RENESAS Tool News

RENESAS TOOL NEWS June 1, 2012: 120601/tn10

Note on Using RX630 Group Renesas Peripheral Driver Library and Peripheral Driver Generator V2 --With Using IEBus of RX630 Group MCUs--

When using RX630 Group Renesas Peripheral Driver Library and Peripheral Driver Generator V2, take note of the following problem:

- With using IEBus of the RX630 group of MCUs

---

1. Products and Versions Concerned

- RX630 Group Renesas Peripheral Driver Library V.1.00
- Peripheral Driver Generator V.2.04 and later

2. Description

In the codes created by using the products concerned,
if a call is made to any of the following callback function after
the data transmission/reception by using IEBus has been completed, you
must clear the IEBus Transmission/Reception Status register.

(1) In the Renesas Peripheral Driver Library concerned, any of the following callback functions:
- R_IEB_MasterSend
- R_IEB_MasterReceive
- R_IEB_SlaveMonitor

(2) In the Peripheral Driver Generator concerned, the callback function of IEB whose names are input on its GUI.

If you do not clear the IEBus Transmission/Reception Status register,
the IEBus interrupt is repetitively generated, and each time a call is
made to a callback function until you clear the register.

2.1 Conditions

(1) In Renesas Peripheral Driver Library

This problem arises if any of the following functions takes a callback function as an argument:

- R_IEB_MasterSend
- R_IEB_MasterReceive
- R_IEB_SlaveMonitor

(2) In Peripheral Driver Generator

This problem arises if any of the following conditions is satisfied:

- "Notify the transmission completion of all data or an error detection by function call"is selected from the Master transmission method list of IEB.
- "Notify the reception completion of all data or an error detection by function call"is selected from the Master reception method list of IEB.
- "Notify the transmission completion of all data, a slave read request, or an error detection by function call"is selected from the Slave monitor method list of IEB.

2.2 Examples

(1) In Renesas Peripheral Driver Library

```
------------------------------------------------------------------
// Send data is set in slave.
R_IEB_SlaveWrite(
   0,
   iebus_data,
   iebus_data_length
);

// Callback function is passed to R_IEB_SlaveMonitor as argument.
/* Monitor channel 0, using polling */
R_IEB_SlaveMonitor(
   0,
   iebus_data,
   &iebus_data_length,
   IEB_CALL_BACK_FUNC
);

// Status flag not cleared when call is made to callback function.
Void IEB_CALL_BACK_FUNC(){

}
------------------------------------------------------------------
```

(2) In Peripheral Driver Generator

(a) Data of IEB is entered as follows:

- "Notify the transmission completion of all data or an error detection by function call"is selected from the Master reception method list.
- Ieb0MasterTrFunc is typed in the Notification function name text box for master transmission.

(b) Then the following program is executed:

```
-----------------------------------------------------------------
 void SetupFunction(void)
 {
    uint8_t tx[]="ABCDEFG";
    R_PG_IEB_Set_C0();    // IEB is set.
    R_PG_IEB_MasterSendData_C0( 1, tx, 5);
                                // Master transmission
 }

 void Ieb0MasterTrFunc(void){
    Function();   // Processing after transmission completed;
                // Transmission Status register not cleared.
}
 -----------------------------------------------------------------
```

3. Workaround

Within the callback function, do the following by using the R_IEB_GetStatus function:

(1) Clearing the Transmission flag

- Make a call to the R_IEB_GetStatus function, and check that the Transmission flag (bit 5 of the 3rd argument) is set to 1 (transmission completed).
- If so, do the following:

```
IEB.IETSR.BYTE = 0x6Fu;    // Transmission Status flag cleared.
```

(2) Clearing the Reception flag

- Make a call to the R_IEB_GetStatus function, and check that the Transmission flag (bit 5 of the 3rd argument) is set to 1 (transmission completed).
- If so, do the following:

```
IEB.IERSR.BYTE = 0xFFu;    // Reception Status flag cleared.
```

