

RX63N グループ、RX631 グループ

R01AN2628JJ0100

Rev.1.00

2015.04.01

割り込みを使用しない調歩同期式 SCIc 送受信

要旨

シリアルコミュニケーションインタフェース(以下、SCI)は、受信データフル、受信エラー、送信終了、送信データエンプティの各割り込み処理を実行せずに、ソフトウェアで各割り込みの要因発生を確認して、特定の処理に分岐させることができます。

本アプリケーションノートでは、RX63N グループ、RX631 グループの調歩同期式 SCI 送受信を使用して、割り込みを使用せずにシリアル送受信を行う方法について説明します。

対象デバイス

- | | | |
|-------------|-------------|----------------------|
| ・RX63N グループ | 177、176 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX63N グループ | 145、144 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX63N グループ | 100 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX631 グループ | 177、176 ピン版 | ROM 容量 : 256KB~2MB |
| ・RX631 グループ | 145、144 ピン版 | ROM 容量 : 256KB~2MB |
| ・RX631 グループ | 100 ピン版 | ROM 容量 : 256KB~2MB |
| ・RX631 グループ | 64、48 ピン版 | ROM 容量 : 256KB~512KB |

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

目次

1. 仕様	3
2. 動作確認条件	4
3. 関連アプリケーションノート	4
4. ハードウェア説明	5
4.1 使用端子一覧	5
5. ソフトウェア説明	7
5.1 動作概要	7
5.1.1 シリアル受信	7
5.1.2 シリアル送信	8
5.2 ファイル構成	9
5.3 オプション設定メモリ	10
5.4 定数一覧	10
5.5 構造体/共用体一覧	13
5.6 変数一覧	14
5.7 関数一覧	27
5.8 関数仕様	28
5.9 フローチャート	31
5.9.1 メイン処理	31
5.9.2 ポート初期設定	32
5.9.3 周辺機能初期設定	32
5.9.4 IRQ 初期設定	33
5.9.5 SCI 初期設定	34
5.9.6 SCI 受信	35
5.9.7 SCI 送信	36
5.9.8 受信データフル時の処理	37
5.9.9 受信エラー時の処理	38
5.9.10 送信データエンプティ時の処理	39
5.9.11 送信終了時の処理	40
6. サンプルコード	41
7. 参考ドキュメント	41

1. 仕様

SCI を使用して、調歩同期式のシリアル送受信を行います。

リセット解除後、スイッチ入力ごとに送受信を 1 度行います。

受信は、12 バイトを受信します。受信したデータは受信バッファに格納し、12 バイトの受信が完了すると、LED1 を点灯します。再度スイッチが入力され 12 バイトの受信が完了すると、LED1 を消灯します(LED1 は反転出力)。受信中にエラーが発生すると、受信動作を終了し、LED2 を点灯します。

送信は、送信バッファに設定した 12 バイトの文字コード「Hello world!」を送信します。12 バイトの送信が完了すると、LED0 が点灯します。再度スイッチが入力され 12 バイトの送信が完了すると、LED0 が消灯します(LED0 は反転出力)。

使用する SCI のチャンネルは、各関数の第 1 引数で選択できます。サンプルコードではチャンネル 0 (SCI0) を選択しています。

- 転送速度 : 57600bps
- データ長 : 8 ビット
- ストップビット : 2 ビット
- パリティ : なし
- ハードウェアフロー制御 : なし

表 1.1 に使用する周辺機能と用途を、図 1.1 に使用例を示します。

表 1.1 使用する周辺機能と用途

周辺機能	用途
SCI (チャンネル 0~12 より選択)	調歩同期式シリアル送受信
I/O ポート	LED 点灯
外部端子割り込み(以下、IRQ)	送受信開始スイッチ

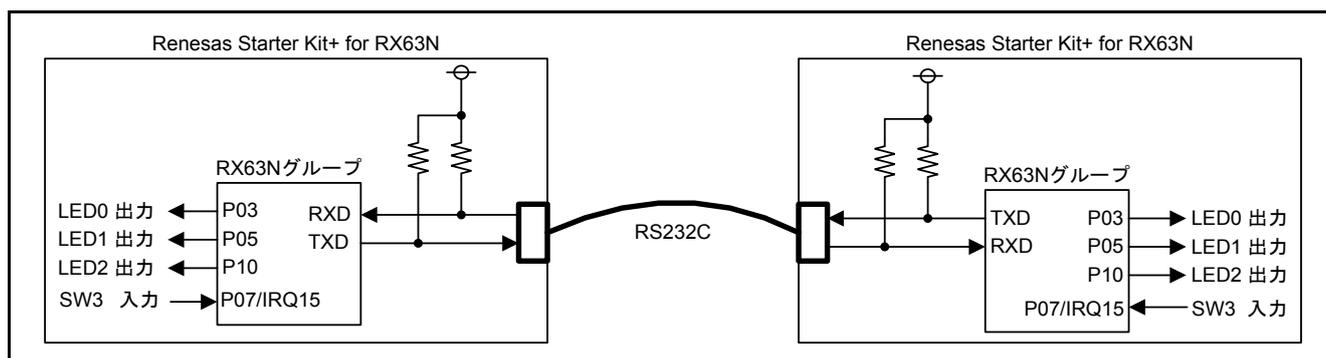


図 1.1 使用例

2. 動作確認条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表 2.1 動作確認条件

項目	内容
使用マイコン	R5F563NBDDFC (RX63N グループ)
動作周波数	メインクロック: 12MHz PLL: 192MHz (メインクロック 1 分周 16 通倍) システムクロック (ICLK): 96MHz (PLL 2 分周) 周辺モジュールクロック B (PCLKB): 48MHz (PLL 4 分周)
動作電圧	3.3V
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded Workshop Version 4.09.01
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family V.1.02 Release 01 コンパイルオプション -cpu=rx600 -output=obj="\$(CONFIGDIR)¥\$(FILELEAF).obj" -debug -nologo (統合開発環境のデフォルト設定を使用しています)
iodef.h のバージョン	Version 1.6A
エンディアン	リトルエンディアン
動作モード	シングルチップモード
プロセッサモード	スーパバイザモード
サンプルコードのバージョン	Version 1.00
使用ボード	Renesas Starter Kit+ for RX63N (製品型名: R0K50563NC000BE)

3. 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。併せて参照してください。

RX63N グループ、RX631 グループ 初期設定例 Rev.1.10 (R01AN1245JJ)

上記アプリケーションノートの初期設定関数を、本アプリケーションノートのサンプルコードで使用しています。Rev は本アプリケーションノート作成時点のものであります。

最新版がある場合、最新版に差し替えて使用してください。最新版はルネサスエレクトロニクスホームページで確認および入手してください。

4. ハードウェア説明

4.1 使用端子一覧

表 4.1、表 4.2 に使用端子と機能を示します。

使用端子は 176 ピン版の製品を想定しています。176 ピン版未満の製品を使用する場合は、使用する製品に合わせて端子を選択してください。

表 4.1 使用端子と機能(1)

端子名	入出力	内容
P03	出力	LED0 出力(SCI 送信完了)
P05	出力	LED1 出力(SCI 受信完了)
P10	出力	LED2 出力(SCI 受信エラー)
P07/IRQ15	入力	送受信開始の外部入力
P21/RXD0	入力	SCI0 の受信データ入力端子(注 1)
P33/RXD0	入力	SCI0 の受信データ入力端子(注 1)
P20/TXD0	出力	SCI0 の送信データ出力端子(注 1)
P32/TXD0	出力	SCI0 の送信データ出力端子(注 1)
P15/RXD1	入力	SCI1 の受信データ入力端子(注 1)
P30/RXD1	入力	SCI1 の受信データ入力端子(注 1)
PF2/RXD1	入力	SCI1 の受信データ入力端子(注 1)
P16/TXD1	出力	SCI1 の送信データ出力端子(注 1)
P26/TXD1	出力	SCI1 の送信データ出力端子(注 1)
PF0/TXD1	出力	SCI1 の送信データ出力端子(注 1)
P12/RXD2	入力	SCI2 の受信データ入力端子(注 1)
P52/RXD2	入力	SCI2 の受信データ入力端子(注 1)
P13/TXD2	出力	SCI2 の送信データ出力端子(注 1)
P50/TXD2	出力	SCI2 の送信データ出力端子(注 1)
P16/RXD3	入力	SCI3 の受信データ入力端子(注 1)
P25/RXD3	入力	SCI3 の受信データ入力端子(注 1)
P17/TXD3	出力	SCI3 の送信データ出力端子(注 1)
P23/TXD3	出力	SCI3 の送信データ出力端子(注 1)
PB0/RXD4	入力	SCI4 の受信データ入力端子(注 1)
PB1/TXD4	出力	SCI4 の送信データ出力端子(注 1)
PA2/RXD5	入力	SCI5 の受信データ入力端子(注 1)
PA3/RXD5	入力	SCI5 の受信データ入力端子(注 1)
PC2/RXD5	入力	SCI5 の受信データ入力端子(注 1)
PA4/TXD5	出力	SCI5 の送信データ出力端子(注 1)
PC3/TXD5	出力	SCI5 の送信データ出力端子(注 1)
P01/RXD6	入力	SCI6 の受信データ入力端子(注 1)
P33/RXD6	入力	SCI6 の受信データ入力端子(注 1)
PB0/RXD6	入力	SCI6 の受信データ入力端子(注 1)
P00/TXD6	出力	SCI6 の送信データ出力端子(注 1)
P32/TXD6	出力	SCI6 の送信データ出力端子(注 1)
PB1/TXD6	出力	SCI6 の送信データ出力端子(注 1)

注 1. 使用する SCI の端子は、SCI_Init 関数の第 1 引数で選択する SCI のチャンネルによります。使用しない SCI の端子は、汎用入出力ポートとして使用できます。

表 4.2 使用端子と機能(2)

端子名	入出力	内容
P92/RXD7	入力	SCI7 の受信データ入力端子(注 1)
P90/TXD7	出力	SCI7 の送信データ出力端子(注 1)
PC6/RXD8	入力	SCI8 の受信データ入力端子(注 1)
PC7/TXD8	出力	SCI8 の送信データ出力端子(注 1)
PB6/RXD9	入力	SCI9 の受信データ入力端子(注 1)
PB7/TXD9	出力	SCI9 の送信データ出力端子(注 1)
P81/RXD10	入力	SCI10 の受信データ入力端子(注 1)
P82/TXD10	出力	SCI10 の送信データ出力端子(注 1)
P76/RXD11	入力	SCI11 の受信データ入力端子(注 1)
P77/TXD11	出力	SCI11 の送信データ出力端子(注 1)
PE2/RXD12	入力	SCI12 の受信データ入力端子(注 1)
PE1/TXD12	出力	SCI12 の送信データ出力端子(注 1)

注 1. 使用する SCI の端子は、SCI_Init 関数の第 1 引数で選択する SCI のチャンネルによります。使用しない SCI の端子は、汎用入出力ポートとして使用できます。

5. ソフトウェア説明

リセット解除後、SW3 を押下すると送受信が開始します。

指定バイト数の受信を完了したときは LED1 を反転出力(点灯⇔消灯)、受信中に受信エラーが発生した場合は LED2 を点灯、指定バイト数の送信を完了したときは LED0 を反転出力(点灯⇔消灯)します。

5.1 動作概要

5.1.1 シリアル受信

送受信を開始すると受信動作を許可し、データ受信を検出して受信データの読み出し(注)、指定バイト数を受信完了したときは、SCI の受信動作を禁止にします。受信中に受信エラーを検出すると、SCI の受信動作を禁止します。図 5.1 にシリアル受信のタイミング図を、以下に図中の番号の動作および処理を示します。

- (1) SCI の初期設定を行い、RXD 端子の端子機能を RXD 入力に切り替えます。[初期設定]
- (2) SW3 入力を検出すると、受信状態フラグを“1” (受信開始)に設定します。
- (3) 受信状態フラグの“1”を検出すると、受信状態フラグを“2” (受信ビジー)にして、エラーフラグをクリア、SCR.RIE ビットを“1” (RXI および ERI 割り込み要求を許可)、SCR.RE ビットを“1” (シリアル受信動作を許可)にします。
- (4) 受信データフルの状態を検出すると、RDR レジスタの値を受信バッファに格納します。エラーフラグで受信エラーを確認した場合は、RDR レジスタをダミーリードし、RE ビットを“0” (シリアル受信動作を禁止)、SSR レジスタのエラーフラグをクリア、RIE ビットを“0” (RXI 割り込み要求を禁止)、受信状態フラグを“0”にします。
- (5) 最終データを受信すると、RDR レジスタの値を受信バッファに格納し、RE ビットを“0”、RIE ビットを“0”、受信状態フラグを“0” (受信待機)にします。

