

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。
5. 輸出に際しては、「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。本資料に記載されている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用途の目的で使用しないでください。また、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に使用することができません。
6. 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。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されてい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生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
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
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、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社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。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、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おりません。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、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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合せください。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。
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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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。

H8S/20103, H8S/20203, H8S/20223 グループ

ELC を使用した SCI3 フレーム送受信

要旨

H8S/20103, H8S/20203, H8S/20223 グループ内蔵の ELC (イベントリンクコントローラ) を使用して, DTC (データトランスファコントローラ) と SCI3 送受信をリンクさせ, CPU の介在なくフレーム送信, フレーム受信を実現します。

対象デバイス

H8S/20103 (R4F20103)

H8S/20203 (R4F20203)

H8S/20223 (R4F20223)

動作確認条件

システムクロック $\phi = \phi_{osc} = 20 \text{ MHz}$

目次

1. 仕様	2
2. 使用機能説明	4
3. 動作原理	15
4. ソフトウェア説明	19
5. フローチャート	23
6. プログラムリスト	32

1. 仕様

本アプリケーションにおける仕様を以下に説明します。ELC を使用した SCI3_1 フレーム送信を図 1、フレーム受信を図 2、フレーム送受信動作概要を図 3 に示します。

- (1) SCI3_1 の送信データテーブルを作成し、ROM に配置します。
- (2) DTC レジスタ情報の先頭アドレスを DTC ベクタテーブルの起動要因に対応する番地に格納します。
- (3) SCI3_1 を調歩同期式通信、ビットレートを 9600 bps、偶数パリティ、1 ストップビットに設定し、送受信を許可します。
- (4) DTC を通常モード、転送サイズを 1 バイトに設定します。
- (5) 転送元アドレスを SCI3_1 の送信データテーブルに設定し、転送先アドレスを SCI3_1 の TDR アドレスに設定します。
- (6) DTC 起動要因を ELSR30 イベントに設定します。
- (7) DTC のリンク元イベント信号を SCI3_1・送信データエンプティに設定します。
- (8) ELSR30 イベント割り込みを許可します。
- (9) イベントリンクを許可します。
- (10) ソフトウェアで SCI3_1 の送信データテーブルの開始 1 バイトを送信します。
- (11) I ビットをクリアし、割り込みを許可します。
- (12) SCI3_1・送信データエンプティが発生するごとに、CPU の介在なく DTC を起動し、送信データを SCI3_1 の TDR にライトされ、連続送信します。
- (13) 32 バイトのフレーム送信が終了すると、送信を禁止し、フレーム受信に切り替えます。
- (14) 転送元アドレスを SCI3_1 の RDR アドレス、転送先アドレスを受信バッファ先頭アドレスに設定します。
- (15) DTC のリンク元イベント信号を SCI3_1・受信データフルに設定します。
- (16) SCI3_1・受信データフルが発生するごとに、CPU の介在なく DTC を起動し、受信データを受信バッファに格納し、連続受信します。
- (17) 32 バイトのフレーム受信が終了すると、受信を禁止します。

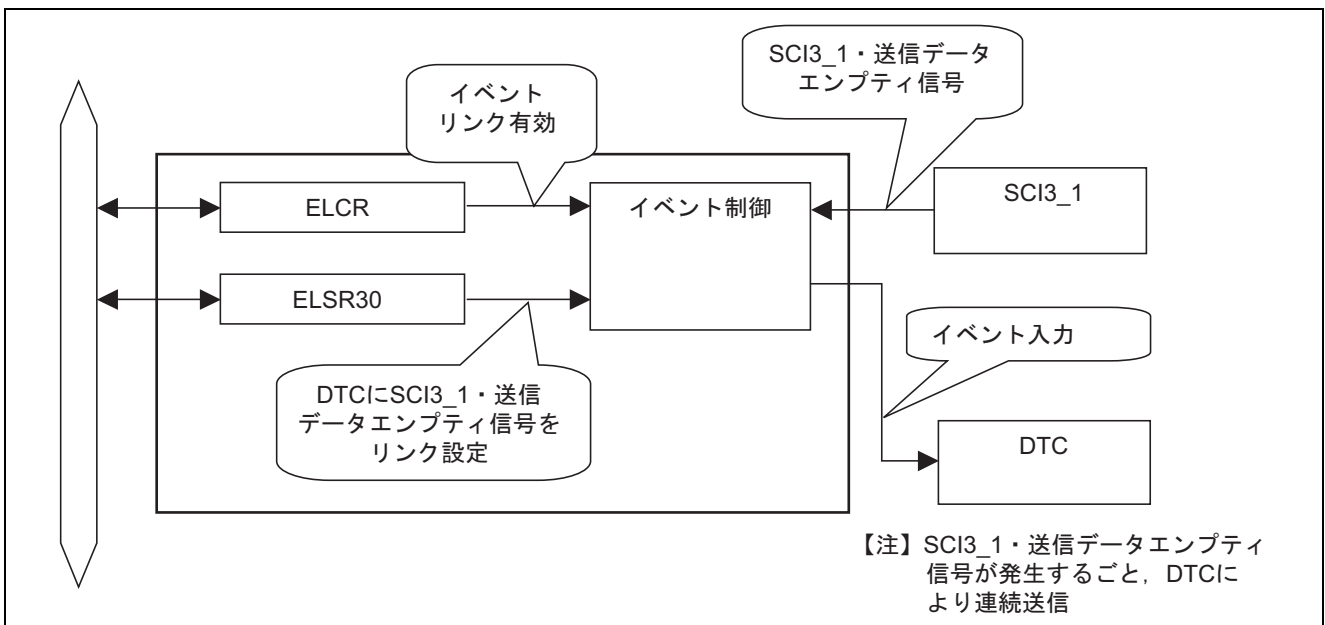


図 1 ELC を使用した SCI3_1 フレーム送信の概要

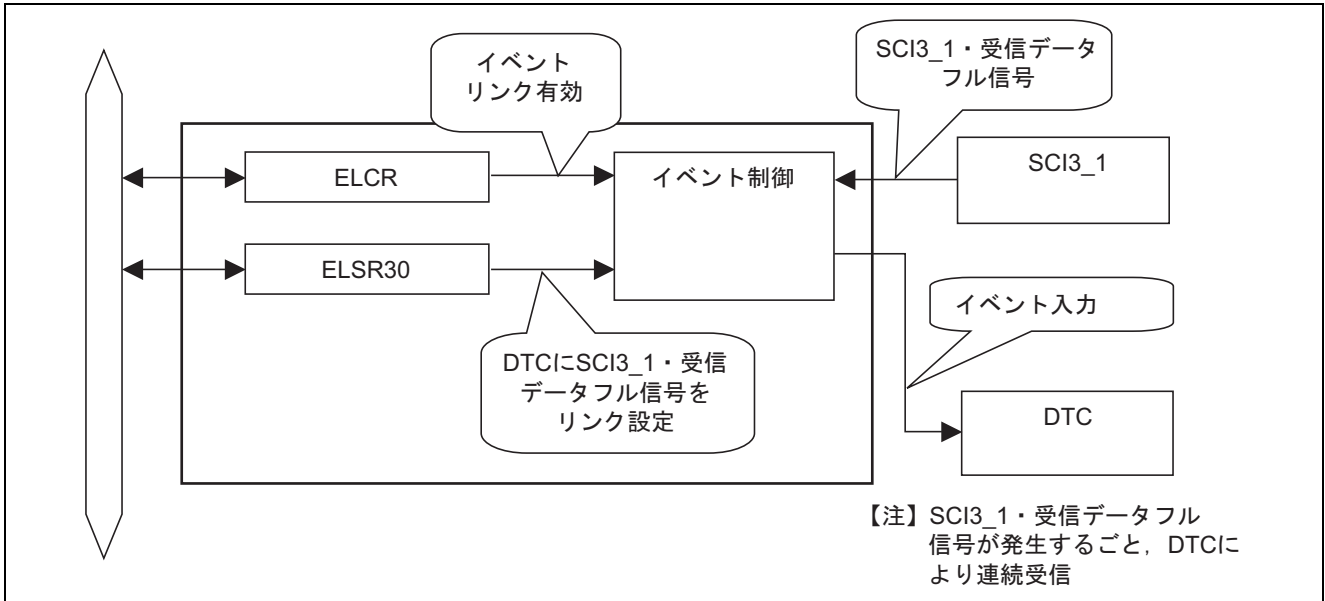


図 2 ELC を使用した SCI3_1 フレーム受信の概要

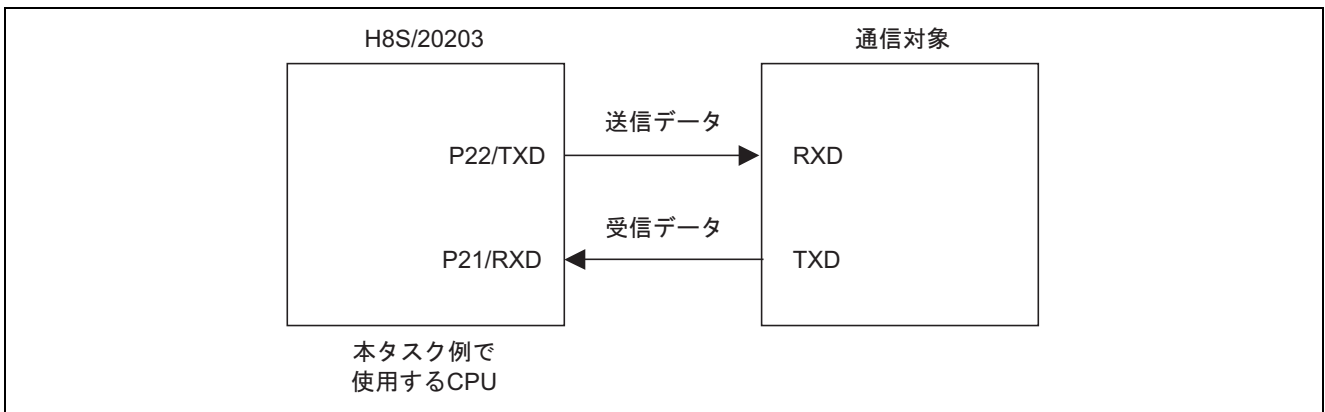


図 3 ELC を使用した SCI3_1 フレーム送受信の動作概要

2. 使用機能説明

2.1 ELC (イベントリンクコントローラ)

以下に、ELC の機能を説明します。イベントリンクコントローラのブロック図を図 4 に示します。

ELC は、各周辺モジュールを出力するイベントをモジュール間で相互に接続 (リンク) します。イベントリンクにより、CPU を介さず直接モジュール間での連携動作が可能になります。

