

**【注意事項】**

R20TS0133JJ0100

Rev.1.00

2017.03.01 号

## RX ファミリ

SCI モジュール Firmware Integration Technology Rev.1.80,  
RX Driver Package Ver.1.11

## 概要

RX ファミリ SCI モジュール Firmware Integration Technology Rev.1.80 (以下、SCI FIT モジュール) および RX Driver Package Ver.1.11 の使用上の注意事項を連絡します。

1. SCI FIT モジュールのクロック同期式モードの送受信処理に関する注意事項

## 1. SCI FIT モジュールのクロック同期式モードの送受信処理に関する注意事項

## 1.1 該当製品

- RX ファミリ SCI モジュール Firmware Integration Technology

リビジョン : Rev.1.80

対応するドキュメントは以下です。

- RX ファミリ SCI モジュール Firmware Integration Technology アプリケーションノート  
資料番号: R01AN1815JJ0180

上記 SCI FIT モジュールが同梱<sup>注1</sup>されている下記製品も対象になります。

注 1: r\_sci\_rx\_v1.80.zip として SCI FIT モジュールが同梱されています。

- RX ファミリ RX Driver Package Ver.1.11

リビジョン : Rev.1.11

対応するドキュメントは以下です。

- RX ファミリ RX Driver Package Ver.1.11 アプリケーションノート  
資料番号: R01AN3467JJ0111

## 1.2 該当 MCU

RX110、RX111、RX113、RX130、RX210、RX230、RX231、RX23T、RX24T、RX63N、RX631、RX64M、RX651、RX65N および RX71M グループ

### 1.3 内容

SCI FIT モジュールのクロック同期式モードの送受信処理において、指定した数よりも多くのデータを送受信する場合があります。

### 1.4 発生条件

以下のいずれかに該当する場合に発生します。

(1) データ受信時

受信用のダミーデータの書き込み後からカウンタのデクリメントまでの間<sup>(注)</sup>に1フレーム分の通信に要する時間以上が経過した場合に発生します。

注：FIFO モード使用時は下記の青破線矢印の区間

FIFO モード未使用時下記の赤破線矢印の区間

```

static sci_err_t sci_receive_sync_data(sci_hdl_t const hdl,
                                       uint8_t      *p_dst,
                                       uint16_t const length)
{
(中略)
#if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
(中略)
        for (cnt=0; cnt<thresh_cnt; cnt++)
        {
            SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
        }
        hdl->tx_cnt -= cnt;
    }
    else
#endif
(中略)
    {
        SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
        hdl->tx_cnt--;
        if (1 < length)
        {
            dummy = hdl->rom->regs->TDR;
            if (0x80 == (hdl->rom->regs->SSR.BYTE & 0x80))
            {
                SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
                hdl->tx_cnt--;
            }
        }
    }
(中略)
}
    
```

ダミーデータの書き込み

カウンタのデクリメント

ダミーデータの書き込み (1回目)

カウンタのデクリメント (1回目)

ダミーデータの書き込み (2回目)

カウンタのデクリメント (2回目)

(2) データ送信時

送信用のデータの書き込み後からカウンタのデクリメントまでの間<sup>(注)</sup>に1フレーム分の通信に要する時間以上が経過した場合に発生します。

注：FIFOモード使用時は下記の青破線矢印の区間

FIFOモード未使用時下記の赤破線矢印の区間

```

static sci_err_t sci_send_sync_data(sci_hdl_t const hdl,
                                   uint8_t      *p_src,
                                   uint8_t      *p_dst,
                                   uint16_t const length,
                                   bool          save_rx_data)
{
(中略)
#if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
(中略)
        for (cnt=0; cnt<thresh_cnt; cnt++)
        {
            SCI_TDR(*hdl->u_tx_data.buf++); ← データの書き込み
        }
        hdl->tx_cnt -= cnt; ← カウンタのデクリメント
    }
    else
#endif
(中略)
    {
        SCI_TDR(*hdl->u_tx_data.buf++); ← データの書き込み
    }
    hdl->tx_cnt--; ← カウンタのデクリメント
}
(中略)

```

## 1.5 回避策

SCI FIT モジュールのソースコード “r\_sci\_rx.c” の “sci\_receive\_sync\_data()” 関数 および “sci\_send\_sync\_data()” 関数を以下のとおり変更してください。