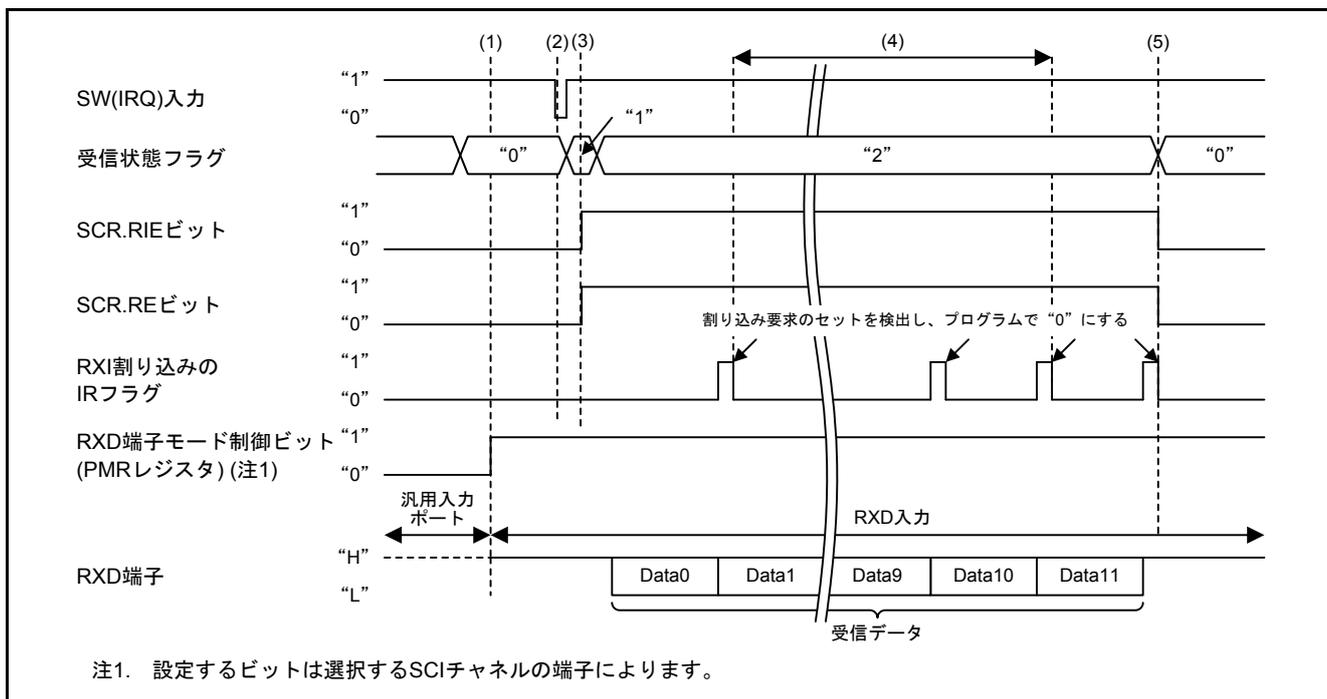


図 5.1 シリアル受信のタイミング図

(注) 動作周波数によっては RDR レジスタ読み出しが追い付かず、オーバランエラーが発生する可能性があります。通信間隔を十分にとるか、適切な動作周波数で使用してください。

5.1.2 シリアル送信

送受信を開始すると送信動作を許可し、送信データエンプティを検出すると送信データを設定、指定バイト数のデータを送信すると SCI の送信動作を禁止します。図 5.2 にシリアル送信のタイミング図を、以下に図中の番号の動作および処理を示します。

- (1) SCI の初期設定を行い、TXD 端子を High 出力にします。[初期設定]
- (2) SW3 入力を検出すると、送信状態フラグを“1” (送信開始)に設定します。
- (3) 送信状態フラグの“1”を検出すると、送信状態フラグを“2” (送信ビジー)にして、SCR.TIE ビットを“1” (TXI 割り込み要求を許可)、SCR.TE ビットを“1” (シリアル送信動作を許可)にします。TE ビットを“1”にすると、TXD 端子が使用可能になるので、TXD 端子モード制御ビットを“1” (周辺機能として使用)にして端子機能を TXD 出力に切り替えます。[送信開始]
- (4) 送信データエンプティの状態を検出すると、次の送信データを設定します。トランスミットデータレジスタ (TDR)に送信バッファの値を書きます。最終データを書いたときは、TIE ビットを“0” (TXI 割り込み要求を禁止)、SCR.TEIE ビットを“1” (TEI 割り込み要求を許可)にします。[データ送信]
- (5) 最終データの送信が完了すると、送信完了の状態になります。送信完了時は、TXD 端子モード制御ビットを“0” (汎用入出力ポートとして使用)にしてから、TE ビットを“0” (シリアル送信動作を禁止)、TEIE ビットを“0” (TEI 割り込み要求を禁止)にします。送信状態フラグは“0” (送信待機)にします。

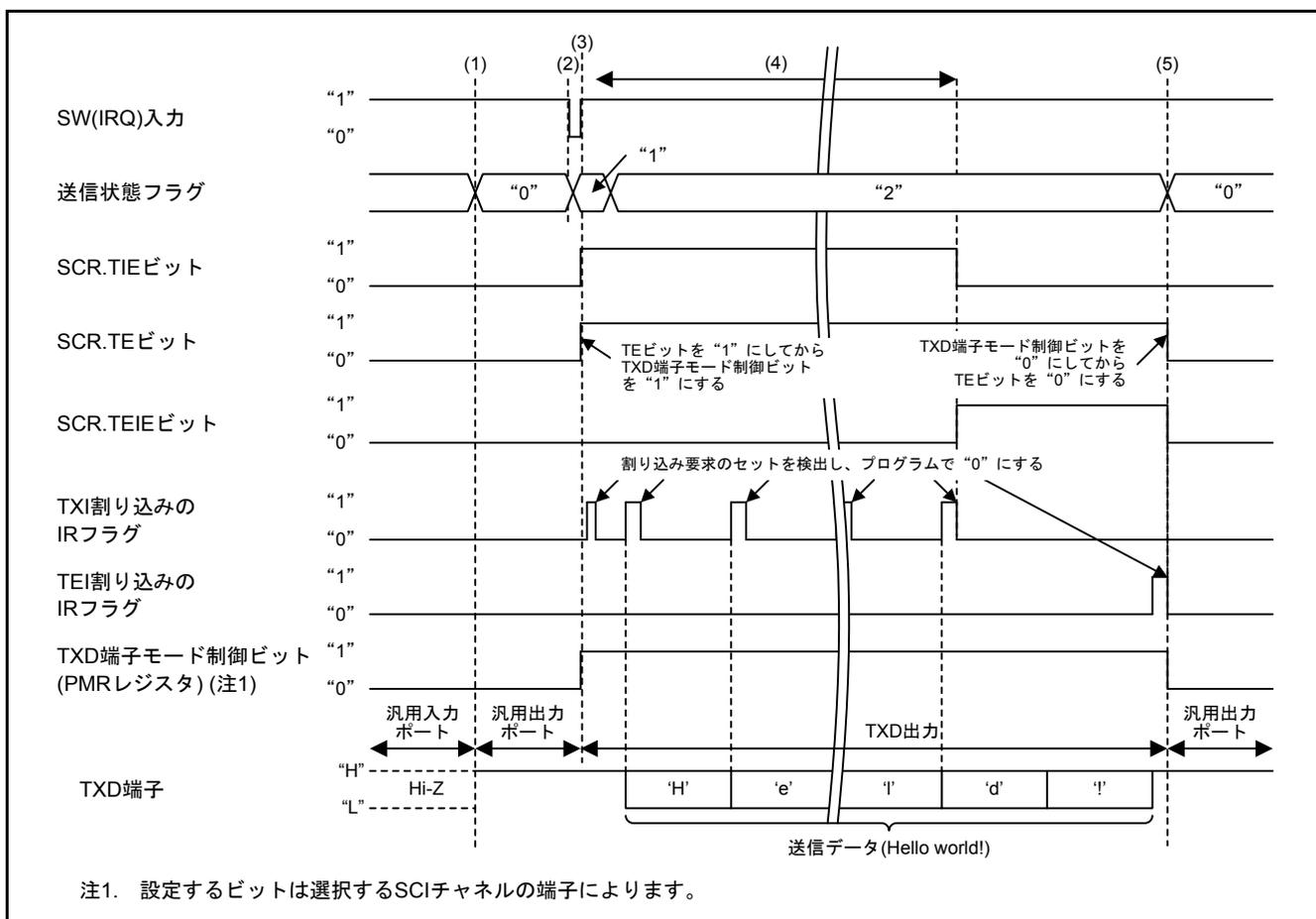


図 5.2 シリアル送信のタイミング図

5.2 ファイル構成

表 5.1 にサンプルコードで使用するファイル、表 5.2 に標準インクルードファイル、表 5.3 に参照する関連アプリケーションノートの関数と設定値を示します。なお、統合開発環境で自動生成されるファイルは除きます。

表 5.1 サンプルコードで使用するファイル

ファイル名	概要
main.c	メイン処理
main.h	main.c のヘッダファイル
sci.c	調歩同期式通信
sci.h	sci.c のヘッダファイル
sci_cfg.h	sci.c のコンフィグレーションヘッダファイル

表 5.2 標準インクルードファイル

ファイル名	概要
stdbool.h	論理型、および論理値に関するマクロを定義します。
stdint.h	指定した幅の整数型を宣言してマクロを定義します。
machine.h	RX ファミリー用組み込み関数の形式を定義します。

表 5.3 参照する関連アプリケーションノートの関数と設定値

(RX63N グループ、RX631 グループ 初期設定例)

ファイル名	関数	設定値
r_init_stop_module.c	R_INIT_StopModule()	DMAC/DTC、EXDMAC、RAM0、RAM1 はモジュールストップ解除
r_init_stop_module.h	-	
r_init_non_existent_port.c	R_INIT_NonExistentPort()	
r_init_non_existent_port.h	-	176 ピン版を指定
r_init_clock.c	R_INIT_Clock()	
r_init_clock.h	-	サブクロックを使用しない

5.3 オプション設定メモリ

表 5.4 にサンプルコードで使用するオプション設定メモリの状態を示します。必要に応じて、お客様のシステムに最適な値を設定してください。

表 5.4 サンプルコードで使用するオプション設定メモリ

シンボル	アドレス	設定値	内容
OFS0	FFFF FF8Fh~FFFF FF8Ch	FFFF FFFFh	リセット後、IWDT は停止 リセット後、WDT は停止
OFS1	FFFF FF8Bh~FFFF FF88h	FFFF FFFFh	リセット後、電圧監視 0 リセット無効 リセット後、HOCO 発振が無効
MDES	FFFF FF83h~FFFF FF80h	FFFF FFFFh	リトルエンディアン

5.4 定数一覧

表 5.5~表 5.9 にサンプルコードで使用する定数を示します。

表 5.5 サンプルコードで使用する定数(main.h)

定数名	設定値	内容
BUF_SIZE	12	バッファサイズ
NULL_SIZE	1	NULL コードサイズ

表 5.6 サンプルコードで使用する定数(main.h)

定数名	設定値	内容
LED0	PORT0.PODR.BIT.B3	LED0 出力データ格納ビット
LED0_PDR	PORT0.PDR.BIT.B3	LED0 方向制御ビット
LED1	PORT0.PODR.BIT.B5	LED1 出力データ格納ビット
LED1_PDR	PORT0.PDR.BIT.B5	LED1 方向制御ビット
LED2	PORT1.PODR.BIT.B0	LED2 出力データ格納ビット
LED2_PDR	PORT1.PDR.BIT.B0	LED2 方向制御ビット
LED3	PORT1.PODR.BIT.B1	LED3 出力データ格納ビット
LED3_PDR	PORT1.PDR.BIT.B1	LED3 方向制御ビット
LED_ON	0	LED 出力データ: 点灯
LED_OFF	1	LED 出力データ: 消灯
SW3_REQ	IR(ICU, IRQ15)	SW3(P07/IRQ15)の IR フラグ
SW_ON	1	SW3 押下検出
SW_OFF	0	SW3 押下未検出

表 5.7 サンプルコードで使用する定数(sci.c)

定数名	設定値	内容
SSR_ERROR_FLAGS	38h	SCI.SSR レジスタのエラーフラグのビットパターン

表 5.8 サンプルコードで使用する定数(sci.h)

