RENESAS Tool News

RENESAS TOOL NEWS on June 1, 2012: 120601/tn9

Note on Using RX210 Group Renesas Peripheral Driver Library and Peripheral Driver Generator V2 --With Using SCI5, SCI6, and SCI12 of RX210 Group MCUs--

When using RX210 Group Renesas Peripheral Driver Library and Peripheral Driver Generator V2, take note of the following problem:

- With using SCI5, SCI6, and SCI12 of the RX210 group of MCUs

---

1. Products and Versions Concerned

- RX210 Group Renesas Peripheral Driver Library V.1.01
- Peripheral Driver Generator V.2.03 and later

2. Description

If SCIn (n = 5, 6, or 12) is set in the Asynchronous mode by using the R_SCI_Create function of RX210 Group Renesas Peripheral Driver Library or the R_PG_SCI_Set_Cn function generated by Peripheral Driver Generator V2, and the TMR clock input is selected as a communication clock, the initial setting of SCIn cannot be made, and the function returns a value of "false".

2.1 Conditions

This problem arises in each of the following conditions:

(1) In Renesas Peripheral Driver Library

SCIn (n = 5, 6, or 12) is set in the Asynchronous mode by using the R_SCI_Create function, and the TMR clock input is selected as a communication clock.

(2) In Peripheral Driver Generator

A call is made to the R_PG_SCI_Set_Cn function (n = 5, 6, or 12) that is generated by selecting the Asynchronous mode from the Mode list and the TMR clock from the transfer clock list.

2.2 Examples

(1) In Renesas Peripheral Driver Library

A call is made to the R_SCI_Create function, to which PDL_SCI_ASYNC and PDL_SCI_CLK_TMR are passed as arguments.

```
-------------------------------------------------------
// SCI5 is set in the Asynchronous mode, and TMR clock selected.
  R_SCI_Create(
    5,
    PDL_SCI_ASYNC | PDL_SCI_CLK_TMR | PDL_SCI_TX_DISCONNECTED |
    PDL_SCI_RX_CONNECTED,
 . . . . . .
  );
-------------------------------------------------------
```

(2) In Peripheral Driver Generator

```
-------------------------------------------------------
  void function(void)
  {
     // SCI5 is set in the Asynchronous mode, and TMR clock selected.
       R_PG_SCI_Set_C5();
  }
-------------------------------------------------------
```

3. Workaround

(1) For Renesas Peripheral Driver Library

If Condition (1) in Section 2.1 is satisfied, assign SCKn of SCIn to any pin before calling the R_SCI_Create function.

Example where SCK5 is assigned to PA1:

```
-----------------------------------------------
/*** Part 1 of Workaround Started ***/
  // Remove protection of Pin function control register.
  MPC.PWPR.BIT.B0WI = 0;
  MPC.PWPR.BIT.PFSWE = 1;

  // Save value of PSEL bit of current PA1 on variable.
  tmp_buf = MPC.PA1PFS.BIT.PSEL;

  // Assign SCK5 to PA1 before calling R_SCI_Create.
```

```
  MPC.PA1PFS.BIT.PSEL = 0xA;
/*** Part 1 of Workaround Ended ***/

  // Set SCI5 in the Asynchronous mode, and TMR clock selected.
  R_SCI_Create(
    5,
    PDL_SCI_ASYNC | PDL_SCI_CLK_TMR | PDL_SCI_TX_DISCONNECTED |
    PDL_SCI_RX_CONNECTED,
. . . . . .
  );

/*** Part 2 of Workaround Started ***/
  // Restore value of PA1 (SCK5 pin) in order not to use SCK5 pin
     from now on.
  MPC.PA1PFS.BIT.PSEL  = tmp_buf;

  // Set protection of Pin function control register.
  MPC.PWPR.BIT.PFSWE = 0;
  MPC.PWPR.BIT.B0WI = 1;
/*** Part 2 of Workaround Ended ***/
-----------------------------------------------
```

(2) For Peripheral Driver Generator

If SCIn (n = 5, 6, or 12) is set in the Asynchronous mode and the TMR clock input is selected, assign SCKn of SCIn to any pin before calling the R_PG_SCI_Set_C5() function.

Example where SCK5 is assigned to PA1:

```
-----------------------------------------------
/*** Part 1 of Workaround Started ***/
  // Remove protection of Pin function control register.
  MPC.PWPR.BIT.B0WI = 0;
  MPC.PWPR.BIT.PFSWE = 1;

  // Save value of PSEL bit of current PA1 on variable.
  tmp_buf = MPC.PA1PFS.BIT.PSEL;

  // Assign SCK5 to PA1 before calling R_PG_SCI_Set_C5()
  MPC.PA1PFS.BIT.PSEL = 0xA;
/*** Part 1 of Workaround Ended ***/

  // Set SCI5 in Asynchronous mode, and TMR clock selected.
  R_PG_SCI_Set_C5();

/*** Part 2 of Workaround Started ***/
```

```
  // Restore value of PA1 (SCK5 pin) in order not to use SCK5 pin
     from now on.
  MPC.PA1PFS.BIT.PSEL  = tmp_buf;

  // Set protection of Pin function control register.
  MPC.PWPR.BIT.PFSWE = 0;
  MPC.PWPR.BIT.B0WI = 1;
/*** Part 2 of Workaround Ended ***/
-----------------------------------------------
```

4. Schedule of Fixing Problem

We are going to fix this problem at a later revision of the product.