- 59 種類のイベント信号を直接モジュールへリンク可能
- タイマ系のモジュールは、イベント入力時の動作の選択が可能
- ポート 3, ポート 6 へのイベントリンクが可能
- ポートの設定により、ポートでのイベント発生が可能
- イベントを接続するポートは、1 ビット単位または任意の複数ビットをグループ化して指定可能
- イベント発生タイマにより、4 チャンネルの任意設定周期のイベント発生が可能

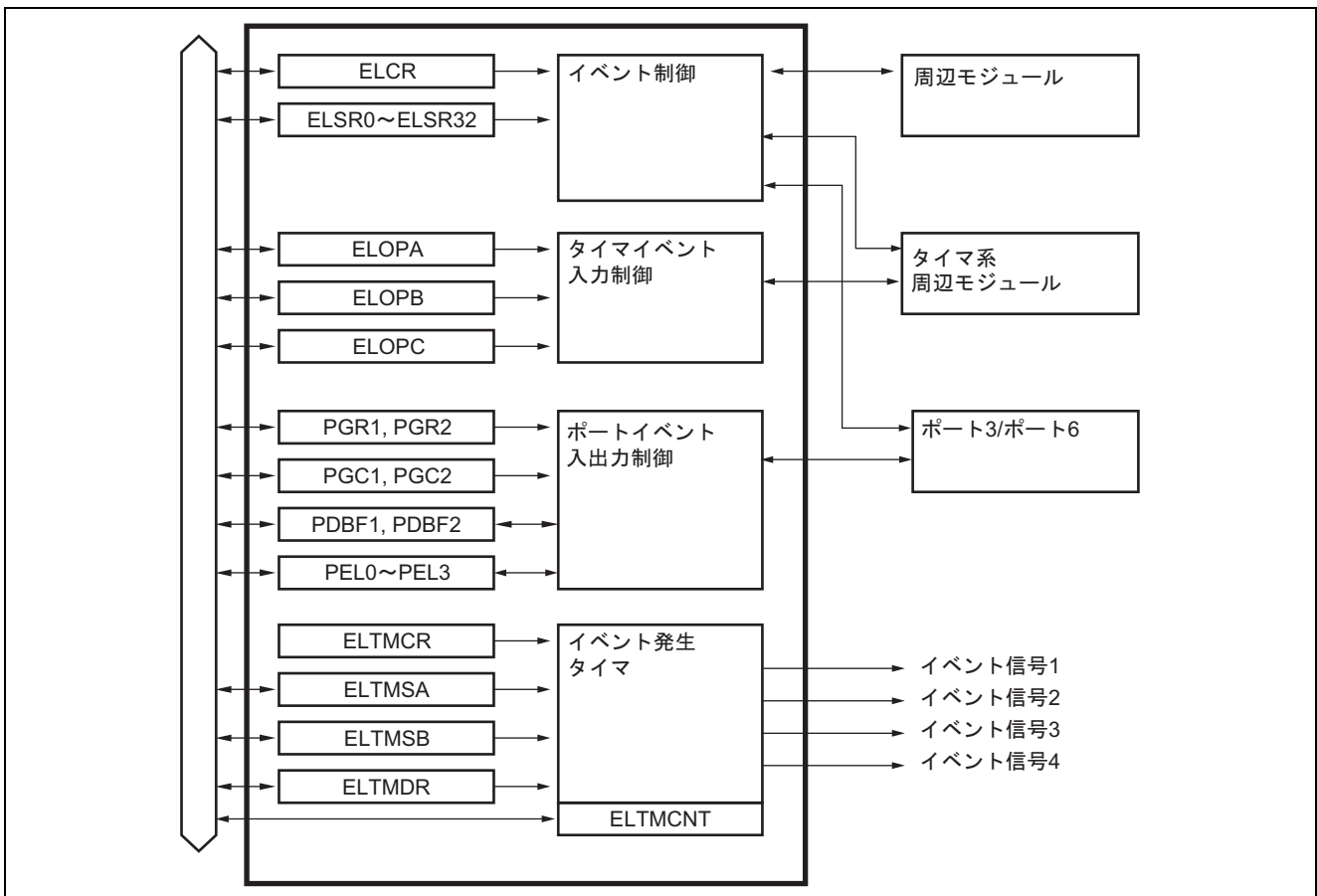


図 4 イベントリンクコントローラのブロック図

2.2 シリアルコミュニケーションインタフェース 3 (SCI3)

本 LSI は、独立した 3 チャンネルのシリアルコミュニケーションインタフェース 3 (SCI3) を備えています。SCI3 は、調歩同期式とクロック同期式の 2 方式のシリアルデータ通信が可能です。調歩同期方式は Univesal Asynchronous Reciever/Transmitter (UART) や、Asynchronous Communication Interface Adapter (ACIA) などの標準の調歩同期式通信用 LSI とのシリアルデータ通信ができます。また、調歩同期方式では複数のプロセッサ間のシリアルデータ通信機能 (マルチプロセッサ通信機能) を備えています。

SCI3_2 は IrDA (Infrared Data Association) 規格バージョン 1.0 に基づく IrDA 通信波形の送受信が可能です。

SCI3 のチャンネル構成を表 1 に、ブロック図を図 5、図 6 に示します。3 チャンネル (SCI3, SCI3_2, SCI3_3) とも同一機能です。なお、中ではチャンネルによる区別を省略します。

- シリアルデータ通信フォーマットを調歩同期式またはクロック同期式に設定可能
- 全二重通信が可能
独立した送信部と受信部を備えているので、送信と受信を同時に行うことができます。また、送信部および受信部ともにダブルバッファ構造になっていますので連続送受信が可能です。
- 内蔵ボーレートジェネレータで任意のビットレートを選択可能
- 送受信クロックソースとして内蔵ボーレートジェネレータまたは外部クロックを選択可能
- 6 種類の割り込み要因
送信終了、送信データエンpty、受信データフル、オーバランエラー、フレーミングエラー、パリティエラーの割り込み要因があります。また、送信データエンpty、受信データフル割り込み要因により、DTC を起動することができます。
- 内部 16 ビットバスによる高速アクセス
TRDCNT, GR の 16 ビットレジスタに対して、16 ビットバスインタフェースによる高速アクセスが可能

調歩同期式モード

- データ長: 7 ビット/8 ビット選択可能
- ストップビット長: 1 ビット/2 ビット選択可能
- パリティ: 偶数パリティ/奇数パリティ/パリティなしから選択可能
- 受信エラーの検出: パリティエラー、オーバランエラー、フレーミングエラー
- ブレークの検出: フレーミングエラー発生時の RXD 端子のレベルを直接読み出すことでブレークを検出可能

クロック同期式モード

- データ長: 8 ビット
- 受信エラーの検出: オーバランエラー

表 1 SCI3 のチャンネル構成

チャンネル	略称	端子	レジスタ	レジスタアドレス	ノイズ除去回路
チャンネル 1	SCI3* ¹	SCK3 RXD TXD	SMR	H'FF0550	あり
			BRR	H'FF0551	
			SCR3	H'FF0552	
			TDR	H'FF0553	
			SSR	H'FF0554	
			RDR	H'FF0555	
			RSR	—	
			TSR	—	
			SPMR	H'FF0556	
チャンネル 2	SCI3_2* ²	SCK3_2 RXD_2/IrRxD TXD_2/IrTxD	SMR_2	H'FF0558	あり
			BRR_2	H'FF0559	
			SCR3_2	H'FF055A	
			TDR_2	H'FF055B	
			SSR_2	H'FF055C	
			RDR_2	H'FF055D	
			RSR_2	—	
			TSR_2	—	
			SPMR_2	H'FF055E	
			IrCR	H'FF05DE	
チャンネル 3	SCI3_3	SCK3_3 RXD_3 TXD_3	SMR_3	H'FF0560	あり
			BRR_3	H'FF0561	
			SCR3_3	H'FF0562	
			TDR_3	H'FF0563	
			SSR_3	H'FF0564	
			RDR_3	H'FF0565	
			RSR_3	—	
			TSR_3	—	
			SPMR_3	H'FF0566	

- 【注】 1. ブートモードによるオンボードプログラミングモードでは、SCI3 のチャンネル 1 を使用します。
2. SCI3_2 は、IrDA (Infrared Data Association) 規格バージョン 1.0 に基づく IrDA 通信波形の送受信が可能です。