定数名	設定値	内容
SCI_CH0	0	SCI チャンネル
SCI_CH1	1	
SCI_CH2	2	
SCI_CH3	3	
SCI_CH4	4	
SCI_CH5	5	
SCI_CH6	6	
SCI_CH7	7	
SCI_CH8	8	
SCI_CH9	9	
SCI_CH10	10	
SCI_CH11	11	
SCI_CH12	12	
SCI_B_TX_BUSY	sci_state.bit.b_tx_busy	送信状態フラグ 0: 送信待機 1: 送信レディ 2: 送信ビジー
SCI_B_RX_BUSY	sci_state.bit.b_rx_busy	受信状態フラグ 0: 受信待機 1: 受信レディ 2: 受信ビジー
SCI_B_RX_ORER	sci_state.bit.b_rx_orer	オーバランエラーフラグ 0: オーバランエラーなし 1: オーバランエラーあり
SCI_B_RX_FER	sci_state.bit.b_rx_fer	フレーミングエラーフラグ 0: フレーミングエラーなし 1: フレーミングエラーあり
SCI_B_RX_PER	sci_state.bit.b_rx_per	パリティエラーフラグ 0: パリティエラーなし 1: パリティエラーあり

表 5.9 サンプルコードで使用する定数(sci_cfg.h)

定数名	設定値	内容
SCI0_PSEL	0	SCI0 の使用端子選択 0 : TXD0 - P20, RXD0 - P21 1 : TXD0 - P32, RXD0 - P33
SCI1_PSEL	0	SCI1 の使用端子選択 0 : TXD1 - P16, RXD1 - P15 1 : TXD1 - P26, RXD1 - P30 2 : TXD1 - PF0, RXD1 - PF2
SCI2_PSEL	0	SCI2 の使用端子選択 0 : TXD2 - P13, RXD2 - P12 1 : TXD2 - P50, RXD2 - P52
SCI3_PSEL	0	SCI3 の使用端子選択 0 : TXD3 - P17, RXD3 - P16 1 : TXD3 - P23, RXD3 - P25
SCI5_PSEL	0	SCI5 の使用端子選択 0 : TXD5 - PA4, RXD5 - PA2 1 : TXD5 - PA4, RXD5 - PA3 2 : TXD5 - PC3, RXD5 - PC2
SCI6_PSEL	0	SCI6 の使用端子選択 0 : TXD6 - P00, RXD6 - P01 1 : TXD6 - P32, RXD6 - P33 2 : TXD6 - PB1, RXD6 - PB0

5.5 構造体/共用体一覧

図 5.3 にサンプルコードで使用する構造体/共用体を示します。

```

typedef struct sci_ch_iodefne /* 各チャンネルの設定情報 */
{
    volatile __evenaccess struct st_sci12 *regs; /* SCIチャンネルのレジスタ構造体のポインタ */
    volatile __evenaccess uint32_t *mstp; /* モジュールストップレジスタのポインタ */
    uint32_t stop_mask; /* モジュールストップレジスタのマスク値 */
    volatile __evenaccess uint8_t *ir_rxi; /* RXI割り込み要求フラグのポインタ */
    volatile __evenaccess uint8_t *ir_txi; /* TXI割り込み要求フラグのポインタ */
    volatile __evenaccess uint8_t *ir_tei; /* TEI割り込み要求フラグのポインタ */
    volatile __evenaccess uint8_t *txd_port_od; /* TXD端子のポート出力データレジスタのポインタ */
    volatile __evenaccess uint8_t *txd_port_d; /* TXD端子のポート方向レジスタのポインタ */
    volatile __evenaccess uint8_t *txd_port_md; /* TXD端子のポートモードレジスタのポインタ */
    uint32_t txd_bit; /* TXD端子のポート関連レジスタのマスク値 */
    volatile __evenaccess uint8_t *txd_pdr_md; /* TXD端子の端子機能選択レジスタのポインタ */
    uint32_t txd_mpc_set; /* TXD端子の端子機能選択レジスタの設定値 */
    volatile __evenaccess uint8_t *rx_d_port_d; /* RXD端子のポート方向レジスタのポインタ */
    volatile __evenaccess uint8_t *rx_d_port_md; /* RXD端子のポートモードレジスタのポインタ */
    uint32_t rx_d_bit; /* RXD端子のポート関連レジスタのマスク値 */
    volatile __evenaccess uint8_t *rx_d_pdr_md; /* RXD端子の端子機能選択レジスタのポインタ */
    uint32_t rx_d_mpc_set; /* RXD端子の端子機能選択レジスタの設定値 */
}sci_ch_iodefne_t;

#pragma bit_order left /* ビットフィールドの並び順指定: 上位ビット側からメンバを割り付ける */
#pragma unpack /* 構造体メンバの境界調整数指定: メンバの型でアライメントする */
typedef union
{
    uint8_t byte;
    struct
    {
        uint8_t b_tx_busy :2; /* 送信ビジーフラグ 0:送信待機 1:送信レディ 2:送信ビジー */
        uint8_t b_rx_busy :2; /* 受信ビジーフラグ 0:送信待機 1:受信レディ 2:受信ビジー */
        uint8_t b_rx_orer :1; /* オーバランエラーフラグ 0:エラーなし 1:オーバランエラー */
        uint8_t b_rx_fer :1; /* フレーミングエラーフラグ 0:エラーなし 1:フレーミングエラー */
        uint8_t b_rx_per :1; /* パリティエラーフラグ 0:エラーなし 1:パリティエラー */
        uint8_t :1; /* 使用しない */
    } bit;
} sci_state_t;
#pragma packoption /* 構造体メンバの境界調整数指定の終了 */
#pragma bit_order /* ビットフィールドの並び順指定の終了 */

```

図 5.3 サンプルコードで使用する構造体/共用体

5.6 変数一覧

表 5.10～表 5.11 にグローバル変数を、表 5.12～表 5.13 に static 型変数を、表 5.14～表 5.41 に SCI チャネル設定情報を示します。

表 5.10 グローバル変数(sci.c)

型	変数名	内容	使用関数
sci_state_t	sci_state	SCI 状態	main SCI_Receive SCI_Transmit sci_rcv_data_full sci_rcv_error sci_tran_end

表 5.11 グローバル変数(sci_cfg.h)

型	変数名	内容	使用関数
sci_ch_iodefint_t	sci_def[13]	各 SCI チャネルの設定情報	SCI_Receive SCI_Transmit sci_rcv_data_full sci_rcv_error sci_tran_end

表 5.12 static 型変数(main.c)

型	変数名	内容	使用関数
static uint8_t	rx_buf[BUF_SIZE]	受信バッファ	main peripheral_init
static uint8_t	tx_buf[]	送信バッファ	main

表 5.13 static 型変数(sci.c)

型	変数名	内容	使用関数
static const uint8_t *	pbuf_tx	送信バッファへのポインタ	SCI_Transmit sci_tran_buffer_empty
static uint8_t	tx_cnt	送信カウンタ	SCI_Transmit sci_tran_buffer_empty
static uint8_t *	pbuf_rx	受信バッファへのポインタ	SCI_Receive sci_rcv_data_full
static uint8_t	rx_cnt	受信カウンタ	SCI_Receive sci_rcv_data_full

表 5.14 SCI チャンネル 0 の設定情報(sci_def[0]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *) &SCI0	SCI チャンネル 0 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 0 モジュールストップレジスタのポインタ
stop_mask	0x80000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI0_RXI0].BYTE	RXI0 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI0_TXI0].BYTE	TXI0 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI0_TEI0].BYTE	TEI0 割り込み要求フラグのポインタ

表 5.15 SCI チャンネル 0 の設定情報(sci_def[0]、SCI0_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT2.PODR.BYTE	TXD0 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT2.PDR.BYTE	TXD0 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT2.PMR.BYTE	TXD0 端子のポートモードレジスタのポインタ
txd_bit	(1<<0)	TXD0 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P20PFS.BYTE	TXD0 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD0 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT2.PDR.BYTE	RXD0 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT2.PMR.BYTE	RXD0 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<1)	RXD0 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P21PFS.BYTE	RXD0 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD0 端子の端子機能選択レジスタの設定値

表 5.16 SCI チャンネル 0 の設定情報(sci_def[0]、SCI0_PSEL = 0 以外を選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT3.PODR.BYTE	TXD0 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT3.PDR.BYTE	TXD0 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT3.PMR.BYTE	TXD0 端子のポートモードレジスタのポインタ
txd_bit	(1<<2)	TXD0 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P32PFS.BYTE	TXD0 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0B	TXD0 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT3.PDR.BYTE	RXD0 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT3.PMR.BYTE	RXD0 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<3)	RXD0 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P33PFS.BYTE	RXD0 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0B	RXD0 端子の端子機能選択レジスタの設定値

表 5.17 SCI チャンネル 1 の設定情報(sci_def[1]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI1	SCI チャンネル 1 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 1 モジュールストップレジスタのポインタ
stop_mask	0x40000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI1_RXI1].BYTE	RXI1 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI1_TXI1].BYTE	TXI1 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI1_TEI1].BYTE	TEI1 割り込み要求フラグのポインタ

表 5.18 SCI チャンネル 1 の設定情報(sci_def[1]、SCI1_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT1.PODR.BYTE	TXD1 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT1.PDR.BYTE	TXD1 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT1.PMR.BYTE	TXD1 端子のポートモードレジスタのポインタ
txd_bit	(1<<6)	TXD1 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P16PFS.BYTE	TXD1 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD1 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT1.PDR.BYTE	RXD1 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT1.PMR.BYTE	RXD1 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<5)	RXD1 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P15PFS.BYTE	RXD1 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD1 端子の端子機能選択レジスタの設定値

表 5.19 SCI チャンネル 1 の設定情報(sci_def[1]、SCI1_PSEL = 1 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT2.PODR.BYTE	TXD1 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT2.PDR.BYTE	TXD1 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT2.PMR.BYTE	TXD1 端子のポートモードレジスタのポインタ
txd_bit	(1<<6)	TXD1 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P26PFS.BYTE	TXD1 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD1 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT3.PDR.BYTE	RXD1 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT3.PMR.BYTE	RXD1 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<0)	RXD1 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P30PFS.BYTE	RXD1 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD1 端子の端子機能選択レジスタの設定値

表 5.20 SCI チャンネル 1 の設定情報(sci_def[1]、SCI1_PSEL = 0,1 以外) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORTF.PODR.BYTE	TXD1 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTF.PDR.BYTE	TXD1 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTF.PMR.BYTE	TXD1 端子のポートモードレジスタのポインタ
txd_bit	(1<<0)	TXD1 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PF0PFS.BYTE	TXD1 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD1 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTF.PDR.BYTE	RXD1 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTF.PMR.BYTE	RXD1 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD1 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PF2PFS.BYTE	RXD1 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD1 端子の端子機能選択レジスタの設定値

表 5.21 SCI チャンネル 2 の設定情報(sci_def[2]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI2	SCI チャンネル 2 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 2 モジュールストップレジスタのポインタ
stop_mask	0x20000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI2_RXI2].BYTE	RXI2 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI2_TXI2].BYTE	TXI2 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI2_TEI2].BYTE	TEI2 割り込み要求フラグのポインタ

表 5.22 SCI チャンネル 2 の設定情報(sci_def[2]、SCI2_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT1.PODR.BYTE	TXD2 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT1.PDR.BYTE	TXD2 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT1.PMR.BYTE	TXD2 端子のポートモードレジスタのポインタ
txd_bit	(1<<3)	TXD2 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P13PFS.BYTE	TXD2 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD2 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT1.PDR.BYTE	RXD2 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT1.PMR.BYTE	RXD2 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD2 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P12PFS.BYTE	RXD2 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD2 端子の端子機能選択レジスタの設定値

表 5.23 SCI チャンネル 2 の設定情報(sci_def[2]、SCI2_PSEL = 0 以外) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT5.PODR.BYTE	TXD2 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT5.PDR.BYTE	TXD2 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT5.PMR.BYTE	TXD2 端子のポートモードレジスタのポインタ
txd_bit	(1<<0)	TXD2 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P50PFS.BYTE	TXD2 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD2 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT5.PDR.BYTE	RXD2 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT5.PMR.BYTE	RXD2 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD2 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P52PFS.BYTE	RXD2 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD2 端子の端子機能選択レジスタの設定値

