[Notes]
Smart Configurator for RH850

R20TS0463EJ0100
Rev.1.00
Aug. 01, 2019

Outline

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

1. When using the input pulse interval measurement function
2. When using the Clocked Serial Interface in Master mode

1. When Using the Input Pulse Interval Measurement Function

1.1 Applicable Products

Smart Configurator for RH850 V1.0.0 or later

1.2 Applicable Devices

RH850 family: RH850/F1KM group

- RH850/F1KM-S1 group: 80-pin and 100-pin products
- RH850/F1KM-S4 group: 100-pin, 144-pin, 176-pin and 233-pin products

1.3 Details

When using the input pulse interval measurement function on the following timer array units, the TAUB*n* channel mode user register (TAUB*n*CMUR*m*) (Note) is generated with an incorrect name (TAUB*n*CMOR*m*) (Note) in the code and correct values cannot be set to the registers.

Note: *n* = unit number, *m* = channel number

- RH850/F1KM-S1 group: 80-pin and 100-pin products

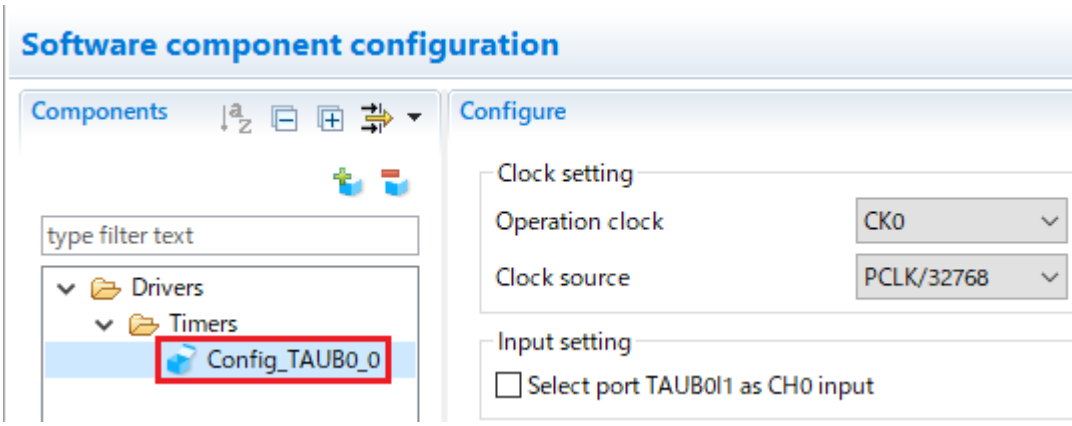
  TAUB0 (Channel 0 to 15)
- RH850/F1KM-S4 group: 100-pin and 144-pin products

  TAUB0 (Channel 0 to 15)
- RH850/F1KM-S4 group: 176-pin and 233-pin products

  TAUB0 (Channel 0 to 15) and TAUB1(Channel 0 to 15)

- Example of when using the input pulse interval measurement function on the TAUB0 channel 0

  Below is an example of when <configuration-name> is Config_TAUB0_0 (default).



The following code is generated in the void R_Config_TAUB0_0_Create(void) function of Config_TAUB0_0.c, and no value is set to the TAUB0.CMUR0 register.

Incorrect code example

```
void R_Config_TAUB0_0_Create(void)
{
    ...
    /* Set compare match register */
    TAUB0.CMOR0 = _TAUB_INPUT_EDGE_FALLING;
    ...
}
```

1.4 Workaround

Correct the generated code in the following source file for the input pulse interval measurement function(Note).

Source file “<configuration-name>.c” function:
“void R_<configuration-name>_Create(void)”

Note: When code is generated again, generated code returns to the state before correction. Therefore, correct the source file each time you generate code.

The following is a modification example of when <configuration-name> is Config_TAUB0_0 (default) in RH850/F1KM. The modification is shown in red.

Before modification

```
void R_Config_TAUB0_0_Create(void)
{
    ...
    /* Set compare match register */
    TAUBn.CMORm = _TAUB_INPUT_EDGE_FALLING;
    ...
}
```

n = unit number, *m* = channel number

After modification

```
void R_Config_TAUB0_0_Create(void)
{
    ...
    /* Set compare match register */
    TAUBn.CMURm = _TAUB_INPUT_EDGE_FALLING;
    ...
}
```

n = unit number, *m* = channel number

1.5 Schedule for Fixing the Problem

This problem will be fixed in the next version. (Scheduled to be released in January 2020.)

2. When Using the Clocked Serial Interface in Master Mode

2.1 Applicable Products

Smart Configurator for RH850 V1.0.0 or later

2.2 Applicable Devices

RH850 family: RH850/F1KM group

- RH850/F1KM-S1 group: 48-pin, 64-pin, 80-pin and 100-pin products
- RH850/F1KM-S4 group: 100-pin, 144-pin, 176-pin and 233-pin products

2.3 Details

When using the following clocked serial interface in Master mode, an incorrect macro definition is generated, and correct values cannot be set in the CSIG*n* control register 2(CSIG*n*CTL2) (Note) and CSIH*n* control register 2(CSIH*n*CTL2) (Note).