以下に詳細を記します。各関数の青文字の処理を赤文字の処理に変更してください。

(1) 1.4 発生条件(1)の場合

■ SCI FIT モジュールのソースコード “r\_sci\_rx.c” の “sci\_receive\_sync\_data()” 関数

修正前: ダミーデータを TDR レジスタにセットし、カウンタを制御している部分の処理 (2 箇所) 抜粋

```
static sci_err_t sci_receive_sync_data(sci_hdl_t const hdl,
                                       uint8_t      *p_dst,
                                       uint16_t const length)
{
  (中略)
  #if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
      (中略)
      if (length > SCI_FIFO_FRAME_SIZE)
      {
        thresh_cnt = SCI_FIFO_FRAME_SIZE;
      }
      else
      {
        if (length < hdl->rx_dflt_thresh)
        {
          hdl->rom->regs->FCR.BIT.RTRG = length;
        }
        thresh_cnt = length;
      }

      for (cnt=0; cnt<thresh_cnt; cnt++)
      {
        SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
      }

      hdl->tx_cnt -= cnt;
    }
    else
  #endif
  (中略)
  {
    SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
    hdl->tx_cnt--;

    if (1 < length)
    {
      dummy = hdl->rom->regs->TDR;
      if (0x80 == (hdl->rom->regs->SSR.BYTE & 0x80))
      {
        SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
        hdl->tx_cnt--;
      }
    }
  }
  (中略)
}
```

修正後:

```
static sci_err_t sci_receive_sync_data(sci_hdl_t const hdl,
                                       uint8_t      *p_dst,
                                       uint16_t const length)
{
  (中略)
  #if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
      (中略)

      if (length > SCI_FIFO_FRAME_SIZE)
      {
        thresh_cnt = SCI_FIFO_FRAME_SIZE;
      }
      else
      {
        if (length < hdl->rx_dflt_thresh)
        {
          hdl->rom->regs->FCR.BIT.RTRG = length;
        }
        thresh_cnt = length;
      }

      hdl->tx_cnt -= thresh_cnt;

      for (cnt=0; cnt<thresh_cnt; cnt++)
      {
        SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
      }
    }
    else
  #endif
  (中略)
  {
    hdl->tx_cnt--;
    SCI_TDR(SCI_CFG_DUMMY_TX_BYTE);
  }
  (中略)
}
```

## (2) 1.4 発生条件(2)の場合

## ■ SCI FIT モジュールのソースコード “r\_sci\_rx.c” の “sci\_send\_sync\_data()” 関数

修正前: データを TDR レジスタにセットし、カウンタを制御している部分の処理 (2 箇所) 抜粋

```
static sci_err_t sci_send_sync_data(sci_hdl_t const hdl,
                                   uint8_t      *p_src,
                                   uint8_t      *p_dst,
                                   uint16_t const length,
                                   bool          save_rx_data)
{
  (中略)
  #if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
      (中略)
      thresh_cnt = hdl->rom->regs->FCR.BIT.RTRG;

      for (cnt=0; cnt<thresh_cnt; cnt++)
      {
        SCI_TDR(*hdl->u_tx_data.buf++);
      }

      hdl->tx_cnt -= cnt;
    }
    else
  #endif
  (中略)
  {
    SCI_TDR(*hdl->u_tx_data.buf++);
    hdl->tx_cnt--;
  }
  (中略)
}
```

修正後:

```
static sci_err_t sci_send_sync_data(sci_hdl_t const hdl,
                                   uint8_t      *p_src,
                                   uint8_t      *p_dst,
                                   uint16_t const length,
                                   bool          save_rx_data)
{
  (中略)
  #if SCI_CFG_FIFO_INCLUDED
    if (true == hdl->fifo_ctrl)
    {
      (中略)
      thresh_cnt = hdl->rom->regs->FCR.BIT.RTRG;

      hdl->tx_cnt -= thresh_cnt;

      for (cnt=0; cnt<thresh_cnt; cnt++)
      {
        SCI_TDR(*hdl->u_tx_data.buf++);
      }
    }
    else
  #endif
  (中略)
  {
    hdl->tx_cnt--;
    SCI_TDR(*hdl->u_tx_data.buf++);
  }
  (中略)
}
```

## 1.6 恒久対策

➤ SCI FIT モジュール

次期リビジョンで改修予定です。

➤ RX ファミリ RX Driver Package

次期リリース予定の RX ファミリ RX Driver Package Ver.1.12 にて、本注意事項を改修した SCI FIT モジュール(Rev.1.90)を同梱する予定です。

以上



改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2017.03.01	-	新規発行

ルネサスエレクトロニクス株式会社  
 〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)

■総合お問い合わせ先  
<https://www.renesas.com/contact/>

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。

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。