表 5.24 SCI チャンネル 3 の設定情報(sci_def[3]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI3	SCI チャンネル 3 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCR.B.LONG	チャンネル 3 モジュールストップレジスタのポインタ
stop_mask	0x10000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI3_RXI3].BYTE	RXI3 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI3_TXI3].BYTE	TXI3 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI3_TEI3].BYTE	TEI3 割り込み要求フラグのポインタ

表 5.25 SCI チャンネル 3 の設定情報(sci_def[3]、SCI3_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT1.PODR.BYTE	TXD3 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT1.PDR.BYTE	TXD3 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT1.PMR.BYTE	TXD3 端子のポートモードレジスタのポインタ
txd_bit	(1<<7)	TXD3 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P17PFS.BYTE	TXD3 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0B	TXD3 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT1.PDR.BYTE	RXD3 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT1.PMR.BYTE	RXD3 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<6)	RXD3 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P16PFS.BYTE	RXD3 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0B	RXD3 端子の端子機能選択レジスタの設定値

表 5.26 SCI チャンネル 3 の設定情報(sci_def[3]、SCI3_PSEL = 0 以外) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT2.PODR.BYTE	TXD3 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT2.PDR.BYTE	TXD3 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT2.PMR.BYTE	TXD3 端子のポートモードレジスタのポインタ
txd_bit	(1<<3)	TXD3 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P23PFS.BYTE	TXD3 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD3 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT2.PDR.BYTE	RXD3 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT2.PMR.BYTE	RXD3 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<5)	RXD3 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P25PFS.BYTE	RXD3 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD3 端子の端子機能選択レジスタの設定値

表 5.27 SCI チャンネル 4 の設定情報(sci_def[4]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI4	SCI チャンネル 4 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 4 モジュールストップレジスタのポインタ
stop_mask	0x08000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI4_RXI4].BYTE	RXI4 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI4_TXI4].BYTE	TXI4 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI4_TEI4].BYTE	TEI4 割り込み要求フラグのポインタ
*txd_port_od	&PORTB.PODR.BYTE	TXD4 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTB.PDR.BYTE	TXD4 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTB.PMR.BYTE	TXD4 端子のポートモードレジスタのポインタ
txd_bit	(1<<1)	TXD4 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PB1PFS.BYTE	TXD4 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD4 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTB.PDR.BYTE	RXD4 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTB.PMR.BYTE	RXD4 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<0)	RXD4 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PB0PFS.BYTE	RXD4 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD4 端子の端子機能選択レジスタの設定値

表 5.28 SCI チャンネル 5 の設定情報(sci_def[5]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI5	SCI チャンネル 5 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 5 モジュールストップレジスタのポインタ
stop_mask	0x04000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI5_RXI5].BYTE	RXI5 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI5_TXI5].BYTE	TXI5 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI5_TEI5].BYTE	TEI5 割り込み要求フラグのポインタ

表 5.29 SCI チャンネル 5 の設定情報(sci_def[5]、SCI5_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORTA.PODR.BYTE	TXD5 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTA.PDR.BYTE	TXD5 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTA.PMR.BYTE	TXD5 端子のポートモードレジスタのポインタ
txd_bit	(1<<4)	TXD5 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PA4PFS.BYTE	TXD5 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD5 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTA.PDR.BYTE	RXD5 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTA.PMR.BYTE	RXD5 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD5 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PA2PFS.BYTE	RXD5 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD5 端子の端子機能選択レジスタの設定値

表 5.30 SCI チャンネル 5 の設定情報(sci_def[5]、SCI5_PSEL = 1 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORTA.PODR.BYTE	TXD5 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTA.PDR.BYTE	TXD5 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTA.PMR.BYTE	TXD5 端子のポートモードレジスタのポインタ
txd_bit	(1<<4)	TXD5 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PA4PFS.BYTE	TXD5 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD5 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTA.PDR.BYTE	RXD5 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTA.PMR.BYTE	RXD5 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<3)	RXD5 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PA3PFS.BYTE	RXD5 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD5 端子の端子機能選択レジスタの設定値

表 5.31 SCI チャンネル 5 の設定情報(sci_def[5]、SCI5_PSEL = 0,1 以外) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORTC.PODR.BYTE	TXD5 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTC.PDR.BYTE	TXD5 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTC.PMR.BYTE	TXD5 端子のポートモードレジスタのポインタ
txd_bit	(1<<3)	TXD5 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PC3PFS.BYTE	TXD5 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD5 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTC.PDR.BYTE	RXD5 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTC.PMR.BYTE	RXD5 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD5 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PC2PFS.BYTE	RXD5 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD5 端子の端子機能選択レジスタの設定値

表 5.32 SCI チャンネル 6 の設定情報(sci_def[6]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI6	SCI チャンネル 6 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 6 モジュールストップレジスタのポインタ
stop_mask	0x02000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI6_RXI6].BYTE	RXI6 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI6_TXI6].BYTE	TXI6 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI6_TEI6].BYTE	TEI6 割り込み要求フラグのポインタ

表 5.33 SCI チャンネル 6 の設定情報(sci_def[6]、SCI6_PSEL = 0 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT0.PODR.BYTE	TXD6 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT0.PDR.BYTE	TXD6 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT0.PMR.BYTE	TXD6 端子のポートモードレジスタのポインタ
txd_bit	(1<<0)	TXD6 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P00PFS.BYTE	TXD6 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD6 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT0.PDR.BYTE	RXD6 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT0.PMR.BYTE	RXD6 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<1)	RXD6 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P01PFS.BYTE	RXD6 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD6 端子の端子機能選択レジスタの設定値

表 5.34 SCI チャンネル 6 の設定情報(sci_def[6]、SCI6_PSEL = 1 選択時) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORT3.PODR.BYTE	TXD6 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT3.PDR.BYTE	TXD6 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT3.PMR.BYTE	TXD6 端子のポートモードレジスタのポインタ
txd_bit	(1<<2)	TXD6 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P32PFS.BYTE	TXD6 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD6 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT3.PDR.BYTE	RXD6 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT3.PMR.BYTE	RXD6 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<3)	RXD6 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P33PFS.BYTE	RXD6 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD6 端子の端子機能選択レジスタの設定値

表 5.35 SCI チャンネル 6 の設定情報(sci_def[6]、SCI6_PSEL = 0,1 以外) (sci_cfg.h)

メンバ	設定値	内容
*txd_port_od	&PORTB.PODR.BYTE	TXD6 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTB.PDR.BYTE	TXD6 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTB.PMR.BYTE	TXD6 端子のポートモードレジスタのポインタ
txd_bit	(1<<1)	TXD6 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PB1PFS.BYTE	TXD6 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0B	TXD6 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTB.PDR.BYTE	RXD6 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTB.PMR.BYTE	RXD6 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<0)	RXD6 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PB0PFS.BYTE	RXD6 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0B	RXD6 端子の端子機能選択レジスタの設定値

表 5.36 SCI チャンネル 7 の設定情報(sci_def[7]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI7	SCI チャンネル 7 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 7 モジュールストップレジスタのポインタ
stop_mask	0x01000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI7_RXI7].BYTE	RXI7 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI7_TXI7].BYTE	TXI7 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI7_TEI7].BYTE	TEI7 割り込み要求フラグのポインタ
*txd_port_od	&PORT9.PODR.BYTE	TXD7 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT9.PDR.BYTE	TXD7 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT9.PMR.BYTE	TXD7 端子のポートモードレジスタのポインタ
txd_bit	(1<<0)	TXD7 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P90PFS.BYTE	TXD7 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD7 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT9.PDR.BYTE	RXD7 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT9.PMR.BYTE	RXD7 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD7 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P92PFS.BYTE	RXD7 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD7 端子の端子機能選択レジスタの設定値

表 5.37 SCI チャンネル 8 の設定情報(sci_def[8]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI8	SCI チャンネル 8 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRC.LONG	チャンネル 8 モジュールストップレジスタのポインタ
stop_mask	0x08000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI8_RXI8].BYTE	RXI8 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI8_TXI8].BYTE	TXI8 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI8_TEI8].BYTE	TEI8 割り込み要求フラグのポインタ
*txd_port_od	&PORTC.PODR.BYTE	TXD8 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTC.PDR.BYTE	TXD8 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTC.PMR.BYTE	TXD8 端子のポートモードレジスタのポインタ
txd_bit	(1<<7)	TXD8 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PC7PFS.BYTE	TXD8 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD8 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTC.PDR.BYTE	RXD8 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTC.PMR.BYTE	RXD8 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<6)	RXD8 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PC6PFS.BYTE	RXD8 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD8 端子の端子機能選択レジスタの設定値

表 5.38 SCI チャンネル 9 の設定情報(sci_def[9]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI9	SCI チャンネル 9 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRC.LONG	チャンネル 9 モジュールストップレジスタのポインタ
stop_mask	0x04000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI9_RXI9].BYTE	RXI9 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI9_TXI9].BYTE	TXI9 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI9_TEI9].BYTE	TEI9 割り込み要求フラグのポインタ
*txd_port_od	&PORTB.PODR.BYTE	TXD9 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTB.PDR.BYTE	TXD9 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTB.PMR.BYTE	TXD9 端子のポートモードレジスタのポインタ
txd_bit	(1<<7)	TXD9 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PB7PFS.BYTE	TXD9 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD9 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTB.PDR.BYTE	RXD9 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTB.PMR.BYTE	RXD9 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<6)	RXD9 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PB6PFS.BYTE	RXD9 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD9 端子の端子機能選択レジスタの設定値

表 5.39 SCI チャンネル 10 の設定情報(sci_def[10]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI10	SCI チャンネル 10 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRC.LONG	チャンネル 10 モジュールストップレジスタのポインタ
stop_mask	0x02000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI10_RXI10].BYTE	RXI10 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI10_TXI10].BYTE	TXI10 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI10_TEI10].BYTE	TEI10 割り込み要求フラグのポインタ
*txd_port_od	&PORT8.PODR.BYTE	TXD10 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT8.PDR.BYTE	TXD10 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT8.PMR.BYTE	TXD10 端子のポートモードレジスタのポインタ
txd_bit	(1<<2)	TXD10 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P82PFS.BYTE	TXD10 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD10 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT8.PDR.BYTE	RXD10 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT8.PMR.BYTE	RXD10 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<1)	RXD10 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P81PFS.BYTE	RXD10 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD10 端子の端子機能選択レジスタの設定値

表 5.40 SCI チャンネル 11 の設定情報(sci_def[11]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI11	SCI チャンネル 11 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRC.LONG	チャンネル 11 モジュールストップレジスタのポインタ
stop_mask	0x01000000	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI11_RXI11].BYTE	RXI11 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI11_TXI11].BYTE	TXI11 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI11_TEI11].BYTE	TEI11 割り込み要求フラグのポインタ
*txd_port_od	&PORT7.PODR.BYTE	TXD11 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORT7.PDR.BYTE	TXD11 端子のポート方向レジスタのポインタ
*txd_port_md	&PORT7.PMR.BYTE	TXD11 端子のポートモードレジスタのポインタ
txd_bit	(1<<7)	TXD11 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.P77PFS.BYTE	TXD11 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0A	TXD11 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORT7.PDR.BYTE	RXD11 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORT7.PMR.BYTE	RXD11 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<6)	RXD11 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.P76PFS.BYTE	RXD11 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0A	RXD11 端子の端子機能選択レジスタの設定値

表 5.41 SCI チャンネル 12 の設定情報(sci_def[12]) (sci_cfg.h)

メンバ	設定値	内容
*regs	(volatile __evenaccess struct st_sci12 *)&SCI12	SCI チャンネル 12 のレジスタ構造体のポインタ
*mstp	&SYSTEM.MSTPCRB.LONG	チャンネル 12 モジュールストップレジスタのポインタ
stop_mask	0x00000010	モジュールストップレジスタのマスク値
*ir_rxi	&ICU.IR[IR_SCI12_RXI12].BYTE	RXI12 割り込み要求フラグのポインタ
*ir_txi	&ICU.IR[IR_SCI12_TXI12].BYTE	TXI12 割り込み要求フラグのポインタ
*ir_tei	&ICU.IR[IR_SCI12_TEI12].BYTE	TEI12 割り込み要求フラグのポインタ
*txd_port_od	&PORTE.PODR.BYTE	TXD12 端子のポート出力データレジスタのポインタ
*txd_port_d	&PORTE.PDR.BYTE	TXD12 端子のポート方向レジスタのポインタ
*txd_port_md	&PORTE.PMR.BYTE	TXD12 端子のポートモードレジスタのポインタ
txd_bit	(1<<1)	TXD12 端子のポート関連レジスタのマスク値
*txd_pdr_md	&MPC.PE1PFS.BYTE	TXD12 端子の端子機能選択レジスタのポインタ
txd_mpc_set	0x0C	TXD12 端子の端子機能選択レジスタの設定値
*rx_d_port_d	&PORTE.PDR.BYTE	RXD12 端子のポート方向レジスタのポインタ
*rx_d_port_md	&PORTE.PMR.BYTE	RXD12 端子のポートモードレジスタのポインタ
rx_d_bit	(1<<2)	RXD12 端子のポート関連レジスタのマスク値
*rx_d_pdr_md	&MPC.PE2PFS.BYTE	RXD12 端子の端子機能選択レジスタのポインタ
rx_d_mpc_set	0x0C	RXD12 端子の端子機能選択レジスタの設定値

