

【注意事項】

R20TS0481JJ0100

Rev.1.00

2019.10.01 号

RX ファミリ
イーサネットモジュール Firmware Integration Technology,
RX Driver Package

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. “R_ETHER_Read_ZC2_BufRelease” 関数または “R_ETHER_Read” 関数を呼び出す場合の注意事項
2. “R_ETHER_Write_ZC2_SetBuf” 関数または “R_ETHER_Write” 関数を呼び出す場合の注意事項

1. “R_ETHER_Read_ZC2_BufRelease” 関数または “R_ETHER_Read” 関数を呼び出す場合の注意事項

1.1 該当製品

本関数は、r_ether_rx_v*.**zip (**はリビジョン番号) ファイルで提供しており、このファイルは以下の(1)と(2)の製品に同梱されています。

- (1) RX ファミリ イーサネットモジュール Firmware Integration Technology
(イーサネット FIT モジュール)

該当するリビジョンとドキュメントは以下のとおりです。

表 1.1 イーサネット FIT モジュール該当製品一覧

イーサネット FIT モジュールのリビジョン	資料番号
Rev.1.00	R01AN2009JJ0100
Rev.1.01	R01AN2009JJ0101
Rev.1.02	R01AN2009JJ0102
Rev.1.10	R01AN2009JJ0110
Rev.1.11	R01AN2009JJ0111
Rev.1.12	R01AN2009JJ0112
Rev.1.13	R01AN2009JJ0113
Rev.1.14	R01AN2009JJ0114
Rev.1.15	R01AN2009JJ0115
Rev.1.16	R01AN2009JJ0116
Rev.1.17	R01AN2009JJ0117

(2) RX Driver Package

(1)のイーサネット FIT モジュールは、以下 RX Driver Package にも同梱されています。
 該当する RX Driver Package の製品名、リビジョンおよび同梱しているイーサネット FIT モジュールのリビジョンは、以下のとおりです。

表 1.2 イーサネット FIT モジュール同梱製品一覧

RX Driver Package の製品名	RX Driver Package のリビジョン	資料番号	同梱しているイーサネット FIT モジュールのリビジョン
RX64M グループ RX Driver Package ユーザーズマニュアル	Rev.1.01	R01AN2460JJ0101	Rev.1.00
RX64M, RX71M グループ RX Driver Package Ver.1.02	Rev.1.04	R01AN2606JJ0104	Rev.1.02
RX ファミリ RX Driver Package Ver.1.10	Rev.1.10	R01AN3345JJ0100	Rev.1.10
RX ファミリ RX Driver Package Ver.1.11	Rev.1.11	R01AN3467JJ0111	Rev.1.11
RX ファミリ RX Driver Package Ver.1.12	Rev.1.12	R01AN3651JJ0112	Rev.1.12
RX ファミリ RX Driver Package Ver.1.13	Rev.1.13	R01AN3859JJ0113	Rev.1.13
RX ファミリ RX Driver Package Ver.1.14	Rev.1.14	R01AN4191JJ0114	Rev.1.14
RX ファミリ RX Driver Package Ver.1.15	Rev.1.15	R01AN4372JJ0115	Rev.1.15
RX ファミリ RX Driver Package Ver.1.16	Rev.1.16	R01AN4471JJ0116	Rev.1.15
RX ファミリ RX Driver Package Ver.1.18	Rev.1.18	R01AN4659JJ0118	Rev.1.15
RX ファミリ RX Driver Package Ver.1.19	Rev.1.19	R01AN4677JJ0119	Rev.1.15
RX ファミリ RX Driver Package Ver.1.20	Rev.1.20	R01AN4794JJ0120	Rev.1.16
RX ファミリ RX Driver Package Ver.1.22	Rev.1.22	R01AN4873JJ0122	Rev.1.17

1.2 該当デバイス

RX63N、RX65N、RX64M、RX71M、および RX72M グループ

1.3 内容および発生条件

受信ディスクリプタを 1 個に設定したときに “R_ETHER_Read_ZC2_BufRelease” 関数^注または “R_ETHER_Read” 関数におけるバッファ開放処理後、イーサネットフレームを受信できなくなる場合があります。

