[Notes]

RX Family

Ethernet Module Using Firmware Integration Technology,

RX Driver Package

R20TS0447EJ0100
Rev.1.00
Jul. 16, 2019

Outline

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

1. When calling the “R_ETHER_Read_ZC2_BufRelease” function or “R_ETHER_Read” function

1. When Calling the “R_ETHER_Read_ZC2_BufRelease” Function or “R_ETHER_Read” Function

1.1 Applicable Products

These functions are provided in the r_ether_rx_v*.**.zip (*.** is a revision number) file included in the products listed in (1) and (2).

(1) RX family Ethernet module using Firmware Integration Technology
(Ethernet FIT module)

The applicable revision numbers and document numbers are as follows.

Table 1.1 Ethernet FIT module applicable products

Ethernet FIT module revision number	Document number
Rev.1.00	R01AN2009EJ0100
Rev.1.01	R01AN2009EJ0101
Rev.1.02	R01AN2009EJ0102
Rev.1.10	R01AN2009EJ0110
Rev.1.11	R01AN2009EJ0111
Rev.1.12	R01AN2009EJ0112
Rev.1.13	R01AN2009EJ0113
Rev.1.14	R01AN2009EJ0114
Rev.1.15	R01AN2009EJ0115
Rev.1.16	R01AN2009EJ0116

(2) RX Driver Package

The Ethernet FIT module in (1) is also included in the RX Driver Package listed below.
The product names and revision numbers of the applicable RX Driver Package and the revision numbers of the Ethernet FIT module are as follows.

Table 1.2 Products that include the Ethernet FIT module

RX Driver Package product name	RX Driver Package revision number	Document number	Revision number of the included Ethernet FIT module
RX64M Group RX Driver Package User's Manual	Rev.1.01	R01AN2460EJ0101	Rev.1.00
RX64M, RX71M Group RX Driver Package Ver.1.02	Rev.1.04	R01AN2606EJ0104	Rev.1.02
RX Family RX Driver Package Ver.1.10	Rev.1.10	R01AN3345EJ0100	Rev.1.10
RX Family RX Driver Package Ver.1.11	Rev.1.11	R01AN3467EJ0111	Rev.1.11
RX Family RX Driver Package Ver.1.12	Rev.1.12	R01AN3651EJ0112	Rev.1.12
RX Family RX Driver Package Ver.1.13	Rev.1.13	R01AN3859EJ0113	Rev.1.13
RX Family RX Driver Package Ver.1.14	Rev.1.14	R01AN4191EJ0114	Rev.1.14
RX Family RX Driver Package Ver.1.15	Rev.1.15	R01AN4372EJ0115	Rev.1.15
RX Family RX Driver Package Ver.1.16	Rev.1.16	R01AN4471EJ0116	Rev.1.15
RX Family RX Driver Package Ver.1.18	Rev.1.18	R01AN4659EJ0118	Rev.1.15
RX Family RX Driver Package Ver.1.19	Rev.1.19	R01AN4677EJ0119	Rev.1.15
RX Family RX Driver Package Ver.1.20	Rev.1.20	R01AN4794EJ0120	Rev.1.16

1.2 Applicable Devices

RX63N, RX65N, RX64M, and RX71M groups

1.3 Details and Conditions

When Ethernet frames are received after the "R_ETHER_Read_ZC2_BufRelease" function (Note) or "R_ETHER_Read" function releases the buffer, the following phenomena may occur.

(1) Received Ethernet frames cannot be read.

(2) Received error frames are read as normal Ethernet frames.

Note: The "R_ETHER_Read_ZC2_BufRelease" function is a lower function of the "R_ETHER_Read" function.

- Example of (1)

  In the “R_ETHER_Read_ZC2_BufRelease” function, the receive descriptor becomes accessible by the EDMAC at (i). As a result of this, a conflict occurs between the access to the receive descriptor from the EDMAC and CPU at (ii), therefore, the received Ethernet frames cannot be read.

- Example of (2)

  In the “R_ETHER_Read_ZC2_BufRelease” function, the receive descriptor becomes accessible by the EDMAC at (i). When the error frame is received, the error bit that is specified by the EDMAC is cleared by the CPU at (ii), therefore, the received error frame is read as a normal Ethernet frame.

```
 int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
 {
--------------------------- (omitted) ---------------------------
        /* When receive data exists */
        if (RACT != (papp_rx_desc[channel]->status & RACT))
        {
            /* Move to next descriptor */
            papp_rx_desc[channel]->status |= RACT;  (i)

            status = RFP1;
            status |= RFP0;
            status |= RFE;
            status |= RFS9_RFOVER;
            status |= RFS8_RAD;
            status |= RFS7_RMAF;
            status |= RFS4_RRF;
            status |= RFS3_RTLF;
            status |= RFS2_RTSF;
            status |= RFS1_PRE;
            status |= RFS0_CERF;

            papp_rx_desc[channel]->status &= (~status);  (ii)
            papp_rx_desc[channel] = papp_rx_desc[channel]->next;
        }
--------------------------- (omitted) ---------------------------
 } /* End of function R_ETHER_Read_ZC2_BufRelease() */
```

1.4 Workaround

Move the blue code in the “R_ETHER_Read_ZC2_BufRelease” function shown in [Before modification] to where the red code is in [After modification].

Before modification:

```
  int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
  {
--------------------------- (omitted) ---------------------------
        /* When receive data exists */
        if (RACT != (papp_rx_desc[channel]->status & RACT))
        {
            /* Move to next descriptor */
            papp_rx_desc[channel]->status |= RACT;

            status = RFP1;
            status |= RFP0;
            status |= RFE;
            status |= RFS9_RFOVER;
            status |= RFS8_RAD;
            status |= RFS7_RMAF;
            status |= RFS4_RRF;
            status |= RFS3_RTLF;
            status |= RFS2_RTSF;
            status |= RFS1_PRE;
            status |= RFS0_CERF;

            papp_rx_desc[channel]->status &= (~status);
            papp_rx_desc[channel] = papp_rx_desc[channel]->next;
        }
--------------------------- (omitted) ---------------------------
} /* End of function R_ETHER_Read_ZC2_BufRelease() */
```

After modification:

```
  int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
  {
--------------------------- (omitted) ---------------------------
        /* When receive data exists */
        if (RACT != (papp_rx_desc[channel]->status & RACT))
        {
            /* Move to next descriptor */
            status = RFP1;
            status |= RFP0;
            status |= RFE;
            status |= RFS9_RFOVER;
            status |= RFS8_RAD;
            status |= RFS7_RMAF;
            status |= RFS4_RRF;
            status |= RFS3_RTLF;
            status |= RFS2_RTSF;
            status |= RFS1_PRE;
            status |= RFS0_CERF;

            papp_rx_desc[channel]->status &= (~status);
            papp_rx_desc[channel]->status |= RACT;
            papp_rx_desc[channel] = papp_rx_desc[channel]->next;
        }
--------------------------- (omitted) ---------------------------
} /* End of function R_ETHER_Read_ZC2_BufRelease() */
```

1.5 Schedule for Fixing the Problem

This problem will be fixed in the later version. (Scheduled to be released in October 2019.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jul.16.19	-	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