5.7 関数一覧

表 5.42～表 5.43 に関数を示します。

表 5.42 関数(main.c)

関数名	概要
main	メイン処理
port_init	ポート初期設定
peripheral_init	周辺機能初期設定
IRQ_Init	IRQ 初期設定

表 5.43 関数(sci.c)

関数名	概要
SCI_Init	SCI 初期設定処理
SCI_Receive	SCI 受信処理
SCI_Transmit	SCI 送信処理
sci_tran_buffer_empty	送信データエンプティ時の処理
sci_tran_end	送信終了時の処理
sci_recv_data_full	受信データフル時の処理
sci_recv_error	受信エラー時の処理

5.8 関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	初期設定後、SW3 入力を検出して SCI の送受信を開始します。
引数	なし
リターン値	なし

port_init	
概要	ポート初期設定
ヘッダ	なし
宣言	void port_init(void)
説明	ポートの初期設定を行います。
引数	なし
リターン値	なし

peripheral_init	
概要	周辺機能初期設定
ヘッダ	なし
宣言	void peripheral_init(void)
説明	使用する周辺機能の初期設定、RAM の初期化を行います。
引数	なし
リターン値	なし

IRQ_Init	
概要	IRQ 初期設定
ヘッダ	なし
宣言	static void IRQ_Init(void)
説明	IRQ15 の初期設定を行います。
引数	なし
リターン値	なし

SCI_Init	
概要	SCI 初期設定処理
ヘッダ	sci.h
宣言	void SCI_Init (uint8_t sci_ch)
説明	SCI の初期設定を行います。
引数	uint8_t sci_ch : 制御する SCI チャンネル
リターン値	なし
SCI_Receive	
概要	SCI 受信処理
ヘッダ	sci.h
宣言	void SCI_Receive (uint8_t sci_ch, uint8_t * pbuf, uint8_t num)
説明	SCI 受信を開始します。
引数	uint8_t sci_ch : 制御する SCI チャンネル uint8_t * pbuf : 受信データ格納ポインタ uint8_t num : 受信バイト数
リターン値	なし
SCI_Transmit	
概要	SCI 送信処理
ヘッダ	sci.h
宣言	void SCI_Transmit (uint8_t sci_ch, const uint8_t * pbuf, uint8_t num)
説明	SCI 送信を開始します。
引数	uint8_t sci_ch : 制御する SCI チャンネル const uint8_t * pbuf : 送信データ格納ポインタ uint8_t num : 送信バイト数
リターン値	なし
sci_rcv_data_full	
概要	受信データフル時の処理
ヘッダ	なし
宣言	static void sci_rcv_data_full (uint8_t sci_ch)
説明	受信データフルの状態呼び出されます。受信データを格納します。最終データを受信すると、シリアル受信を禁止します。
引数	uint8_t sci_ch : 制御する SCI チャンネル
リターン値	なし
備考	この処理は、1 フレームのデータ通信が完了するまでに一括して処理してください。

sci_recv_error

概要	受信エラー時の処理
ヘッダ	なし
宣言	static void sci_recv_error (uint8_t sci_ch)
説明	SCI 受信エラー発生を検出した際に呼び出されます。
引数	uint8_t sci_ch : 制御する SCI チャンネル
リターン値	なし

sci_tran_buffer_empty

概要	送信データエンプティ時の処理
ヘッダ	なし
宣言	static void sci_tran_buffer_empty (uint8_t sci_ch)
説明	送信データエンプティの状態呼び出されます。送信データを書き込みます。最終データを送信すると、TXI 割り込み要求を禁止し、TEI 割り込み要求を許可します。
引数	uint8_t sci_ch : 制御する SCI チャンネル
リターン値	なし
備考	この処理は、1 フレームのデータ通信が完了するまでに一括して処理してください。

sci_tran_end

概要	送信終了時の処理
ヘッダ	なし
宣言	static void sci_tran_end (uint8_t sci_ch)
説明	送信終了の状態呼び出されます。シリアル送信を禁止します。
引数	uint8_t sci_ch : 制御する SCI チャンネル
リターン値	なし

