではないので、生成するように設定してください。



(4) コード生成を実行してください。

(5) r_cg_lvd.c の R_LVD1_Create(void)関数に R_LVD1_Create_UserInit() が追加されます。

```

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

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

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

    R_LVD1_Create_UserInit();
}
    
```

(6) r_cg_lvd_user.c の R_LVD1_Create_UserInit(void)関数に R_LVD1_Start() を追加してください。

r_cg_lvd_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 */
}
    
```

(1)~(6)の修正により、R_LVD1_Create() が実行されると、R_LVD1_Create_UserInit()が呼ばれます。
R_LVD1_Create_UserInit()では、R_LVD1_Start()が実行されるので、電圧検出回路(LVD1)が開始されます。

※(1)~(6)の修正を行った後にコード生成を実行しても修正内容は消えません。

以下に設定後に生成される関数全体を示します。

```
void R_LVD1_Create(void)
{
    /* Disable LVD1 interrupt */
    IEN(LVD,LVD1) = 0U;
    /*Disable LVD1*/
    SYSTEM.LVCMPCR.BIT.LVD1E = 0U;
    /* Set control registers */
    SYSTEM.LVDLVLRL.BYTE &= 0xF0U;
    SYSTEM.LVDLVLRL.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. マイコンのクロック設定前に電圧検出回路(LVD)を有効にする方法

(1) 電圧検出回路(LVDn) ^(注) を設定してください。注：n=1 or 2

以降、LVD1を設定した場合

(2) コード生成を実行してください。

(3) r_cg_hardware_setup.cのR_Systeminit(void)関数を編集し、R_LVD1_Create()関数の呼び出し位置を変更してください。さらに電圧検出回路(LVD1)を開始してください。

r_cg_hardware_setup.c

修正前：

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

修正後：下記の赤文字が修正箇所です。

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

(4) r_cg_lvd.c の R_LVD1_Start(void)関数を編集し、LVD 開始の待ち時間を変更します。

r_cg_hardware_setup.c

修正前：_781_LVD1_STABLE_WAIT_TIMEは、クロック発生回路の設定によって値が異なります。

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

修正後：下記の赤文字が修正箇所です。

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

※コード生成を実行するたびに(3)、(4)の修正が必要です。

1.5 恒久対策

次期バージョンで改修予定です。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2018.03.01	-	新規発行

ルネサスエレクトロニクス株式会社

〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)

■総合お問い合わせ先

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

本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。

ニュース本文中の URL を予告なしに変更または中止することがありますので、あらかじめご承知ください。

すべての商標および登録商標は、それぞれの所有者に帰属します。