注: “R_ETHER_Read_ZC2_BufRelease” 関数は R_ETHER_Read” 関数の下位関数です

➤ 発生例

“R_ETHER_Read_ZC2_BufRelease” 関数において、(i)以降で EDMAC が EDRRR レジスタの RR ビットに 0 をセットにして、受信機能を停止させた場合、イーサネット FIT モジュールが受信を再開できないため、イーサネットフレームを受信できません。

```
int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
{
----- (中略) -----

    papp_rx_desc[channel]->status &= (~status);
    papp_rx_desc[channel]->status |= RACT;
    papp_rx_desc[channel] = papp_rx_desc[channel]->next;
}

pether_ch = g_eth_control_ch[channel].pether_control;
phy_access = g_eth_control_ch[channel].phy_access;
pedmac_adr = pether_ch[phy_access].pedmac;

if (0x00000000L == pedmac_adr->EDRRR.LONG) (i)
{
    /* Restart if stopped */
    pedmac_adr->EDRRR.LONG = 0x00000001L;
}

ret = ETHER_SUCCESS;
----- (中略) -----
} /* End of function R_ETHER_Read_ZC2_BufRelease() */
```

1.4 回避策

“R_ETHER_Read_ZC2” および “R_ETHER_Read_ZC2_BufRelease” 関数内に赤文字の処理を追加してください。

変更前: “R_ETHER_Read_ZC2”

```
int32_t R_ETHER_Read_ZC2 (uint32_t channel, void **pbuf)
{
    int32_t num_recvd;
    int32_t ret;
    int32_t complete_flag;
    int32_t ret2;

----- (中略) -----

    ret = ETHER_NO_DATA;
    complete_flag = ETHER_ERR_OTHER;
    while (ETHER_SUCCESS != complete_flag)
    {
        /* When receive data exists. */
        if (RACT != (papp_rx_desc[channel]->status & RACT))
        {
----- (中略) -----
    } /* End of function R_ETHER_Read_ZC2() */
```

変更後: “R_ETHER_Read_ZC2”

```
int32_t R_ETHER_Read_ZC2 (uint32_t channel, void **pbuf)
{
    int32_t num_recvd;
    int32_t ret;
    int32_t complete_flag;
    int32_t ret2;
    #if (1 == ETHER_CFG_EMAC_RX_DESCRIPTOR)
        volatile struct st_edmac __evenaccess * pedmac_adr;
        const ether_control_t * pether_ch;
        uint32_t phy_access;
    #endif
    ----- (中略) -----
    ret = ETHER_NO_DATA;
    complete_flag = ETHER_ERR_OTHER;

    #if (1 == ETHER_CFG_EMAC_RX_DESCRIPTOR)
        pether_ch = g_eth_control_ch[channel].pether_control;
        phy_access = g_eth_control_ch[channel].phy_access;
        pedmac_adr = pether_ch[phy_access].pedmac;
    #endif

    while (ETHER_SUCCESS != complete_flag)
    {
        /* When receive data exists. */
        #if (1 == ETHER_CFG_EMAC_RX_DESCRIPTOR)
            if (0x00000000L == pedmac_adr->EDRRR.LONG)
        #else
            if (RACT != (papp_rx_desc[channel]->status & RACT))
        #endif
        {
            ----- (中略) -----
        }
    }
} /* End of function R_ETHER_Read_ZC2() */
```

变更前: “R_ETHER_Read_ZC2_BufRelease”

```
int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
{
----- (中略) -----
    /* When the Link up processing is completed */
    else
    {
        /* When receive data exists */
        if (RACT != (papp_rx_desc[channel]->status & RACT))
        {
----- (中略) -----
    } /* End of function R_ETHER_Read_ZC2_BufRelease() */
}
```