5.9 フローチャート

5.9.1 メイン処理

図 5.4 にメイン処理のフローチャートを示します。

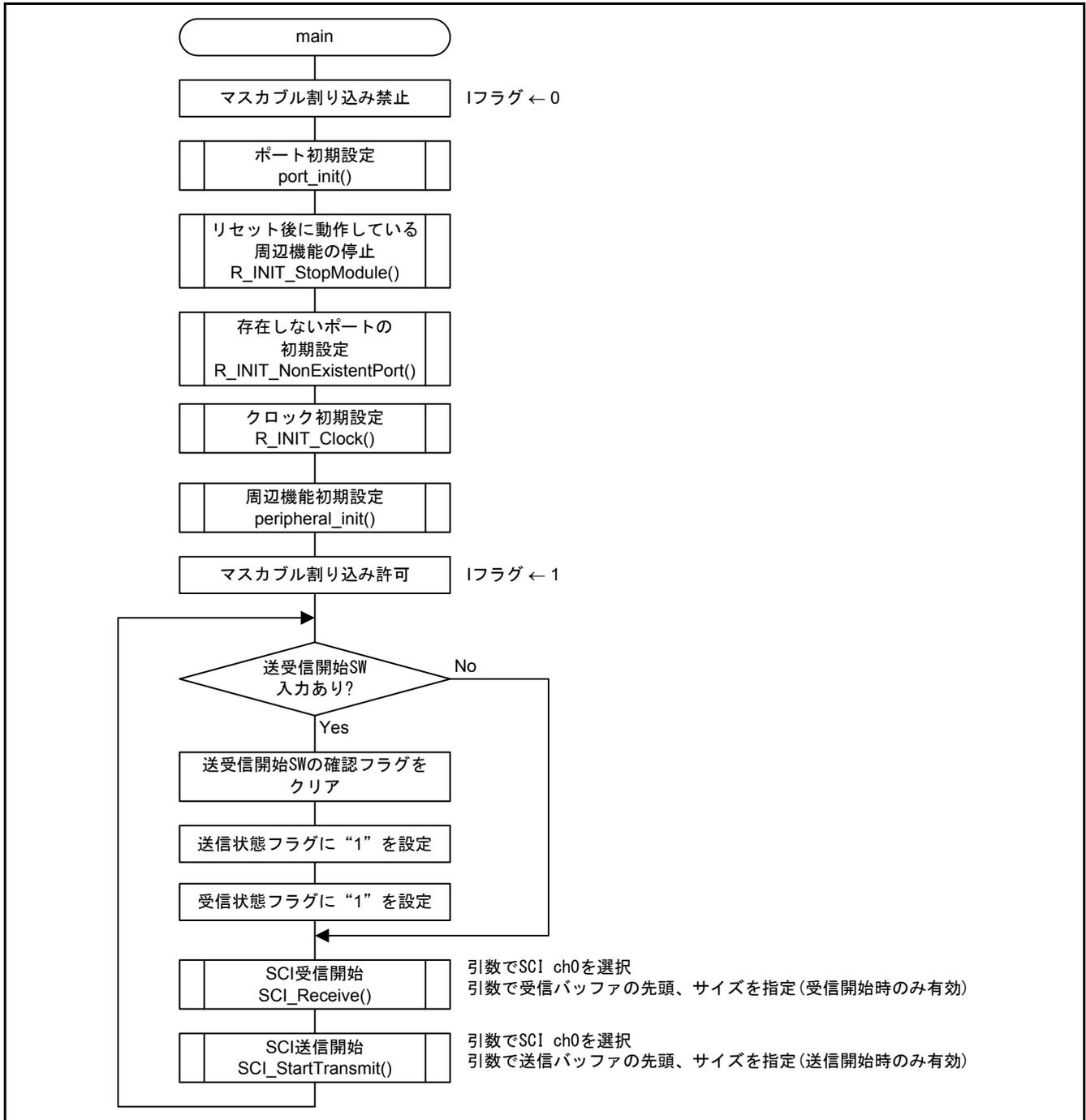


図 5.4 メイン処理

5.9.2 ポート初期設定

図 5.5 にポート初期設定のフローチャートを示します。

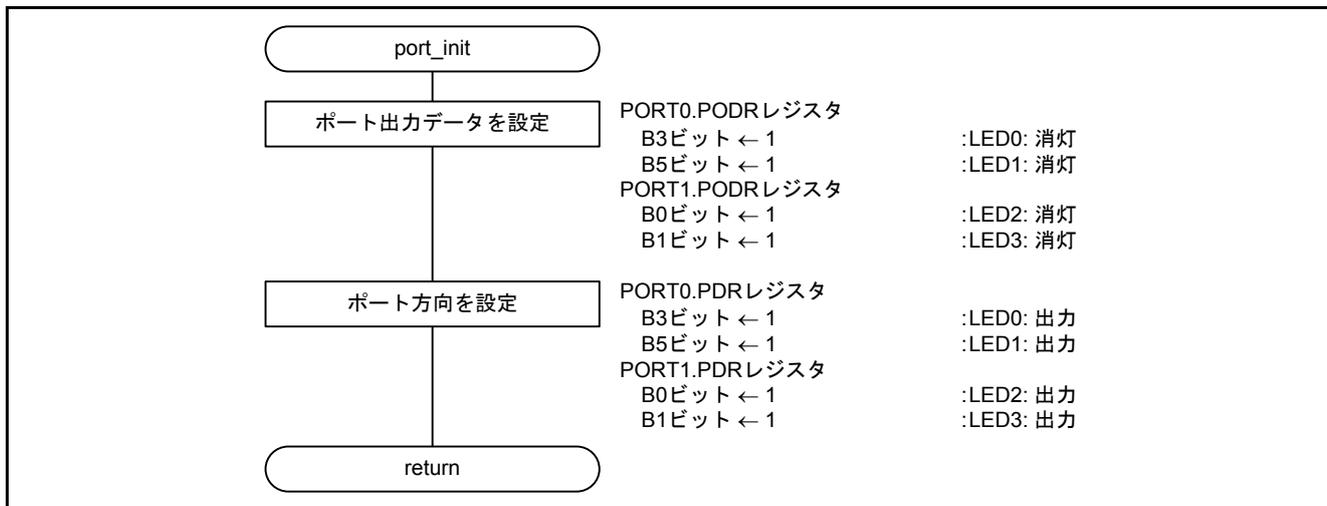


図 5.5 ポート初期設定

5.9.3 周辺機能初期設定

図 5.6 に周辺機能初期設定のフローチャートを示します。

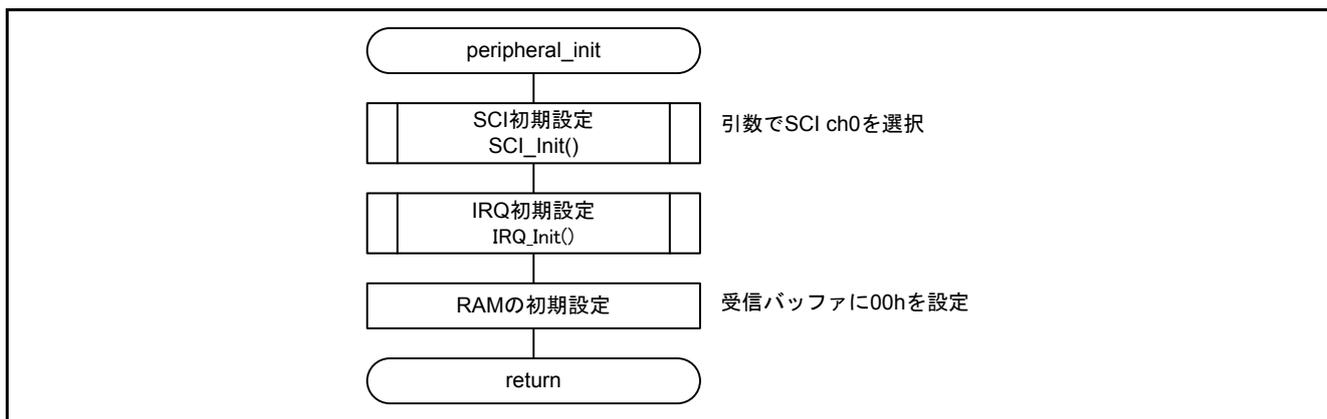


図 5.6 周辺機能初期設定

5.9.4 IRQ 初期設定

図 5.7 に IRQ 初期設定のフローチャートを示します。

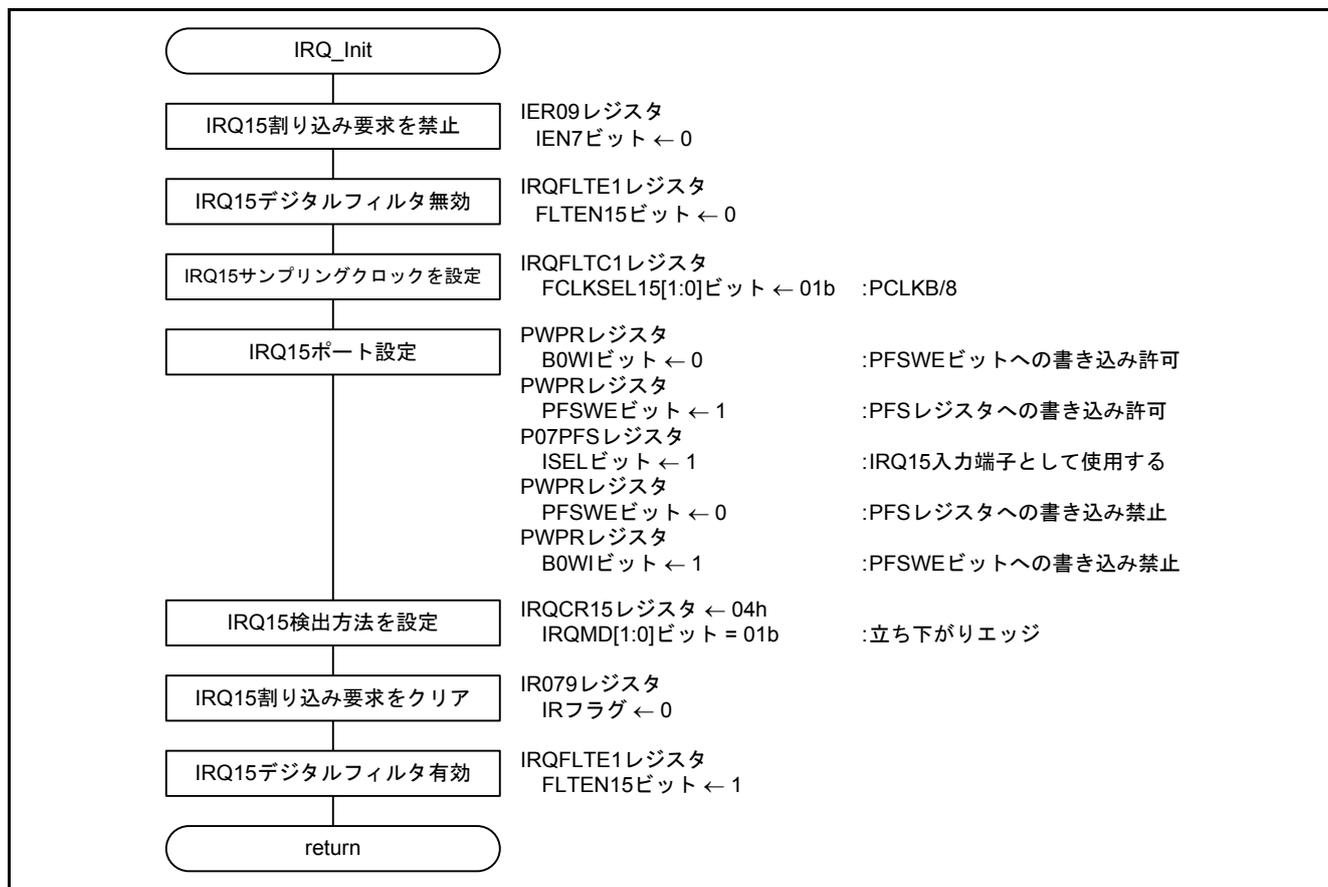


図 5.7 IRQ 初期設定

5.9.5 SCI 初期設定

図 5.8、図 5.9 に SCI 初期設定のフローチャートを示します。

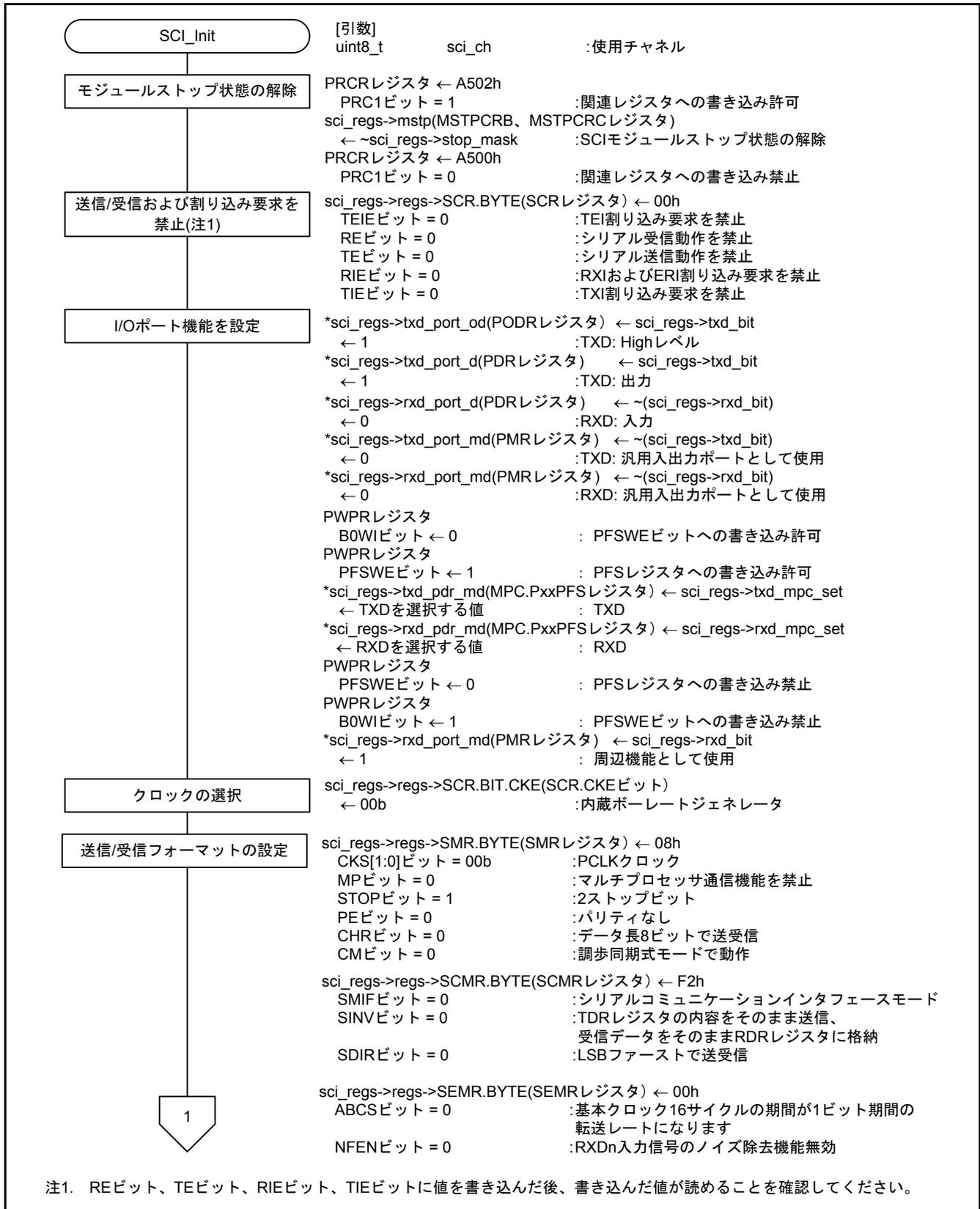


図 5.8 SCI 初期設定(1)

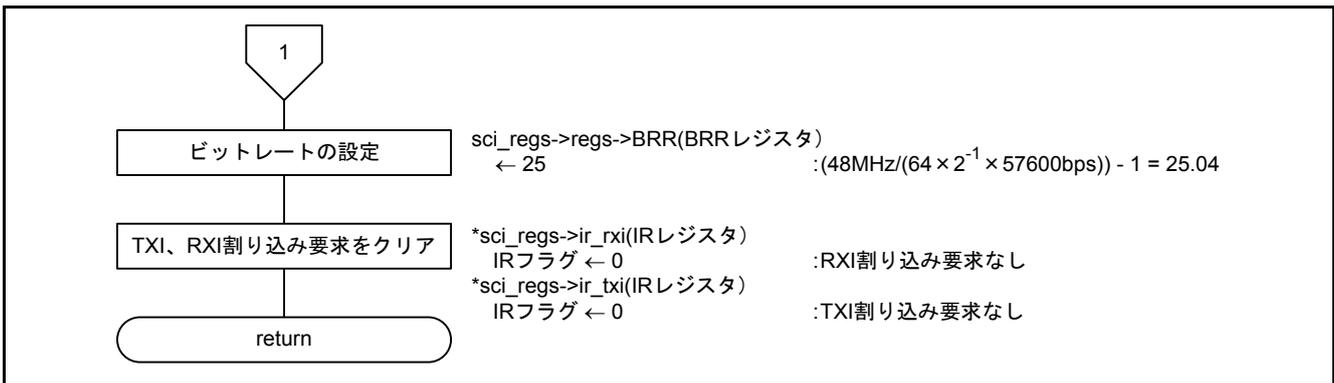


図 5.9 SCI 初期設定(2)