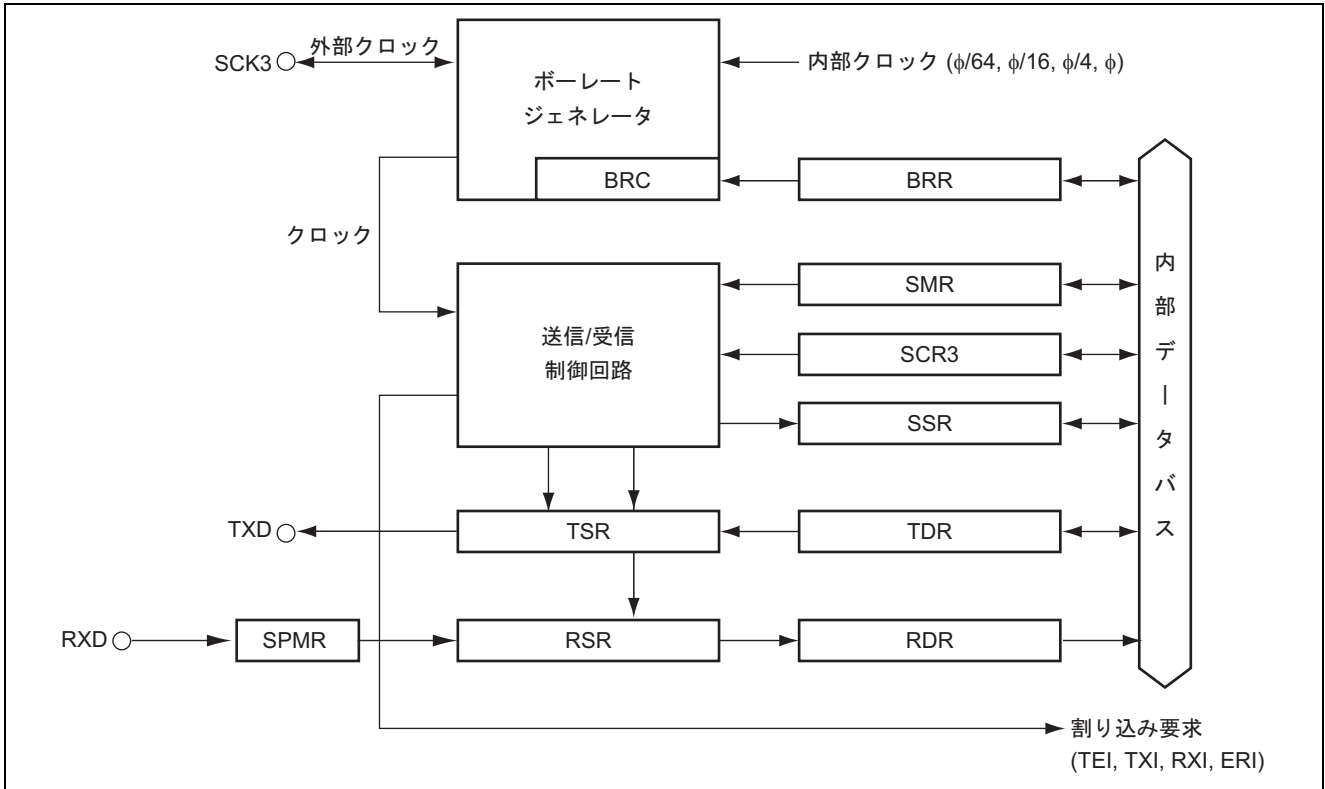


図 5 SCI3, SCI3_3 のブロック図

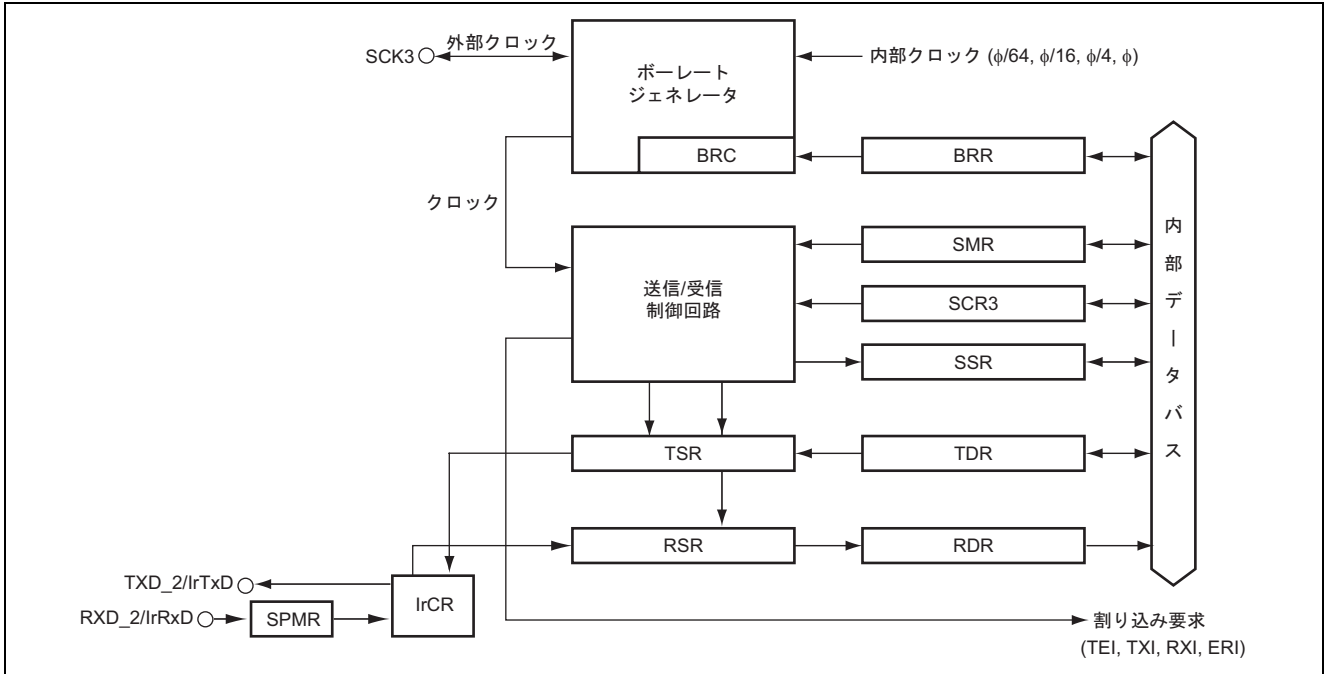


図 6 SCI3_2 のブロック図

2.3 データトランスファコントローラ (DTC)

以下に、DTC の機能を説明します。

本 LSI は、DTC を内蔵しています。DTC は、割り込みまたはソフトウェアによって起動され、データ転送を行うことができます。

図 7 に DTC のブロック図を示します。

- 任意チャネル数の転送可能
- 転送モード: 3 種類
 - (1) ノーマルモード
 - 1 回の転送要求で 1 バイトまたは 1 ワードの転送を実行
 - メモリアドレスを 1 または 2 増減
 - 転送回数は 1 ~ 65,536
 - (2) リピートモード
 - 1 回の転送要求で 1 バイトまたは 1 ワードの転送を実行
 - メモリアドレスを 1 または 2 増減
 - 転送回数は (1 ~ 256) 転送後、初期状態を回復して動作を継続
 - (3) ブロック転送モード
 - 1 回の転送要求で指定したブロックサイズの転送
 - 転送回数は 1 ~ 65,536
 - ソースまたはデスティネーションのいずれかをブロックエリアに指定可能
- 1 つの起動要因で複数データの連続転送が可能(チェイン転送)
- 16M バイトのアドレス空間を直接指定可能
- ソフトウェアによる起動が可能
- 転送単位をバイト/ワードに設定可能
- DTC を起動した割り込みを CPU に要求可能
- モジュールスタンバイモードの設定可能

DTC のレジスタ情報は内蔵 RAM に配置されます。DTC と内蔵 RAM 間は 32 ビットバスで接続されていますので、DTC のレジスタ情報のリード/ライトを 32 ビット 1 ステートで実行できます。

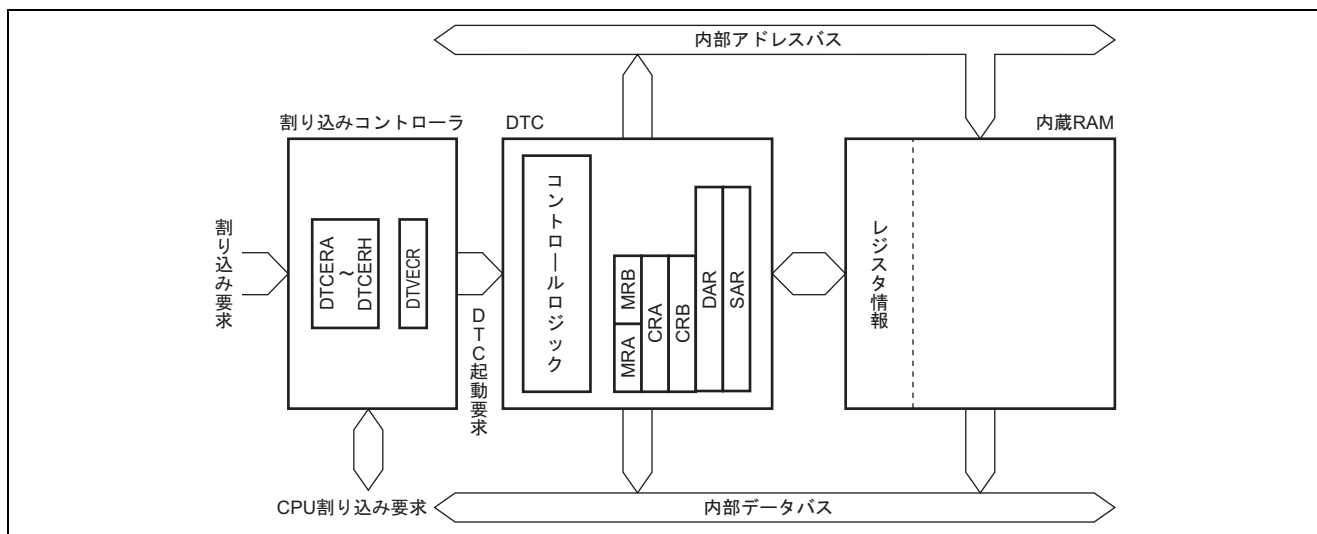


図7 DTCのブロック図

2.3.1 起動要因

DTC は割り込み要求またはソフトウェアによる DTVECR へのライト動作により起動します。起動する割り込み要因は DTCER で選択します。1 回のデータ転送 (チェーン転送の場合 連続した最後の転送) 終了時に、起動要因フラグは SCI3_1 の RDRF フラグになります。

割り込みで DTC を起動する場合は CPU のマスクレベルおよび割り込みコントローラに設定されたプライオリティレベルの影響を受けません。複数の起動要因が同時に発生した場合には、割り込み要因のデフォルトの優先順位に従って DTC が起動します。起動要因と DTCER のクリアを表 2 に、DTC 起動要因制御ブロック図を図 8 に示します。割り込みコントローラの詳細は「H8S/20103, H8S/20203, H8S/20223 ハードウェアマニュアル (RJJ09B0491) 割り込みコントローラ」を参照してください。

表 2 起動要因と DTCER のクリア

起動要因	DISEL ビットが 0 で、指定した回数の転送が終了していないとき	DISEL ビットが 1 のとき、または指定した回数の転送が終了したとき
ソフトウェア起動	SWDTE ビットは 0 にクリア	<ul style="list-style-type: none"> SWDTE ビットは 1 を保持 CPU に割り込みを要求
割り込み起動	<ul style="list-style-type: none"> DTCER の対応するビットは 1 を保持 起動要因フラグは 0 にクリア 	<ul style="list-style-type: none"> DTCER の対応するビットは 0 にクリア 起動要因フラグは 1 を保持 起動要因となった割り込みを CPU に要求