Note: *n* = unit number

- RH850/F1KM-S1 group: 48-pin and 64-pin products

  CSIG0 and CSIH0
- RH850/F1KM-S1 group: 80-pin products

  CSIG0 and CSIH0 to 2
- RH850/F1KM-S1 group: 100-pin products

  CSIG0 and CSIH0 to 3
- RH850/F1KM-S4 group: 100-pin products

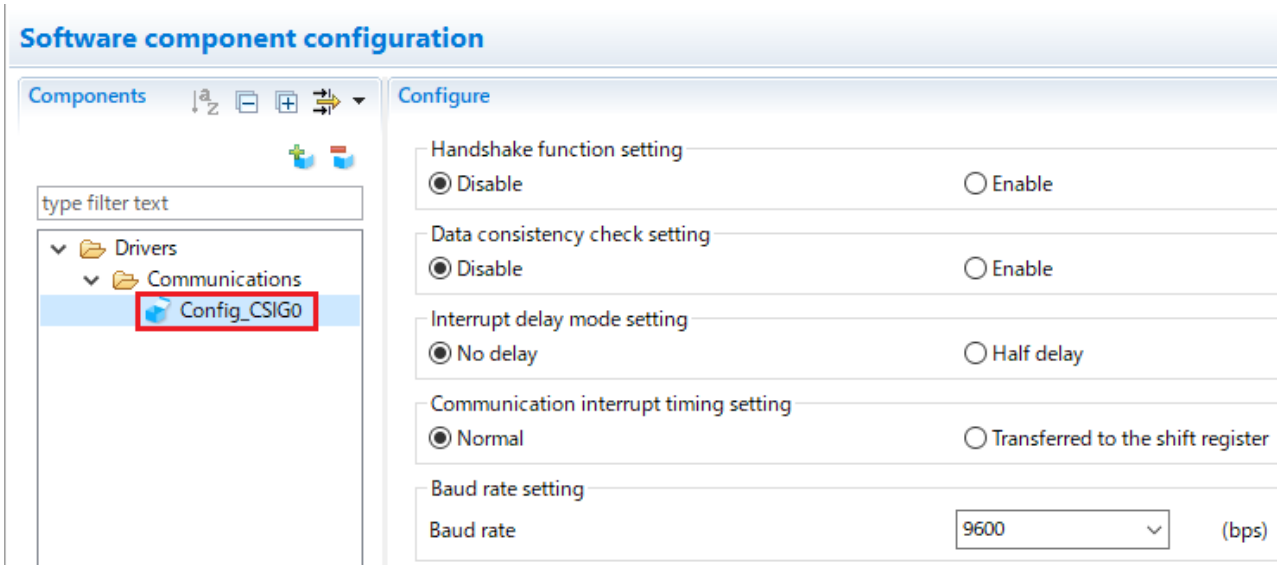
  CSIG0 and CSIH0 to 3
- RH850/F1KM-S4 group: 144-pin products

  CSIG0 to 1 and CSIH0 to 3
- RH850/F1KM-S4 group: 176-pin and 233-pin products

  CSIG0 to 3 and CSIH0 to 3

- Example of when using CSIG0.

  Below is an example of when <configuration-name> is Config_CSIG0 (default).



An incorrect value is generated for _CSIG0_SELECT_BASIC_CLOCK in the macro definition of Config_CSIG0.h.

Incorrect code example

```
#define _CSIG0_SELECT_BASIC_CLOCK       (0x0002U) /* Selects the basic clock */
```

2.4 Workaround

Fix the macro definition manually in the following source file for the Clocked Serial Interface (Note).

Note: When code is generated again, generated code returns to the state before correction. Therefore, correct the source file each time you generate code.

Below is an example of modification. The modification is shown in red.

- For CSIG*n*

Source file “<configuration name>.h” macro definition: _CSIG*n*_SELECT_BASIC_CLOCK
n = unit number

Before modification

```
#define _CSIGn_SELECT_BASIC_CLOCK      (0x????U) /* Selects the basic clock */
```

n = unit number, ???? varies depending on the setting.

After modification

```
#define _CSIGn_SELECT_BASIC_CLOCK      (0x????U <<13U) /* Selects the basic clock */
```

n = unit number, ???? varies depending on the setting.

- For CSIH*n*

Source file “<configuration name>.h” macro definition: _CSIG*n*_SELECT_BASIC_CLOCK
n = unit number

Before modification

```
#define _CSIHn_SELECT_BASIC_CLOCK      (0x????U) /* Selects the basic clock */
```

n = unit number, ???? varies depending on the setting.

After modification

```
#define _CSIHn_SELECT_BASIC_CLOCK      (0x????U <<13U) /* Selects the basic clock */
```

n = unit number, ???? varies depending on the setting.

2.5 Schedule for Fixing the Problem

This problem will be fixed in the next version. (Scheduled to be released in January 2020.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Aug.01.19	-	First edition issued

Corporate Headquarters

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

www.renesas.com

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/

Trademarks

TS Colophon 4.0