変更後: “R_ETHER_Read_ZC2_BufRelease”

```
int32_t R_ETHER_Read_ZC2_BufRelease (uint32_t channel)
{
----- (中略) -----

    /* When the Link up processing is completed */
    else
    {
        #if (1 == ETHER_CFG_EMAC_RX_DESCRIPTOR)
            pether_ch = g_eth_control_ch[channel].pether_control;
            phy_access = g_eth_control_ch[channel].phy_access;
            pedmac_adr = pether_ch[phy_access].pedmac;
        #endif

        /* When receive data exists */
        #if (1 == ETHER_CFG_EMAC_RX_DESCRIPTOR)
            if (0x00000000L == pedmac_adr->EDRRR.LONG)
        #else
            if (RACT != (papp_rx_desc[channel]->status & RACT))
        #endif
        {
----- (中略) -----
    } /* End of function R_ETHER_Read_ZC2_BufRelease() */
}
```

1.5 恒久対策

今後のバージョンで改修予定です。(2020年2月予定)

2. “R_ETHER_Write_ZC2_SetBuf” 関数または “R_ETHER_Write” 関数を呼び出す場合の注意事項

2.1 該当製品

本関数は、r_ether_rx_v*.**zip (**はリビジョン番号) ファイルで提供しており、このファイルは以下の(1)と(2)の製品に同梱されています。

- (1) RX ファミリ イーサネットモジュール Firmware Integration Technology (イーサネット FIT モジュール)

該当するリビジョンとドキュメントは以下のとおりです。

表 2.1 イーサネット FIT モジュール該当製品一覧

イーサネット FIT モジュールのリビジョン	資料番号
Rev.1.00	R01AN2009JJ0100
Rev.1.01	R01AN2009JJ0101
Rev.1.02	R01AN2009JJ0102
Rev.1.10	R01AN2009JJ0110
Rev.1.11	R01AN2009JJ0111
Rev.1.12	R01AN2009JJ0112
Rev.1.13	R01AN2009JJ0113
Rev.1.14	R01AN2009JJ0114
Rev.1.15	R01AN2009JJ0115
Rev.1.16	R01AN2009JJ0116
Rev.1.17	R01AN2009JJ0117

(2) RX Driver Package

(1)のイーサネット FIT モジュールは、以下 RX Driver Package にも同梱されています。
 該当する RX Driver Package の製品名、リビジョンおよび同梱しているイーサネット FIT モジュールのリビジョンは、以下のとおりです。

表 2.2 イーサネット FIT モジュール同梱製品一覧

RX Driver Package の製品名	RX Driver Package のリビジョン	資料番号	同梱しているイーサネット FIT モジュールのリビジョン
RX64M グループ RX Driver Package ユーザーズマニュアル	Rev.1.01	R01AN2460JJ0101	Rev.1.00
RX64M, RX71M グループ RX Driver Package Ver.1.02	Rev.1.04	R01AN2606JJ0104	Rev.1.02
RX ファミリ RX Driver Package Ver.1.10	Rev.1.10	R01AN3345JJ0100	Rev.1.10
RX ファミリ RX Driver Package Ver.1.11	Rev.1.11	R01AN3467JJ0111	Rev.1.11
RX ファミリ RX Driver Package Ver.1.12	Rev.1.12	R01AN3651JJ0112	Rev.1.12
RX ファミリ RX Driver Package Ver.1.13	Rev.1.13	R01AN3859JJ0113	Rev.1.13
RX ファミリ RX Driver Package Ver.1.14	Rev.1.14	R01AN4191JJ0114	Rev.1.14
RX ファミリ RX Driver Package Ver.1.15	Rev.1.15	R01AN4372JJ0115	Rev.1.15
RX ファミリ RX Driver Package Ver.1.16	Rev.1.16	R01AN4471JJ0116	Rev.1.15
RX ファミリ RX Driver Package Ver.1.18	Rev.1.18	R01AN4659JJ0118	Rev.1.15
RX ファミリ RX Driver Package Ver.1.19	Rev.1.19	R01AN4677JJ0119	Rev.1.15
RX ファミリ RX Driver Package Ver.1.20	Rev.1.20	R01AN4794JJ0120	Rev.1.16
RX ファミリ RX Driver Package Ver.1.22	Rev.1.22	R01AN4873JJ0122	Rev.1.17

2.2 該当デバイス

RX63N、RX65N、RX64M、RX71M、および RX72M グループ

2.3 内容および発生条件

送信ディスクリプタを 1 個に設定したときに、“R_ETHER_Write_ZC2_SetBuf” 関数^注または “R_ETHER_Write” 関数における送信開始処理後、イーサネットフレームを送信できなくなる場合があります。