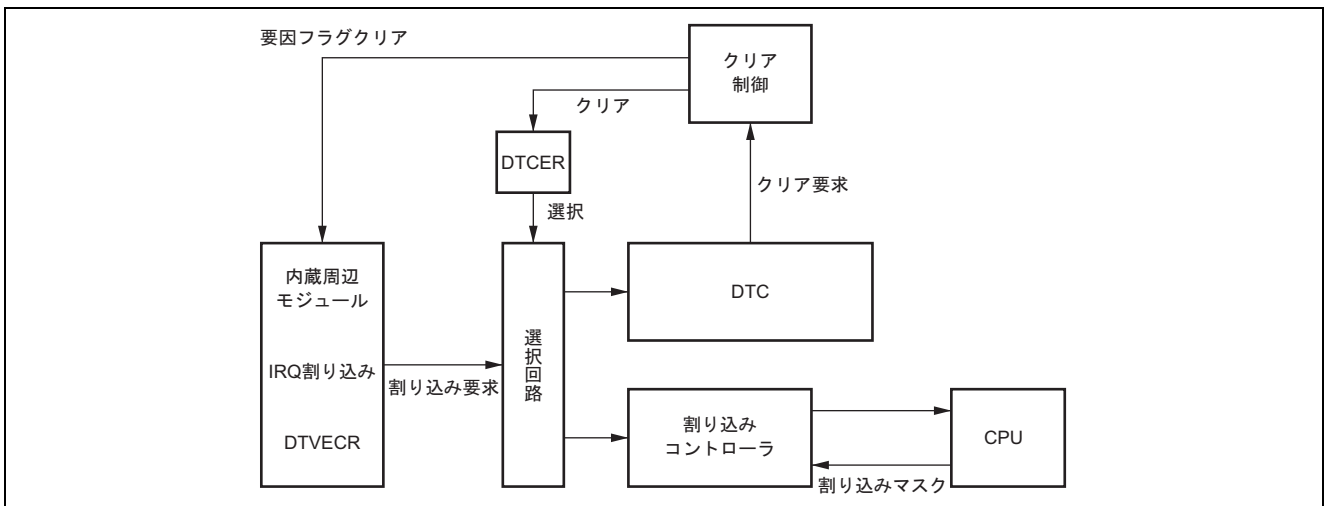


図 8 DTC 起動要因制御ブロック図

2.3.2 レジスタ情報の配置と DTC ベクタテーブル

レジスタ情報は、内蔵 RAM 上に配置してください。アドレスは 4 の倍数の番地としてください。図 9 にアドレス空間上でのレジスタ情報の配置方法を示します。レジスタ情報の先頭アドレスから MRA, SAR, MRB, DAR, CRA, CRB の順に配置してください。また、各レジスタ情報の先頭アドレスを DTC ベクタテーブルの起動要因に対応する番地に格納してください。図 10 に DTC ベクタアドレスとレジスタ情報との対応を示します。DTC は起動要因別にベクタテーブルからレジスタ情報の先頭アドレスをリードし、この先頭アドレスからレジスタ情報をリードします。

表 3 に割り込み要因と DTC ベクタアドレスおよび対応する DTCE を示します。

ソフトウェアで起動する場合のベクタアドレスは $H'400 + (DTVECR[6:0] \times 2)$ となります。たとえば、VOFR = H'0000, DTVECR が H'18 のとき、ベクタアドレスは H'430 となります。

ベクタアドレスの構造は、2 バイト単位になっています。先頭アドレスの下位 2 バイトを設定してください。VOFR を設定することで、ベクタアドレスを可変にすることが可能です。VOFR の詳細については「H8S/20103, H8S/20203, H8S/20223 ハードウェアマニュアル (RJJ09B0491) 割り込みコントローラ」を参照してください。

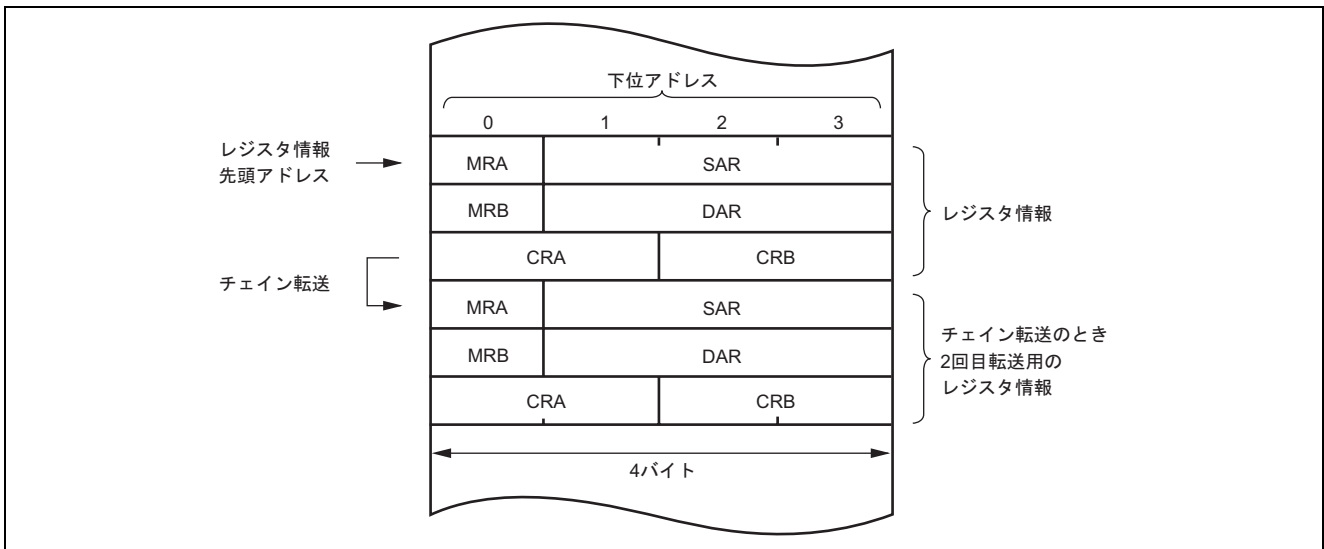


図 9 アドレス空間上での DTC レジスタ情報の配置

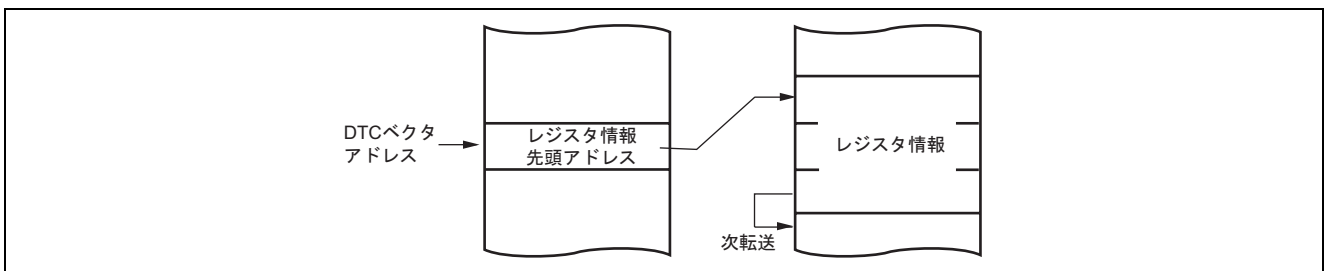


図 10 DTC ベクタアドレスとレジスタ情報との対応

起動要因発生元	起動要因	ベクタ番号	ベクタアドレス* ¹	DTCE* ⁵	優先順位
タイマ RD ユニット 1 チャンネル 3* ⁴	ITDMC1_3 インプットキャプチャ C/ コンペアマッチ C	95	H'4BE ~ H'4BF	DTCEF1	↑ 高 ↓ 低
	ITDMD1_3 インプットキャプチャ D/ コンペアマッチ D	96	H'4C0 ~ H'4C1	DTCEF0	
タイマ RE	ITESC	100	H'4C8 ~ H'4C9	DTCEG4	
	ITEMI	101	H'4CA ~ H'4CB	DTCEG3	
	ITEHR	102	H'4CC ~ H'4CD	DTCEG2	
	ITEDY	103	H'4CE ~ H'4CF	DTCEG1	
	ITEWK	104	H'4D0 ~ H'4D1	DTCEG0	
タイマ RG	ITGMA インプットキャプチャ A/ コンペアマッチ A	109	H'4DA ~ H'4DB	DTCEH3	
	ITGMB インプットキャプチャ B/ コンペアマッチ B	110	H'4DC ~ H'4DD	DTCEH2	

- 【注】
1. ベクタアドレスは VOFR = H'0000 のときの低位 11 ビットを示しています。
 2. H8S/20223 グループのみ搭載しています。他製品では予約となります。
 3. H8S/20103 グループのみ搭載しています。他製品では予約となります。
 4. H8S/20103 グループにはありません。他製品では予約となります。
 5. 対応する割り込みのない DTCE ビットは予約ビットとなります。0 をライトしてください。

3. 動作原理

本タスク例の ELC を使用した SCI3 フレーム送信動作原理を図 11 に示します。図 11 に示すようなハードウェア処理, およびソフトウェア処理によって, ELC を使用した SCI3 フレーム送信をします。TDRE が "1" となり, 送信データエンプティ信号が発生すると, CPU の介在なく DTC が起動し, 次の送信データを TDR にライトします。ELC を使用した SCI3 フレーム送信の DTC 動作を図 12 に示します。

本タスク例の ELC を使用した SCI3 フレーム受信動作原理を図 13 に示します。図 13 に示すようなハードウェア処理, およびソフトウェア処理によって, ELC を使用した SCI3 フレーム受信をします。RDRF が "1" となり, 受信データフル信号が発生すると, CPU の介在なく DTC が起動し, 受信データを RAM に配置した受信データバッファに格納します。ELC を使用した SCI3 フレーム受信の DTC 動作を図 14 に示します。

