

【注意事項】

R20TS0946JS0100

Rev.1.00

2023.06.16 号

RX ファミリ

Serial NOR Flash Memory アクセスクロック同期制御モジュール

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. 特定の関数（表 1.3 記載）において、誤ったエラーコードを返す場合がある

1. 特定の関数（表 1.3 記載）において、誤ったエラーコードを返す場合がある

1.1 該当製品

- 1) Serial NOR Flash memory アクセスクロック同期制御モジュール（FLASH SPI FIT モジュール）

該当するリビジョンおよびドキュメントは以下のとおりです。

表 1.1 FLASH SPI FIT モジュール該当製品

FLASH SPI FIT モジュール リビジョン	資料番号
Rev.3.03	R01AN2662JJ0303
Rev.3.02	R01AN2662JJ0302
Rev.3.01	R01AN2662JJ0301
Rev.3.00	R01AN2662JJ0300

- 2) RX Driver Package

- 1) の FLASH SPI FIT モジュールは 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、および同梱される FLASH SPI FIT モジュールのリビジョンは以下のとおりです。

表 1.2 FLASH SPI FIT モジュール該当製品

RX Driver Package 製品名	RX Driver Package リビジョン	資料番号	FLASH SPI FIT モジュール リビジョン
RX ファミリ RX Driver Package Ver.1.36	Rev.1.36	R01AN6515xx0136	Rev.3.03
RX ファミリ RX Driver Package Ver.1.34	Rev.1.34	R01AN6323xx0134	Rev.3.03
RX ファミリ RX Driver Package Ver.1.33	Rev.1.33	R01AN6073xx0133	Rev.3.03
RX ファミリ RX Driver Package Ver.1.32	Rev.1.32	R01AN6013xx0132	Rev.3.02
RX ファミリ RX Driver Package Ver.1.31	Rev.1.31	R01AN5975xx0131	Rev.3.02
RX ファミリ RX Driver Package Ver.1.30	Rev.1.30	R01AN5882xx0130	Rev.3.02
RX ファミリ RX Driver Package Ver.1.29	Rev.1.29	R01AN5826xx0129	Rev.3.01
RX ファミリ RX Driver Package Ver.1.27	Rev.1.27	R01AN5600xx0127	Rev.3.01

RX Driver Package 製品名	RX Driver Package リビジョン	資料番号	FLASH SPI FIT モジュール リビジョン
RX ファミリ RX Driver Package Ver.1.26	Rev.1.26	R01AN5401xx0126	Rev.3.01
RX ファミリ RX Driver Package Ver.1.25	Rev.1.25	R01AN5371xx0125	Rev.3.01
RX ファミリ RX Driver Package Ver.1.24	Rev.1.24	R01AN5267xx0124	Rev.3.01
RX ファミリ RX Driver Package Ver.1.23	Rev.1.23	R01AN4976xx0123	Rev.3.01
RX ファミリ RX Driver Package Ver.1.22	Rev.1.22	R01AN4873xx0122	Rev.3.01
RX ファミリ RX Driver Package Ver.1.20	Rev.1.20	R01AN4794JJ0120	Rev.3.01
RX ファミリ RX Driver Package Ver.1.19	Rev.1.19	R01AN4677JJ0119	Rev.3.00

1.2 該当デバイス

RX110、RX111、RX113、RX130、RX13T、RX140 グループ

RX230、RX231、RX23E-A、RX23T、RX23W、RX24T、RX24U グループ

RX64M、RX65N、RX660、RX66N、RX66T、RX671 グループ

RX71M、RX72M、RX72N、RX72T グループ

1.3 内容

r_flash_spi_drvif.c 内のいくつかの関数で間違った if 条件が見つかりました。結果として if 条件が満たされず、if 文下のブロックの処理が実行されません。

```
//r_flash_spi_drvif.c
...

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD) //ここが間違っています!!
{
    Return FLASH_SPI_ERR_HARD; //この部分が入力されません!!
}
...

// Note:
// MEMDRV_SUCCESS = 0
// MEMDRV_ERR_HARD = -2
```