3.1 Example in Renesas Peripheral Driver Library

```
-----------------------------------------------------------------
 // Send data is set in slave.
 R_IEB_SlaveWrite(
```

```
    0,
    iebus_data,
    iebus_data_length
);

// Callback function is passed to R_IEB_SlaveMonitor as argument.
/* Monitor channel 0, using call back function */
R_IEB_SlaveMonitor(
    0,
    iebus_data,
    &iebus_data_length,
    IEB_CALL_BACK_FUNC
);

// Within callback function generated by IEB interrupt, check that
// transmission/reception completed by using R_IEB_GetStatus; then
// clear status flags.
Void IEB_CALL_BACK_FUNC(){
    R_IEB_GetStatus(
        0,
        &StatusValue,
        &SendstatusValue,
        &RcvstatusValue,
        PDL_NO_PTR,
        PDL_NO_PTR
    );

// Checking that Transmission flag set to 1
    if((SendstatusValue & 0x20) != 0x00){
        IEB.IETSR.BYTE = 0x6Fu;  // Transmission Status flag cleared.
    }

// Checking that Reception flag set to 1
    if((RcvstatusValue & 0x20) != 0x00){
        // Check that 4th argument of R_IEB_MasterReceive (reception
        // length; &iebus_data_length in this example) is not 0.
        if(iebus_data_length != 0x00){
            IEB.IERSR.BYTE = 0xFFu;  // Reception Status flag cleared.
        }
    }
}
```

3.2 Examples in Peripheral Driver Generator

(1) Example 1: The case where "Notify the transmission completion of

all data or an error detection by function call"is selected from the Master transmission method list of IEB, and Ieb0MasterTrFunc is typed in the Notification function name text box for master transmission.

```
// I/O register definition file included.
#include "iodefine_Renesas Peripheral Driver Library.h"

// Callback function for notifying the result of master transmission
void Ieb0MasterTrFunc(void){
   bool complete;
// Read the value of the Transmission flag
   R_PG_IEB_GetTransmitStatus_C0(
      0,
      &complete,
      0,
      0,
      0,
      0
   );
   if(complete){
   // Transmission Status register cleared.
      IEB.IETSR.BYTE = 0x6Fu;
   }
   Function();    // Processing after transmission completed.
}
```

(2) Example 2: The case where "Notify the transmission completion of all data or an error detection by function call"is selected from the Master reception method list of IEB, and Ieb0MasterReFunc is typed in the Notification function name text box for master reception.

```
// Callback function for notifying the result of master reception
// I/O register definition file included.
#include "iodefine_Renesas Peripheral Driver Library.h"

void Ieb0MasterReFunc(void){
   bool complete;
   uint8_t count;
// Read the value of the Reception flag
   R_PG_IEB_GetReceiveStatus_C0(
      0,
      0,
```

```
        &complete,
        0,
        0,
        0,
        0,
        0
    );
    R_PG_IEB_GetReceivedDataCount_C0(&count);
    if((complete)&&(count)){
    // Reception Status flag cleared.
        IEB.IERSR.BYTE = 0xFFu;
    }
    Function();    // Processing after reception completed.
}
```

---

(3) Example 3: The case where "Notify the transmission completion of all data, a slave read request, or an error detection by function call"is selected from the Slave monitor method list, and Ieb0SlaveFunc is typed in the Notification function name text box for slave monitoring.

```
// I/O register definition file included.
#include "iodefine_Renesas Peripheral Driver Library.h"

// Callback function for notifying the result of slave monitoring
void Ieb0SlaveFunc(void){
// Read the value of the Transmission flag
    bool tx_complete;
    bool rx_complete;
    uint8_t count;
    R_PG_IEB_GetTransmitStatus_C0(
        0,
        &tx_complete,
        0,
        0,
        0,
        0
    );
    if(tx_complete){
    // Transmission Status register cleared.
        IEB.IETSR.BYTE = 0x6Fu;
    }
    // Read the value of the Reception flag
    R_PG_IEB_GetReceiveStatus_C0(
```

```
        0,
        0,
        &rx_complete,
        0,
        0,
        0,
        0,
        0
    );
    R_PG_IEB_GetReceivedDataCount_C0(&count);
    if((rx_complete)&&(count)){
    // Reception Status register cleared
        IEB.IERSR.BYTE = 0xFFu;
    }
    Function();    // Processing after event detected.
}
------------------------------------------------------------------
```

4. Schedule of Fixing Problem

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