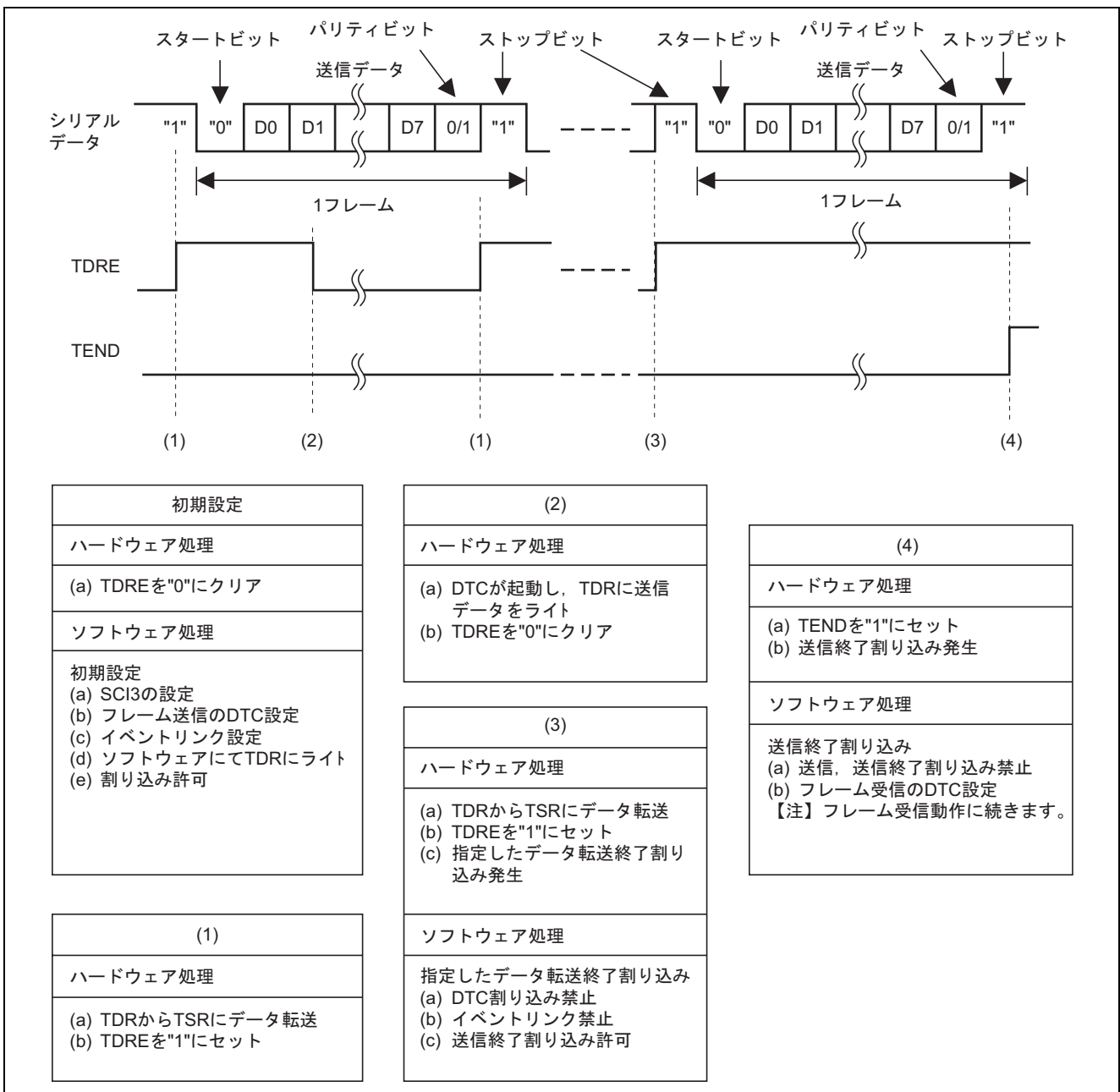


図 11 本タスク例の ELC を使用した SCI3 フレーム送信動作原理

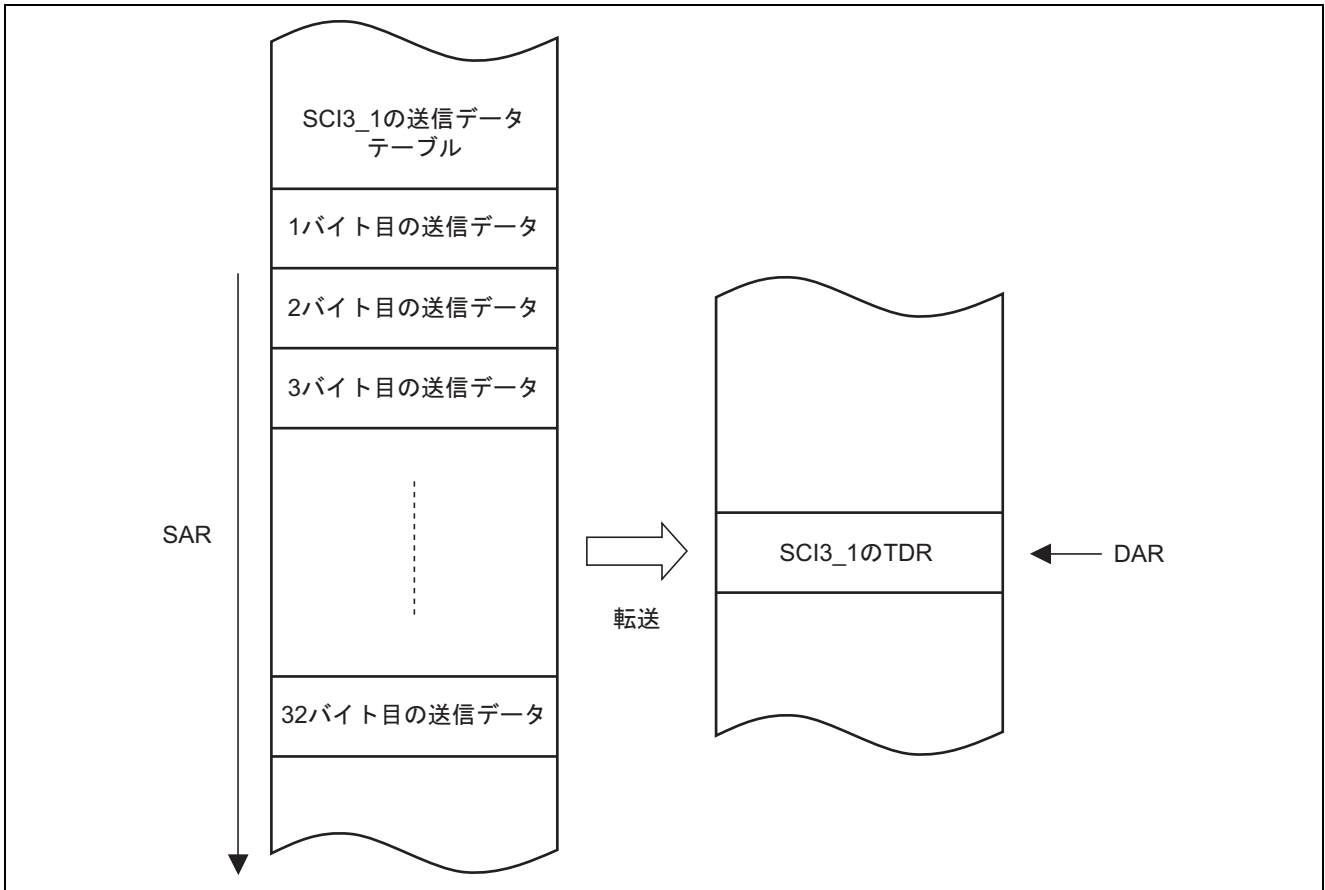


図 12 本タスク例の ELC を使用した SCI3 フレーム送信の DTC 動作

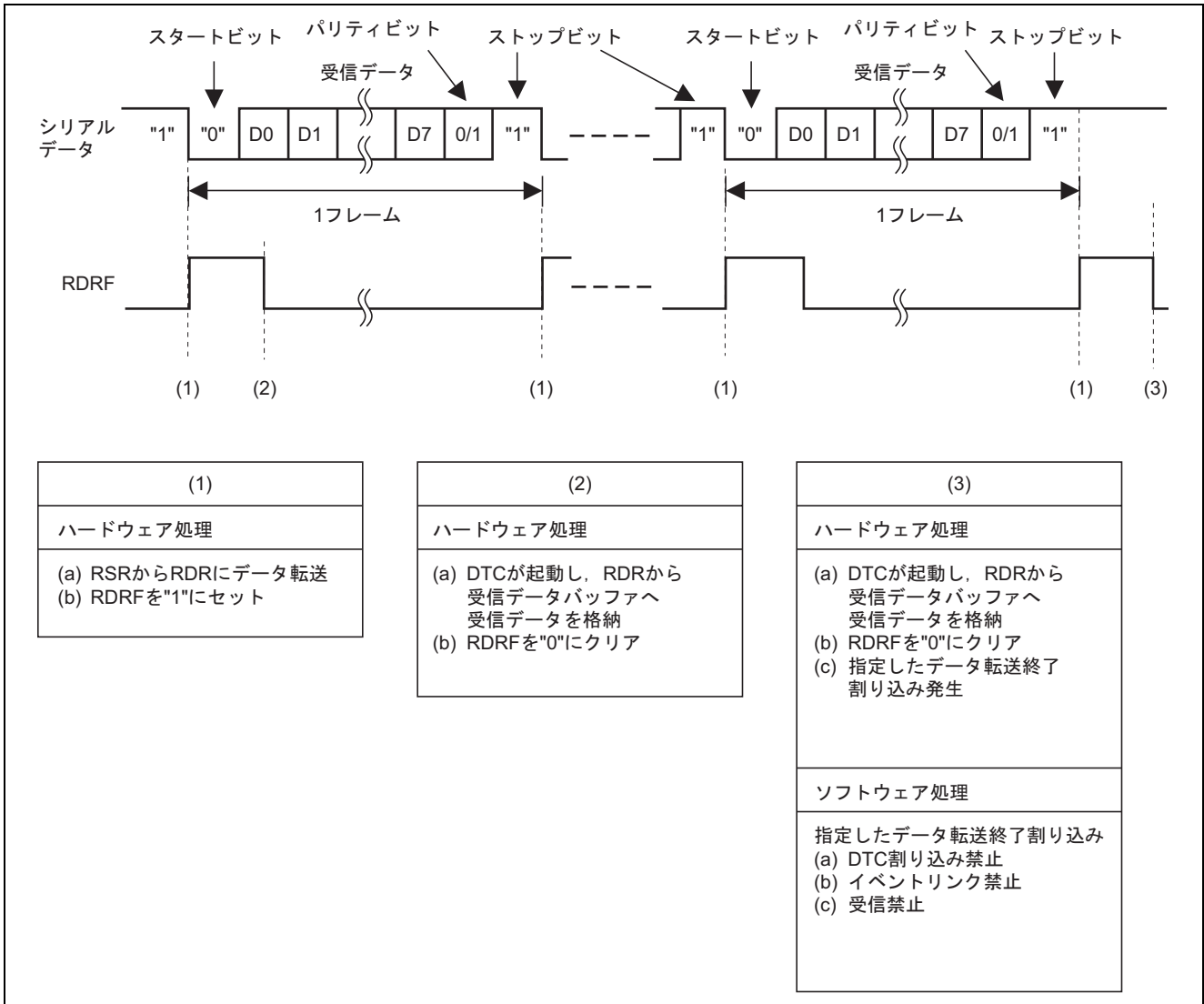


図 13 本タスク例の ELC を使用した SCI3 フレーム受信動作原理

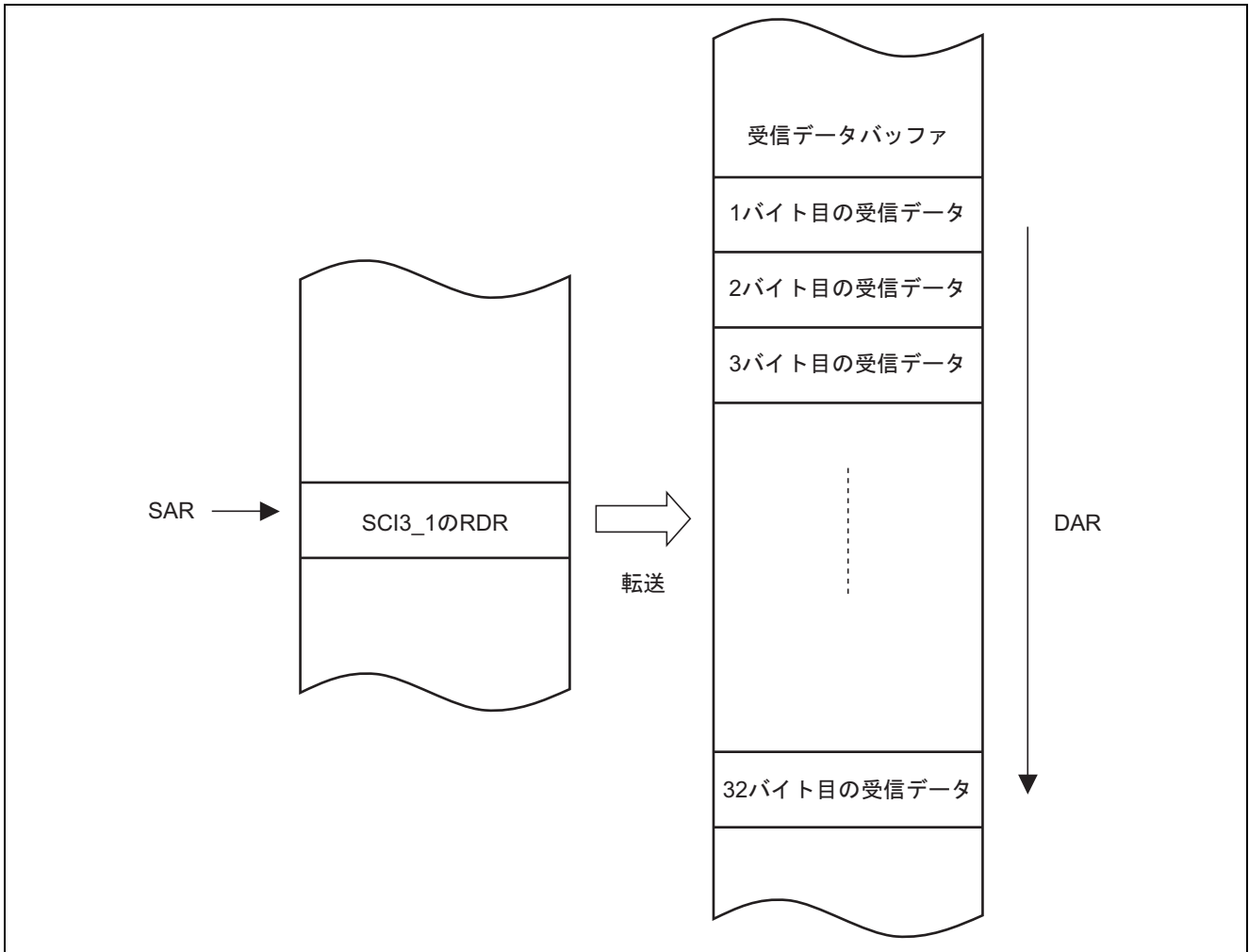


図 14 本タスク例の ELC を使用した SCI3 フレーム受信の DTC 動作

4. ソフトウェア説明

4.1 モジュール説明

表 4 に本タスク例におけるモジュール説明を示します。

表 4 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	各関数の呼び出し, 割り込み許可
システムの初期化ルーチン	h8s_sysinit	モジュールスタンバイの設定, WDT の停止, システムクロックの設定, バスマスタ動作クロックの設定
SCI3_1 設定ルーチン	init_sci31	SCI3 チャンネル 1 の設定
フレーム送信 DTC 設定ルーチン	init_trs_dtc	フレーム送信の DTC 設定, ELC の設定
フレーム受信 DTC 設定ルーチン	init_rcv_dtc	フレーム受信の DTC 設定, ELC の設定
SCI3_1 送信開始ルーチン	trs_st_sci31	SCI3 チャンネル 1 送信開始
ELSR30 イベント割り込みルーチン	INT_ELC2FP_ELC	割り込み要求フラグのクリア, ELSR30 イベント割り込み禁止, イベントリンク禁止, フレーム送信の場合, 送信終了割り込み許可 フレーム受信の場合, 受信禁止
SCI3_1 送信終了割り込みルーチン	INT_TEI_SCI31	送信禁止, 送信終了割り込み禁止, init_rcv_dtc 関数の呼び出し

4.2 引数の説明

本タスク例では, 引数を使用していません。

4.3 使用内部レジスタ説明

表 5 に本タスク例における使用内部レジスタを説明します。

表 5 使用内部レジスタ説明

レジスタ名	シンボル	機能	アドレス	設定値
PMR2	PMR22	P22 端子を TXD 端子に設定	H'FF0001	1
	PMR21	P21 端子を RXD 端子に設定		1
PMRJ	PMRJ[1:0]	PJ0, PJ1 端子を OSC1, OSC2 端子に設定	H'FF000C	B'11
PUCR2	PUCR21	P21 端子のプルアップ MOS をオンに設定	H'FF0011	1
ELCSR	ELIE2	ELF2 割り込みを許可	H'FF0528	1
	ELF2	ELSR30 イベントフラグのクリア		0
DT CERB	ELC2FP	ELSR30 イベント割り込み要因を DTC 起動要因に設定	H'FF0535	1
DTVECR	DTVEC6 ~ DTVEC0	DTC 起動ベクタ番号を設定	H'FF053D	B'0000000

レジスタ名	シンボル	機能	アドレス	設定値
SMR	COM	調歩同期式モードに設定	H'FF0550	0
	CHR	データ長を 8 ビットのフォーマットで送受信		0
	PE	送信時はパリティビットを付加し, 受信時はパリティチェックを行う		1
	PM	偶数パリティで送受信に設定		0
	STOP	1 ストップビットに設定		0
	MP	マルチプロセッサ通信機能を禁止に設定		0
	CKS[1:0]	BRR と合わせて 9600 bps に設定		B'00
BRR		SMR の CKS[1:0] と合わせて 9600 bps に設定	H'FF0551	64
SCR3	TIE	TXI 割り込み要求を禁止	H'FF0552	0
	RIE	RXI および ERI 割り込み要求を禁止		0
	TE	送信動作を許可		1
	RE	受信動作を許可		1
	TEIE	TEI 割り込み要求をイネーブルに設定		1
	CKE[1:0]	クロックソースを内部ポーレートジェネレータに設定		B'00
TDR		送信データをライト	H'FF0553	任意