本注意事項に該当する関数と影響を受ける API は以下のとおりです。

表 1.3 本注意事項に該当する関数と影響を受ける API

本注意事項に該当する関数	影響を受ける API
<code>r_flash_spi_drvif_tx</code>	<code>R_FLASH_SPI_Set_Write_Protect</code> , <code>R_FLASH_SPI_Write_Configuration</code> , <code>R_FLASH_SPI_Quad_Enable</code> , <code>R_FLASH_SPI_Quad_Disable</code> , <code>R_FLASH_SPI_Write_Data_Page</code> , <code>R_FLASH_SPI_Erase</code> , <code>R_FLASH_SPI_Polling</code> , <code>R_FLASH_SPI_Read_Status</code> , <code>R_FLASH_SPI_Read_Security</code> , <code>R_FLASH_SPI_Read_ID</code> , <code>R_FLASH_SPI_Set_4byte_Address_Mode</code> , <code>R_FLASH_SPI_Read_Data</code> , <code>R_FLASH_SPI_Read_Configuration</code>
<code>r_flash_spi_drvif_tx_add</code>	<code>R_FLASH_SPI_Write_Data_Page</code>
<code>r_flash_spi_drvif_tx_data</code>	<code>R_FLASH_SPI_Write_Data_Page</code>
<code>r_flash_spi_drvif_rx</code>	<code>R_FLASH_SPI_Read_Configuration</code> , <code>R_FLASH_SPI_Set_Write_Protect</code> , <code>R_FLASH_SPI_Quad_Enable</code> , <code>R_FLASH_SPI_Quad_Disable</code> , <code>R_FLASH_SPI_Write_Configuration</code> , <code>R_FLASH_SPI_Read_Security</code> , <code>R_FLASH_SPI_Polling</code> , <code>R_FLASH_SPI_Read_ID</code> , <code>R_FLASH_SPI_Read_Status</code> , <code>R_FLASH_SPI_Write_Data_Page</code>
<code>r_flash_spi_drvif_rx_add</code>	N/A
<code>r_flash_spi_drvif_rx_data</code>	<code>R_FLASH_SPI_Read_Data</code>

1.4 発生条件

- `r_flash_spi_drvif_tx()` で、`R_MEMDRV_Tx()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
//r_flash_spi_drvif_tx()
...
ret_drv = R_MEMDRV_Tx(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

- `r_flash_spi_drvif_tx_add()` で、`R_MEMDRV_TxData()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
// r_flash_spi_drvif_tx_add()
...
ret_drv = R_MEMDRV_TxData(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

- `r_flash_spi_drvif_tx_data()` で、`R_MEMDRV_TxData()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
//r_flash_spi_drvif_tx_data()
...
ret_drv = R_MEMDRV_TxData(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

- `r_flash_spi_drvif_rx()` で、`R_MEMDRV_Rx()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
// r_flash_spi_drvif_rx()
...
ret_drv = R_MEMDRV_Rx(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

- `r_flash_spi_drvif_rx_add()` で、`R_MEMDRV_RxData()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
// r_flash_spi_drvif_rx_add()
...
ret_drv = R_MEMDRV_RxData(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

- `r_flash_spi_drvif_rx_data()` で、`R_MEMDRV_RxData()` が `ret_drv = MEMDRV_ERR_HARD` を返すと、`FLASH_SPI_ERR_OTHER` が呼び出し元に返されますが、これは間違いです。

```
// r_flash_spi_drvif_rx_data()
...
ret_drv = R_MEMDRV_RxData(devno, &memdrv_info);

if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)
{
    return FLASH_SPI_ERR_HARD;
}
else if (MEMDRV_SUCCESS > ret_drv)
{
    return FLASH_SPI_ERR_OTHER;
}
return FLASH_SPI_SUCCESS;
```

1.5 回避策

暫定的な対策：if 条件を以下のように変更してください。

変更前：if (MEMDRV_SUCCESS == MEMDRV_ERR_HARD)

変更後：if (MEMDRV_ERR_HARD == ret_drv)

FLASH SPI FIT を Rev.3.10 以降にアップデートしてください。

1.6 恒久対策

上記の問題は FLASH SPI FIT Rev.3.10 で改修されています。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Jun.16.23	-	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。