5.9.6 SCI 受信

図 5.10 に SCI 受信のフローチャートを示します。

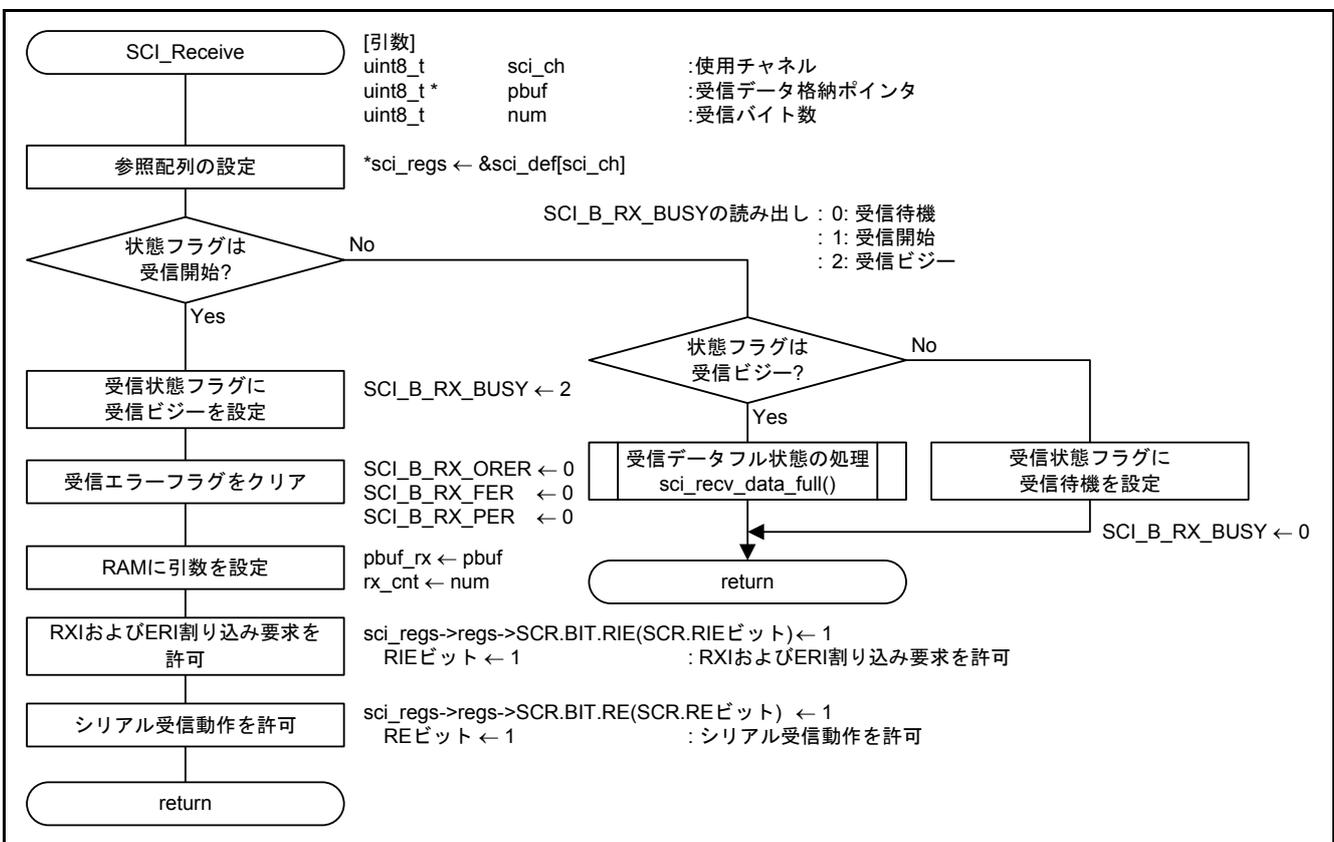


図 5.10 SCI 受信

5.9.7 SCI 送信

図 5.11 に SCI 送信のフローチャートを示します。

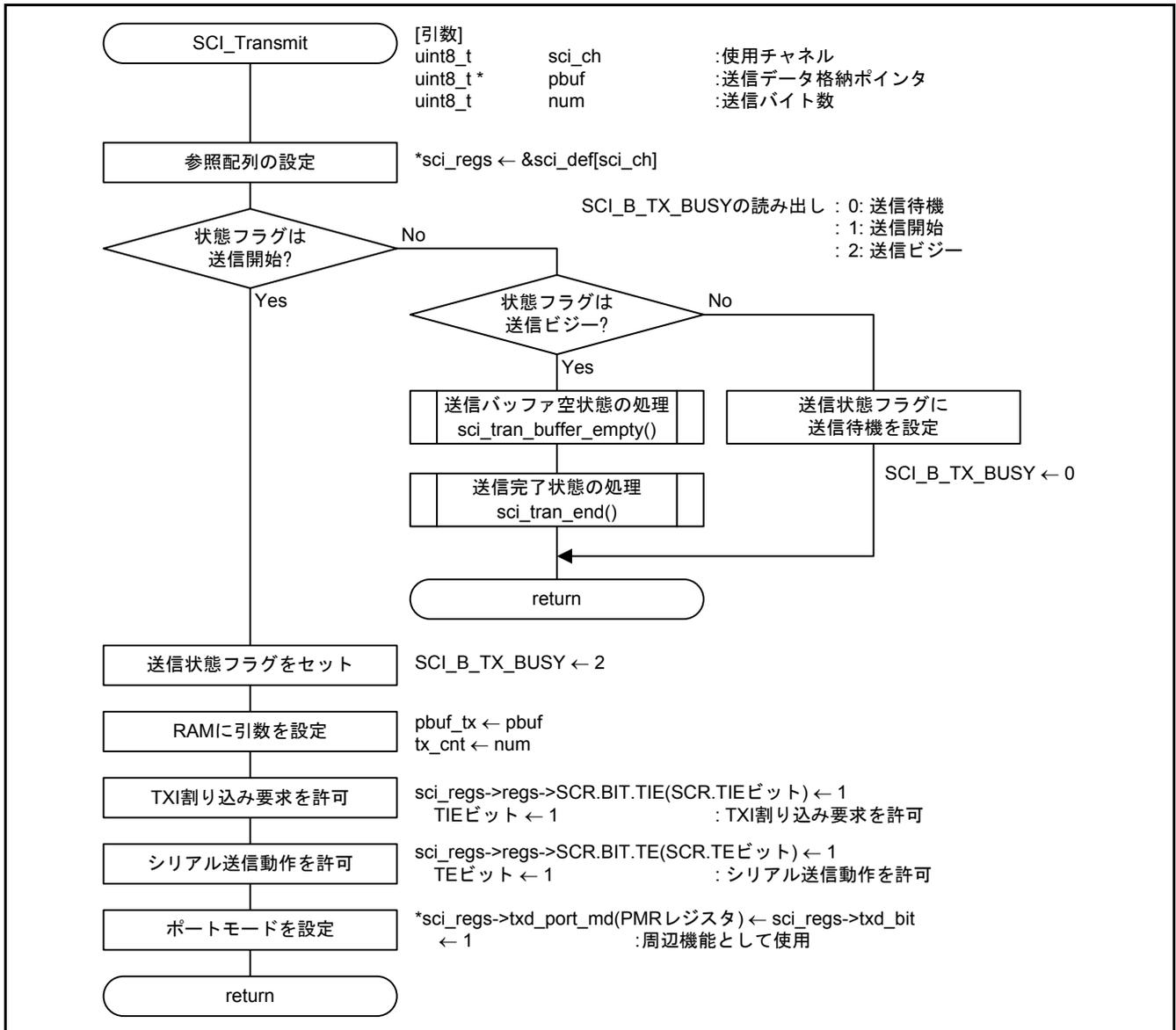


図 5.11 SCI 送信

5.9.8 受信データフル時の処理

図 5.12、図 5.13 に受信データフル時の処理のフローチャートを示します。

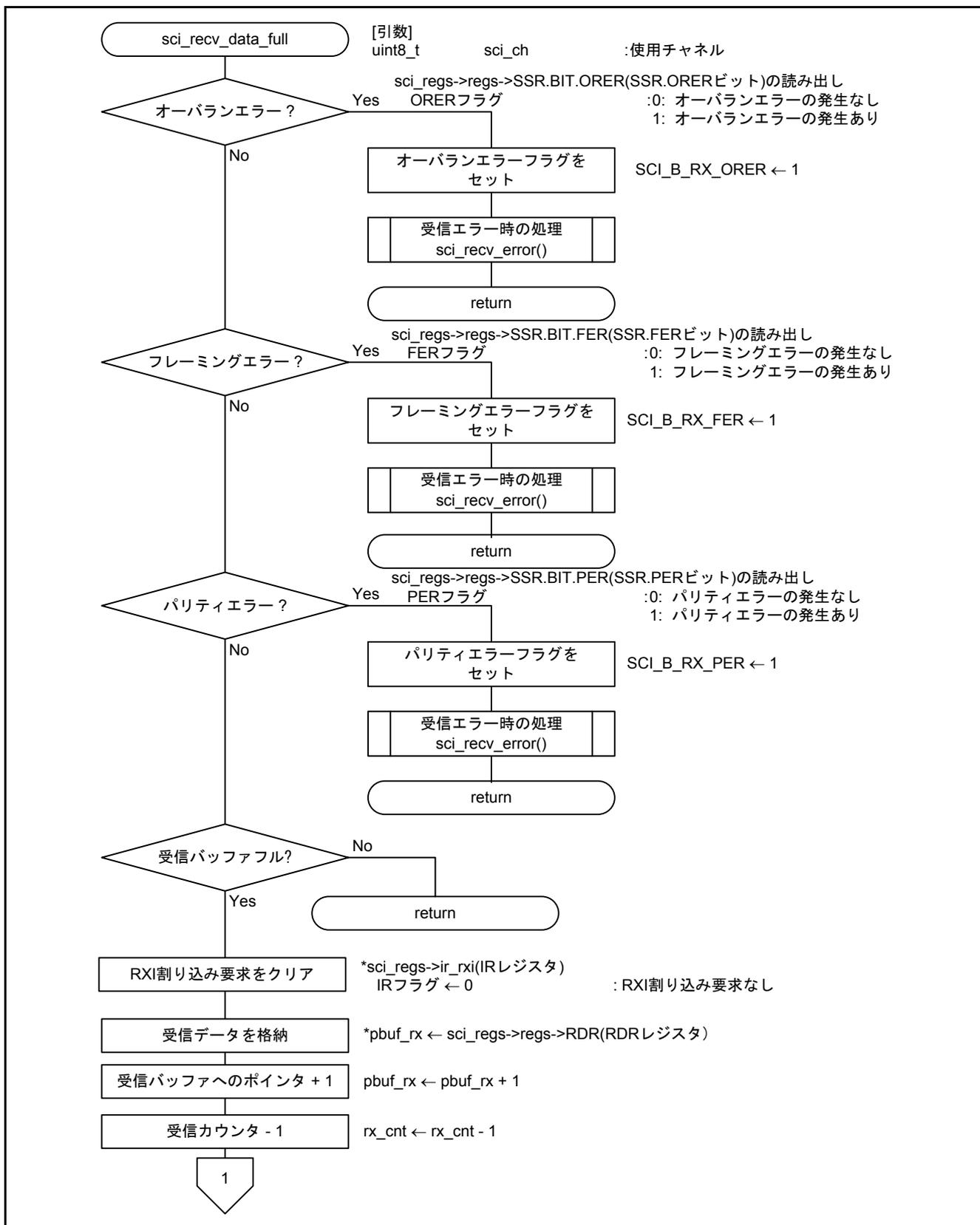


図 5.12 受信データフル時の処理(1)

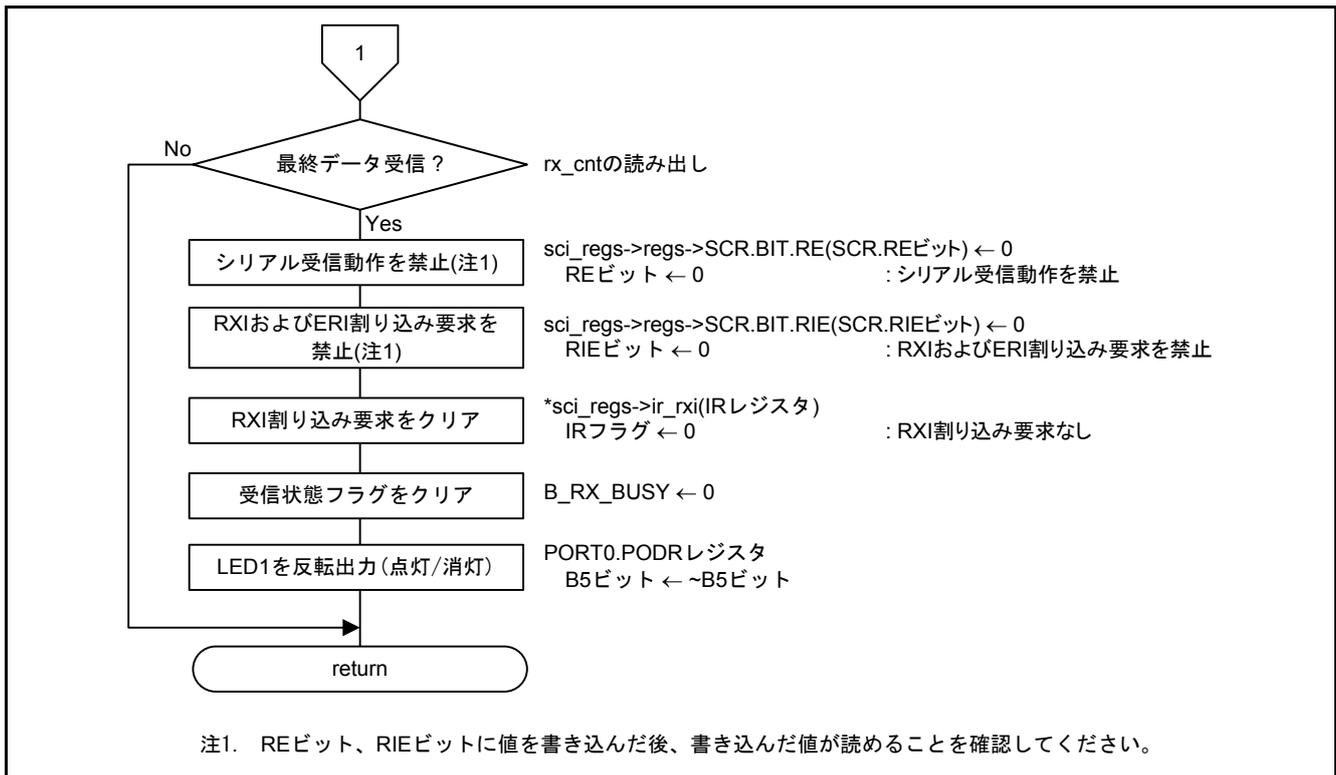


図 5.13 受信データフル時の処理(2)