SSR	TDRE	[1 になる条件] <ul style="list-style-type: none"> SCR3 の TE が "0" のとき TDR から TSR へデータ転送されたとき [0 になる条件] <ul style="list-style-type: none"> CPU が "1" の状態をリードした後, "0" をライトしたとき CPU が TDR へ送信データをライトしたとき DTC で TDR へ送信データを転送, かつ DTC の設定がフラグのクリア条件を満たしたとき*¹ 	H'FF0554	0 or 1
	RDRF	[1 になる条件] <ul style="list-style-type: none"> 受信が正常終了し, RSR から RDR へ受信データが転送されたとき [0 になる条件] <ul style="list-style-type: none"> CPU が "1" の状態をリードした後, "0" をライトしたとき CPU が RDR のデータをリードしたとき DTC で RDR から受信データを転送, かつ DTC の設定がフラグのクリア条件を満たしたとき*¹ 		0 or 1
	TEND	[1 になる条件] <ul style="list-style-type: none"> SCR3 の TE が "0" のとき 送信キャラクタの最後尾ビットの送信時, TDRE が "1" のとき [0 になる条件] <ul style="list-style-type: none"> TDRE = 1 の状態をリードした後, TDRE フラグに "0" をライトしたとき TDR へ送信データをライトしたとき 		0 or 1
RDR		受信データをリード	H'FF0555	不定

レジスタ名	シンボル	機能	アドレス	設定値
ELSR30		DTC とタイマ RD1・チャンネル 0・コンペアマッチ A 信号をリンク設定	H'FF069E	H'4A or H'4C
ELCR	ELCON	全イベントリンク有効に設定	H'FF06BC	1
SYSCCR	PHIHSEL	ϕ high クロックソースを ϕ osc に設定	H'FF06D0	1
LPCR1	PSCSTP	PSC 分周回路動作	H'FF06D1	0
	PHIBSEL	ϕ base クロックソースを ϕ high に設定		1
LPCR2	PHI[2:0]	システムクロック ϕ を ϕ base に設定	H'FF06D2	B'000
LPCR3	PHIS[2:0]	バスマスタ動作クロック ϕ s を ϕ に設定	H'FF06D3	B'000
OSCCSR		ϕ osc 発振安定時間設定	H'FF06D5	H'0E
TMWD		WDT にクロック入力禁止	H'FFFF99	H'F7
TCSRWD		TMWD 書き込み制御	H'FFFF9A	H'A3
MSTCR1	MSTWDT	WDT モジュールスタンバイ解除	H'FFFFDC	0
	MSTDTC	DTC モジュールスタンバイ解除		0
MSTCR2	MSTSCI3_1	SCI3_1 モジュールスタンバイ解除	H'FFFFDD	0
MRA* ²	SM[1:0]	フレーム送信時, 転送後, SAR をインクリメント フレーム受信時, 転送後, SAR 固定	H'FFDF80	B'10 B'00
	DM[1:0]	フレーム送信時, 転送後, DAR 固定 フレーム受信時, 転送後, DAR をインクリメント		B'00 B'10
	MD[1:0]	DTC をノーマルモードに設定		B'00
	Sz	バイトサイズ転送に設定		0
SAR* ²		転送元アドレスを設定します。 フレーム送信時, SCI3_1 の送信データテーブル フレーム受信時, SCI3_1 の RDR	H'FFDF81	H'000A63 H'FF0555
MRB* ²	CHNE	チェイン転送を行わないに設定	H'FFDF84	0
	DISEL	指定されたデータ転送を終了したときだけ, CPU に対して割り込み要求を発生に設定		0
DAR* ²		転送先アドレスを設定します。 フレーム送信時, SCI3_1 の TDR フレーム受信時, SCI3_1 の受信データバッファ	H'FFDF85	H'000553 H'FFDF8C
CRA* ²		転送回数を設定 フレーム送信時, フレームサイズ - 1 フレーム受信時, フレームサイズ	H'FFDF88	31 32
CRB* ²		—	H'FFDF8A	—