注: “R_ETHER_Write_ZC2_SetBuf” 関数は R_ETHER_Write” 関数の下位関数です

➤ 発生例

“R_ETHER_Write_ZC2_SetBuf” 関数において、(i)以降で EDMAC が EDTRR レジスタの TR ビットに 0 をセットして、送信機能を停止させた場合、イーサネット FIT モジュールは送信を再開できないため、イーサネットフレームを送信できません。

```

ether_return_t R_ETHER_Write_ZC2_SetBuf (uint32_t channel, const
uint32_t len)
{
----- (中略) -----

/* When the Link up processing is completed */
else
{
/* The data of the buffer is made active. */
papp_tx_desc[channel]->bufsize = len;
papp_tx_desc[channel]->status &= (~(TFP1 | TFP0));
papp_tx_desc[channel]->status |= ((TFP1 | TFP0) | TACT);
papp_tx_desc[channel] = papp_tx_desc[channel]->next;

pether_ch = g_eth_control_ch[channel].pether_control;
phy_access = g_eth_control_ch[channel].phy_access;
pedmac_adr = pether_ch[phy_access].pedmac;

if (0x00000000L == pedmac_adr->EDTRR.LONG) (i)
{
/* Restart if stopped */
pedmac_adr->EDTRR.LONG = 0x00000001L;
}

ret = ETHER_SUCCESS;

----- (中略) -----

} /* End of function R_ETHER_Write_ZC2_SetBuf() */

```

2.4 回避策

“R_ETHER_Write_ZC2_GetBuf” 関数内に赤文字の処理を追加してください。

変更前:

```
ether_return_t R_ETHER_Write_ZC2_GetBuf (uint32_t channel, void
**pbuf, uint16_t *pbuf_size)
{
    ether_return_t ret;
    ----- (中略) -----

    /* When the Link up processing is completed */
    else
    {
        /* All transmit buffers are full */
        if (TACT == (papp_tx_desc[channel]->status & TACT))
        {
            ret = ETHER_ERR_TACT;
        }
        else
        {
            /* Give application another buffer to work with */
            (*pbuf) = papp_tx_desc[channel]->buf_p;
            (*pbuf_size) = ETHER_CFG_BUFSIZE;
            ret = ETHER_SUCCESS;
        }
    }

    ----- (中略) -----
} /* End of function R_ETHER_Write_ZC2_GetBuf() */
```

変更後:

```

ether_return_t R_ETHER_Write_ZC2_GetBuf (uint32_t channel, void
**pbuf, uint16_t *pbuf_size)
{
    ether_return_t ret;
    #if (1 == ETHER_CFG_EMAC_TX_DESCRIPTOR)
        volatile struct st_edmac R_BSP_EVENACCESS_SFR * pedmac_adr;
        const ether_control_t * pether_ch;
        uint32_t phy_access;
    #endif
    ----- (中略) -----
    /* When the Link up processing is completed */
    else
    {
        #if (1 == ETHER_CFG_EMAC_TX_DESCRIPTOR)
            pether_ch = g_eth_control_ch[channel].pether_control;
            phy_access = g_eth_control_ch[channel].phy_access;
            pedmac_adr = pether_ch[phy_access].pedmac;
        #endif

        /* All transmit buffers are full */
        #if (1 == ETHER_CFG_EMAC_TX_DESCRIPTOR)
            if (0x00000000L != pedmac_adr->EDTRR.LONG)
        #else
            if (TACT == (papp_tx_desc[channel]->status & TACT))
        #endif
        {
            ret = ETHER_ERR_TACT;
        }
        else
        {
            /* Give application another buffer to work with */
            (*pbuf) = papp_tx_desc[channel]->buf_p;
            (*pbuf_size) = ETHER_CFG_BUFSIZE;
            ret = ETHER_SUCCESS;
        }
    }
    ----- (中略) -----
} /* End of function R_ETHER_Write_ZC2_GetBuf() */

```

2.5 恒久対策

今後のバージョンで改修予定です。(2020年2月予定)

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Oct.01.19	-	新規発行

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。

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。