5.9.9 受信エラー時の処理

図 5.14 に受信エラー時の処理のフローチャートを示します。

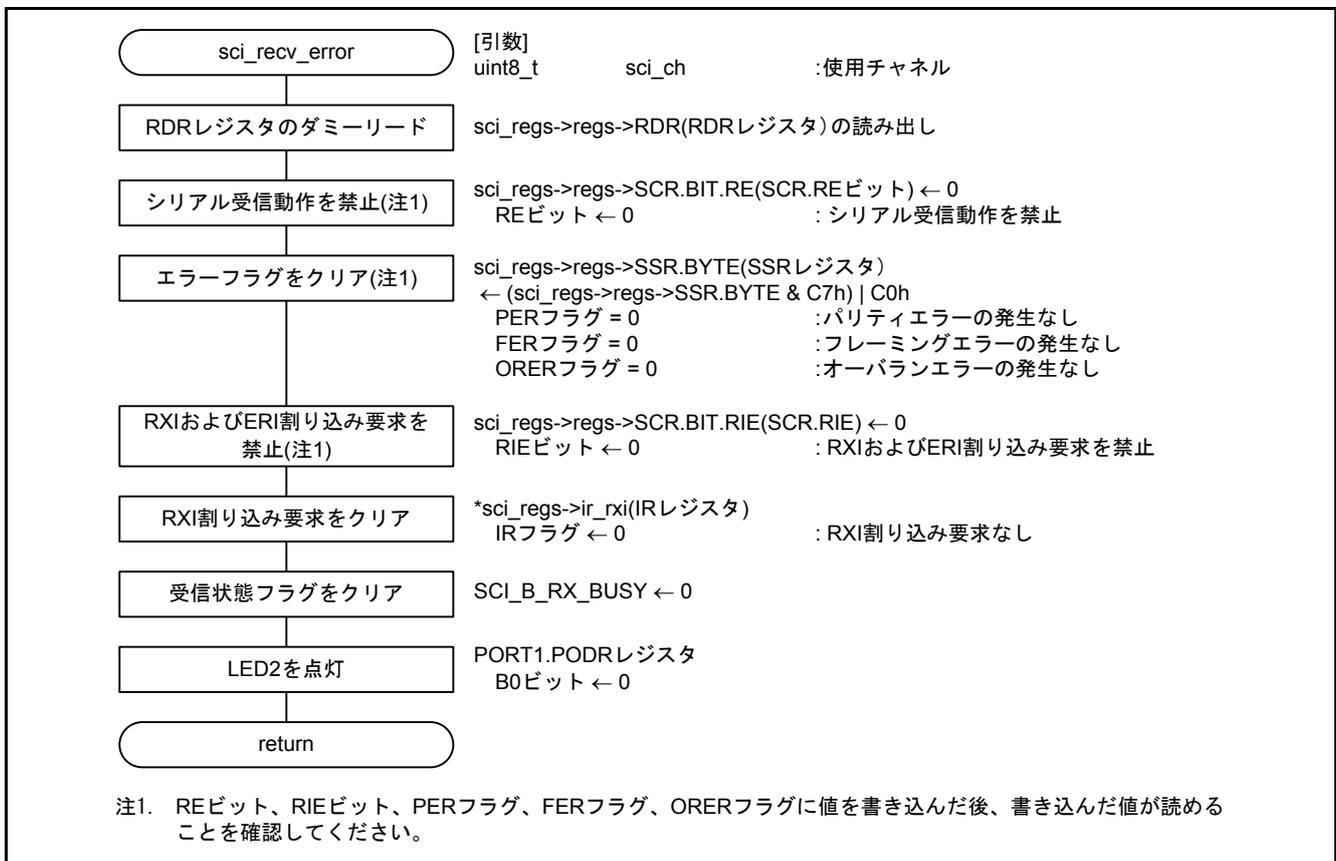


図 5.14 受信エラー時の処理

5.9.10 送信データエンプティ時の処理

図 5.15 に送信データエンプティ時の処理のフローチャートを示します。

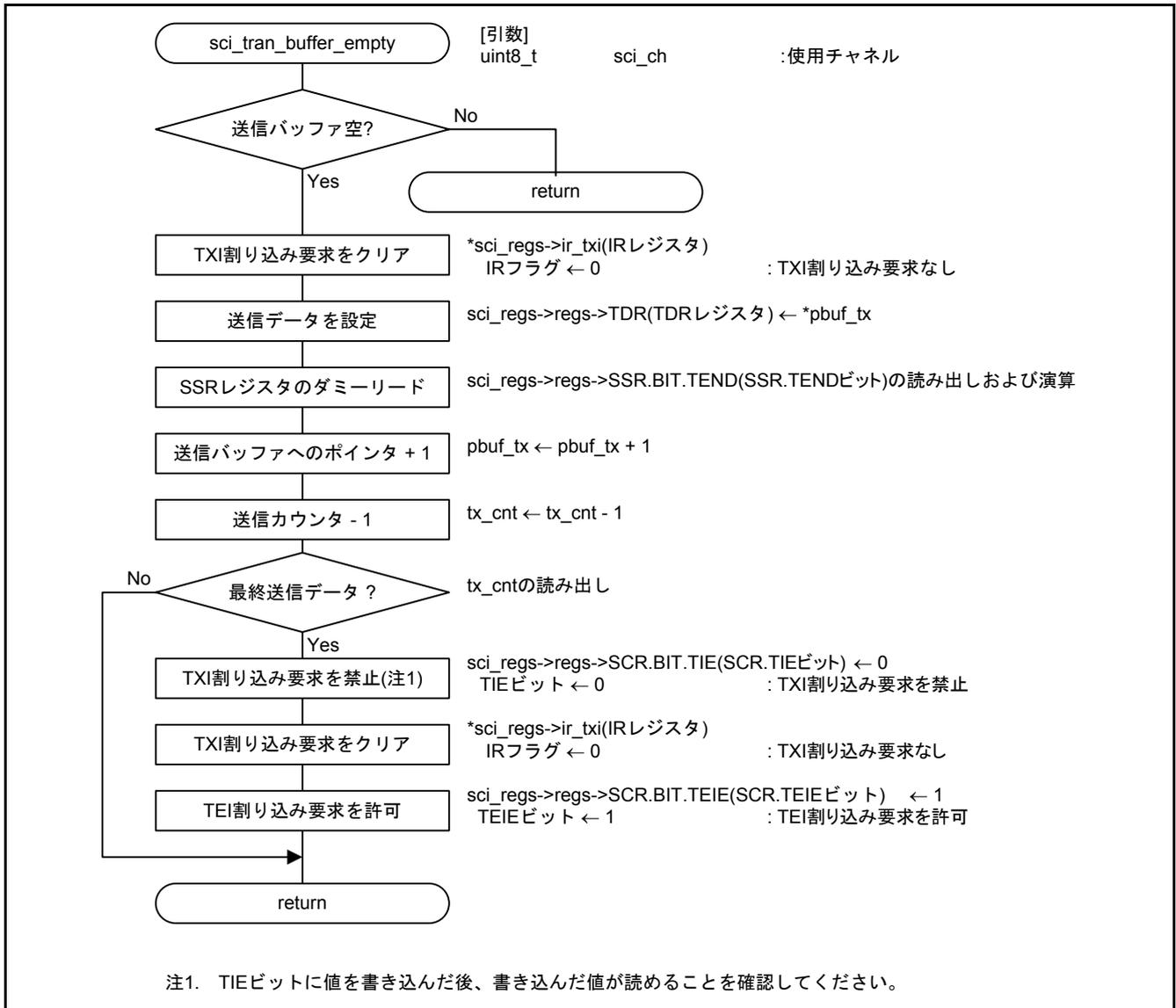


図 5.15 送信データエンプティ時の処理

5.9.11 送信終了時の処理

図 5.16 に送信終了時の処理のフローチャートを示します。

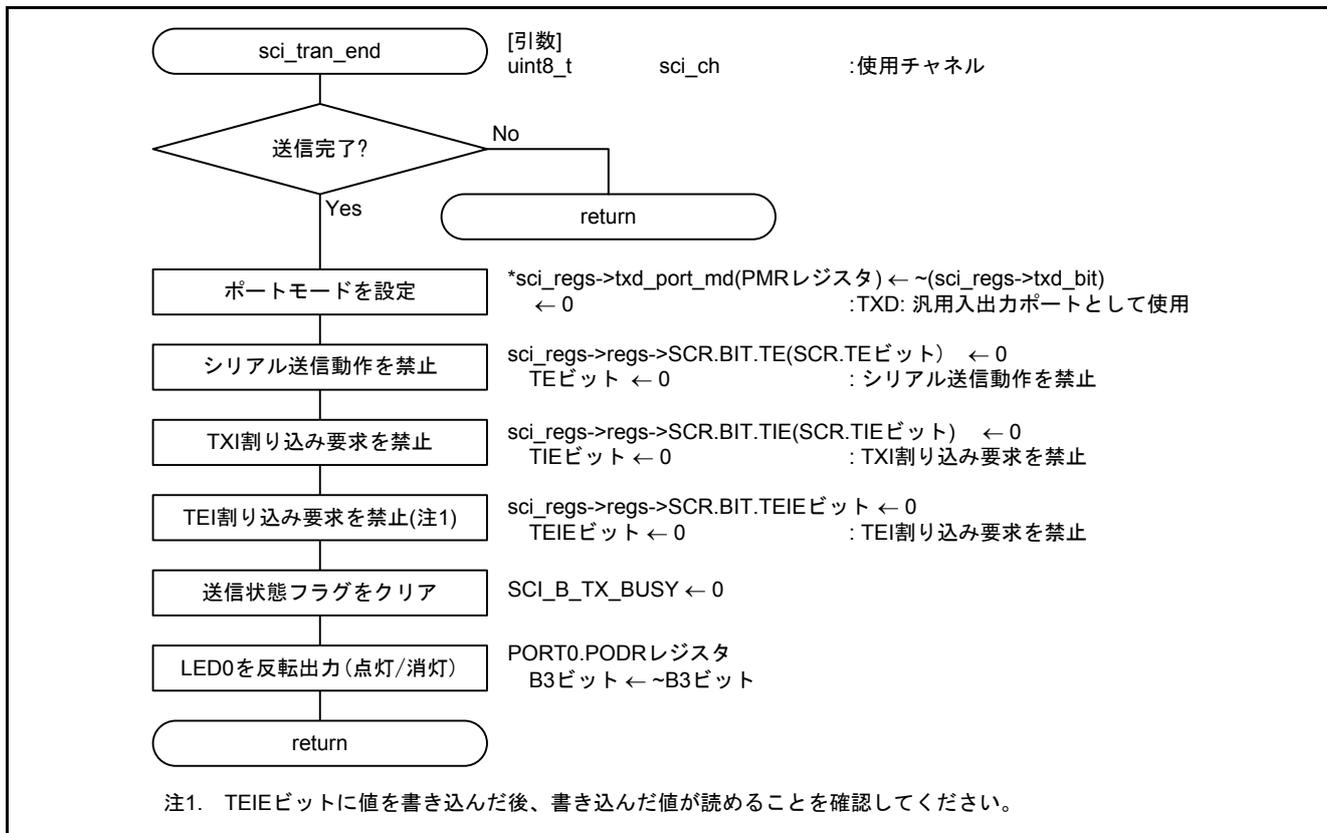


図 5.16 送信終了時の処理

6. サンプルコード

サンプルコードは、ルネサス エレクトロニクスホームページから入手してください。

7. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX63N グループ、RX631 グループ ユーザーズマニュアル ハードウェア編 Rev.1.80 (R01UH0041JJ)
(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

ユーザーズマニュアル：開発環境

RX ファミリ C/C++コンパイラパッケージ V.1.01 ユーザーズマニュアル Rev.1.00 (R20UT0570JJ)
(最新版をルネサス エレクトロニクスホームページから入手してください。)

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com>

お問合せ先

<http://japan.renesas.com/contact/>

改訂記録	RX63N グループ、RX631 グループ 割り込みを使用しない調歩同期式 SC1c 送受信
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2015.04.01	—	初版発行

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。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。

外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレス（予約領域）のアクセス禁止

【注意】リザーブアドレス（予約領域）のアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違っていると、内部 ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が異なる製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 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ものです。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。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、
家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、
防災・防犯装置、各種安全装置等
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じて、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っていません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問い合わせください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事情報に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサス エレクトロニクス株式会社

営業お問い合わせ窓口

<http://www.renesas.com>

営業お問い合わせ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒100-0004 千代田区大手町2-6-2 (日本ビル)

技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問い合わせ窓口：<http://japan.renesas.com/contact/>