【注】 1. DTC は以下の 3 つの条件をすべて満足したときに, 周辺モジュールフラグをクリアします。

- a. DISEL ビットが 0 のとき
 - b. 転送回転カウンタが 0 でないとき (ノーマルおよびリピートモードの場合: CRA, ブロックモードの場合: CRB)
 - c. チェイン転送でないとき
2. DTC レジスタ情報は RAM に配置しています。

4.4 使用 RAM 説明

表 6 に本タスク例における使用 RAM 説明を示します。(DTC レジスタ情報は RAM に配置しています。)

表 6 使用 RAM 説明

ラベル名	機能	メモリ使用量	使用モジュール名
rcv_buf[32]	SCI3_1 受信データバッファ	1 バイト × 32	—

4.5 使用定義説明

表 7 に本タスク例における使用定義説明を示します。

表 7 使用定義説明

ラベル名	説明	定義値
FRAME_SIZE	フレーム送信, フレーム受信サイズの設定	32

4.6 使用定数説明

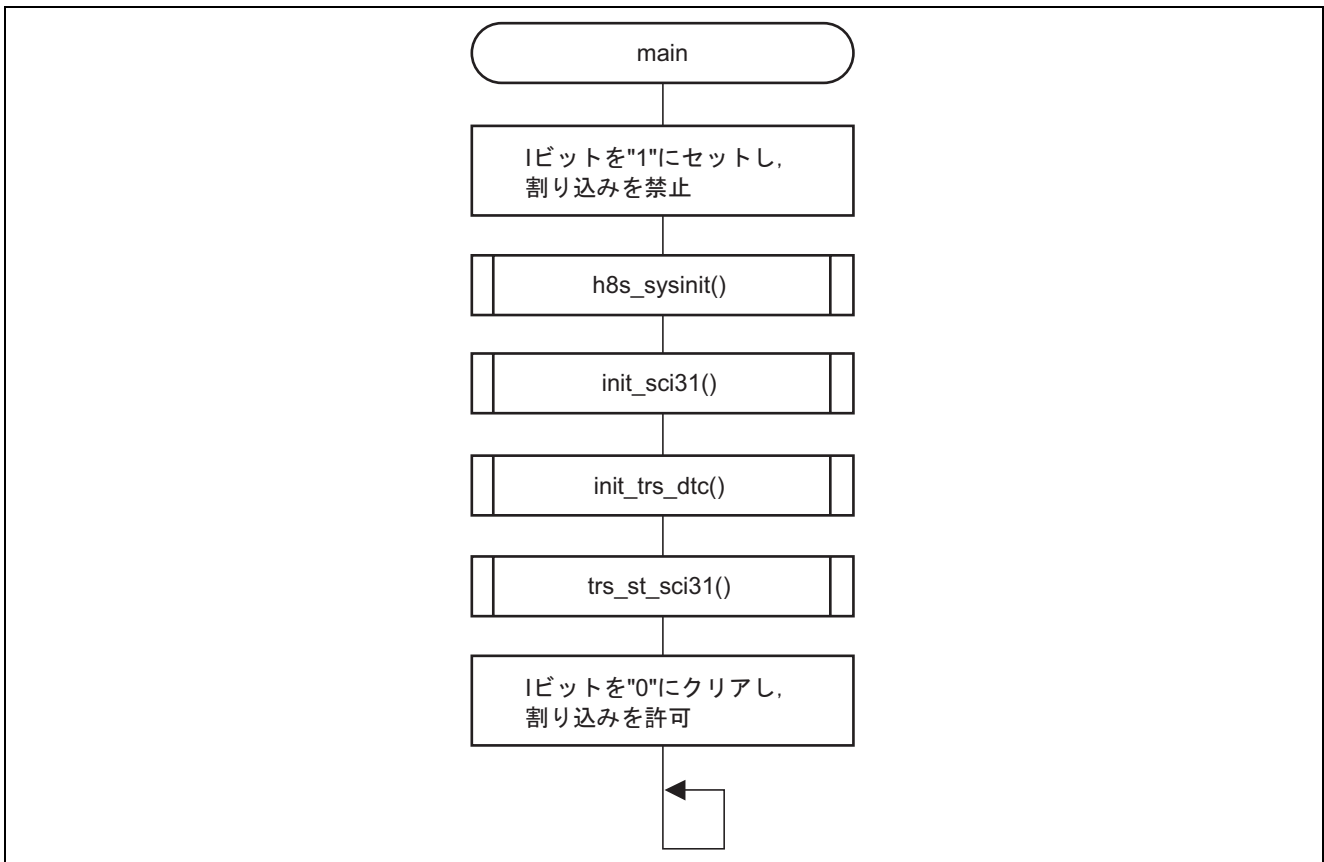
表 8 に本タスク例における使用定数説明を示します。

表 8 使用定数説明

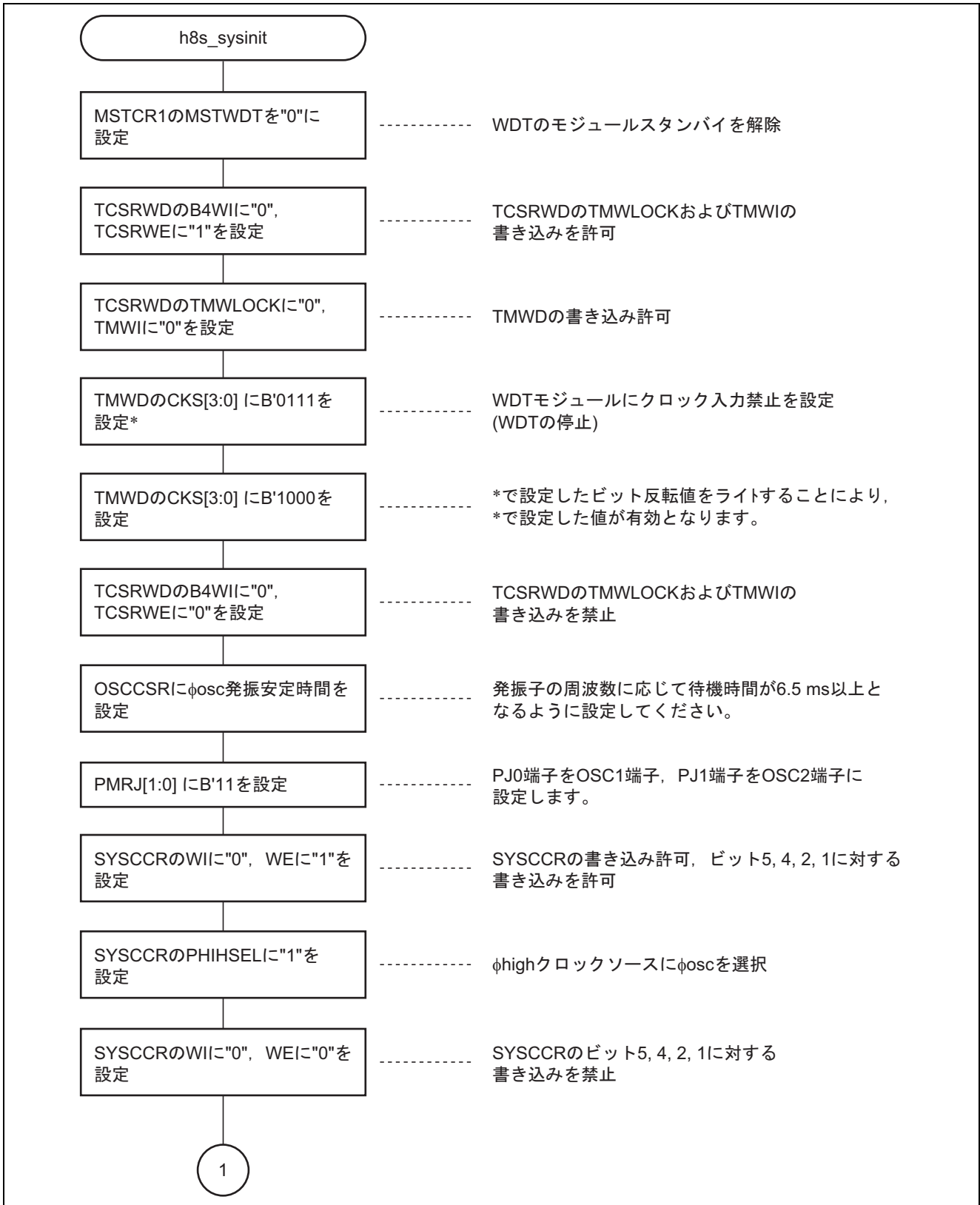
ラベル名	説明	アドレス	定数値
dtcvect_tbl[]	DTC 起動要因に対応する DTC ベクタアドレスに DTC レジスタ情報の先頭アドレスを設定	H'000448	H'DF80
trs_buf[0] ~ trs_buf[31]	SCI3_1 送信データテーブル	H'000A62	H'11, H'22, H'33, H'44, H'55, H'66, H'77, H'88, H'99, H'AA, H'BB, H'CC, H'DD, H'EE, H'FF, H'FE, H'DC, H'BA, H'98, H'76, H'54, H'32, H'10, H'FC, H'DB, H'CA, H'B9, H'A8, H'97, H'86, H'75, H'64

5. フローチャート

5.1 メインルーチン

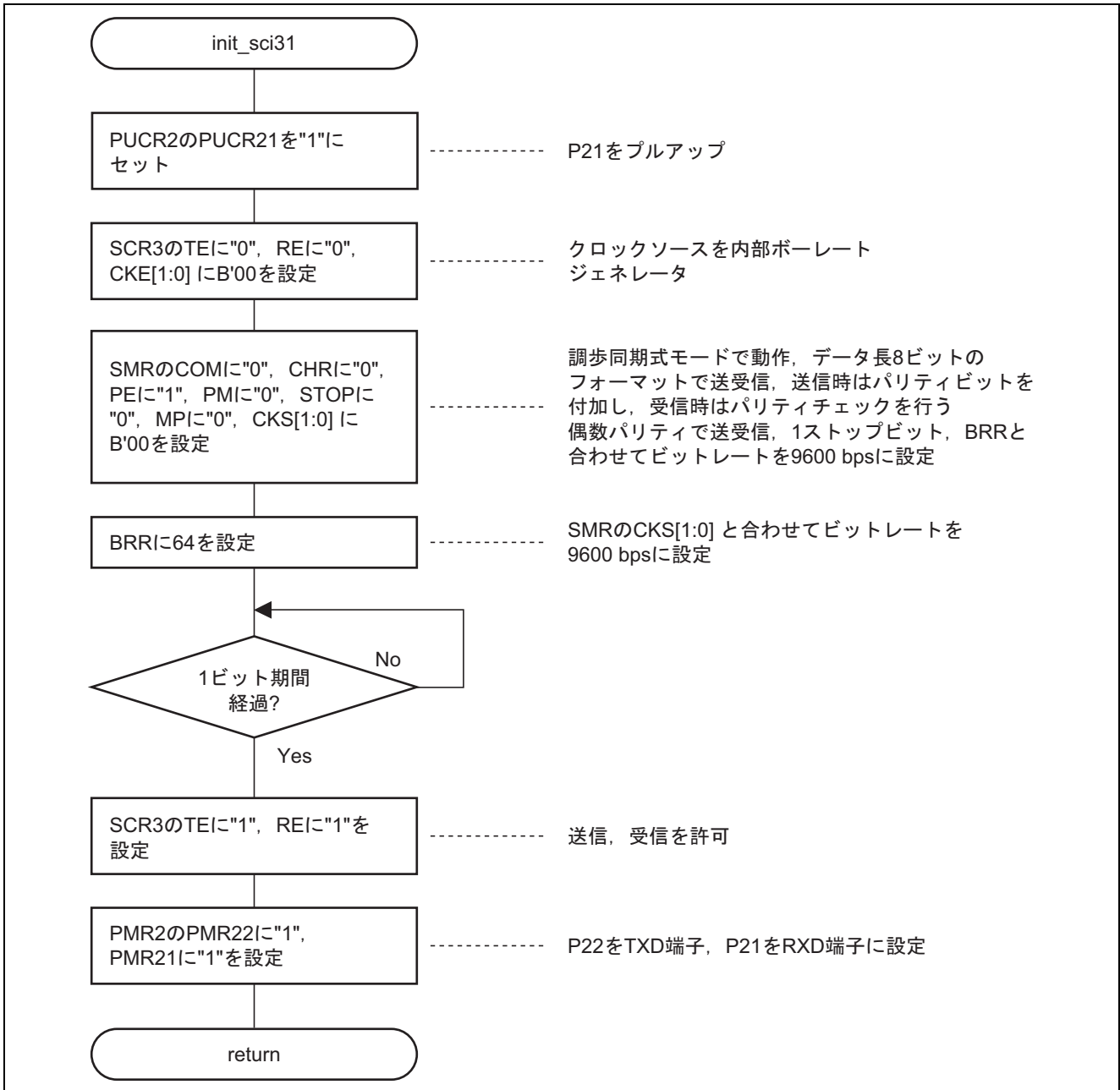


5.2 システムの初期化ルーチン

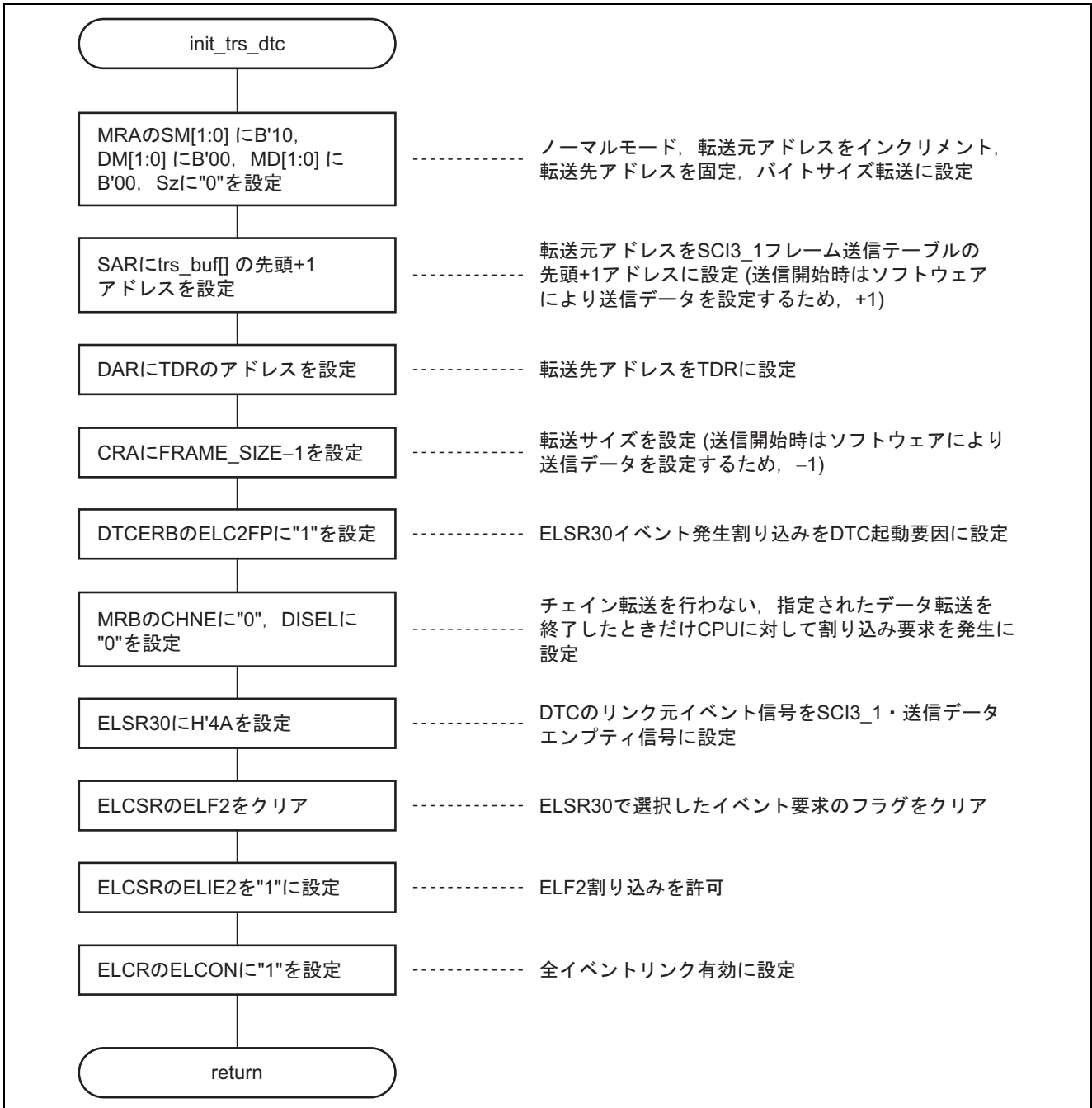




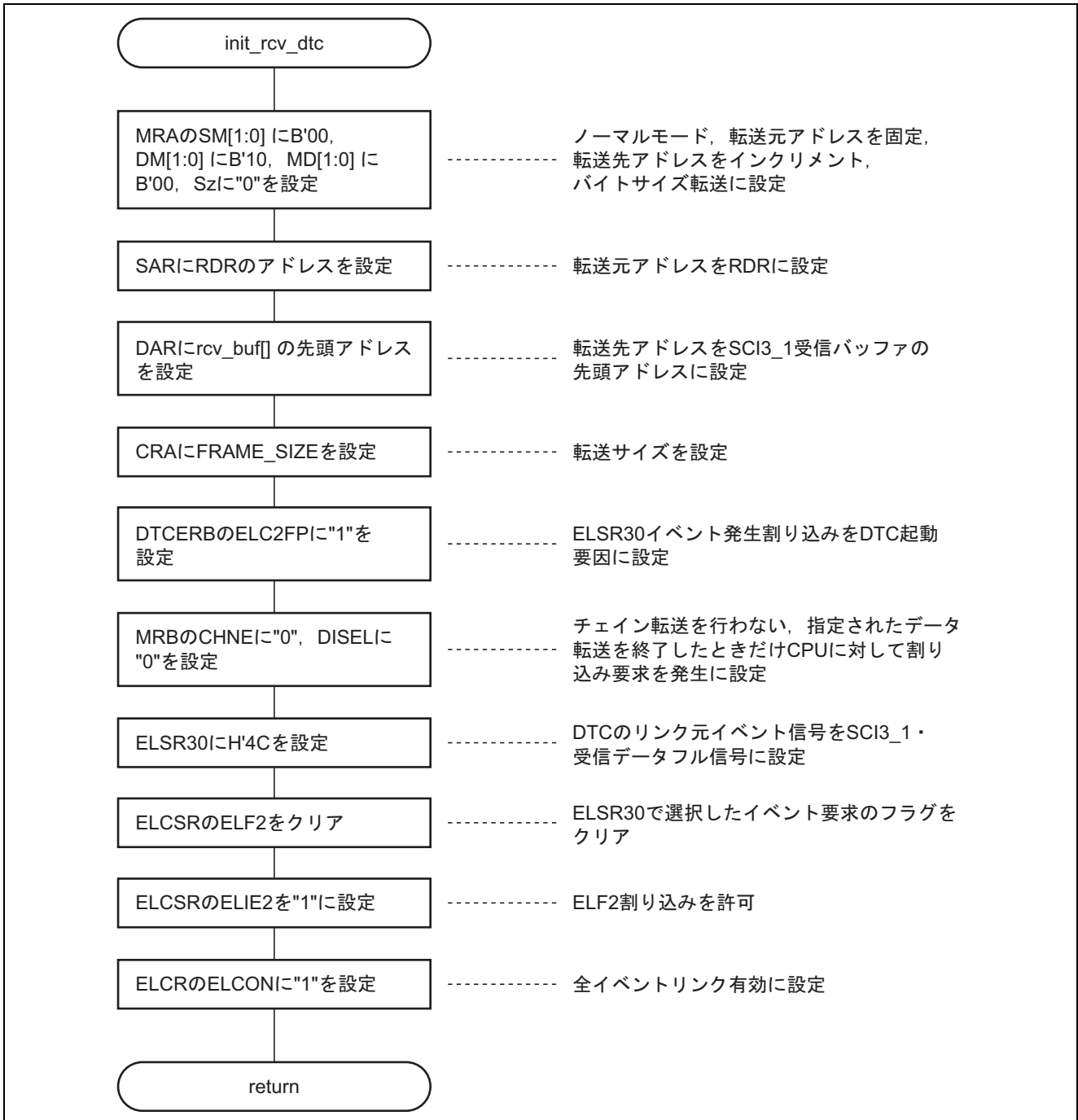
5.3 SCI3_1 設定ルーチン



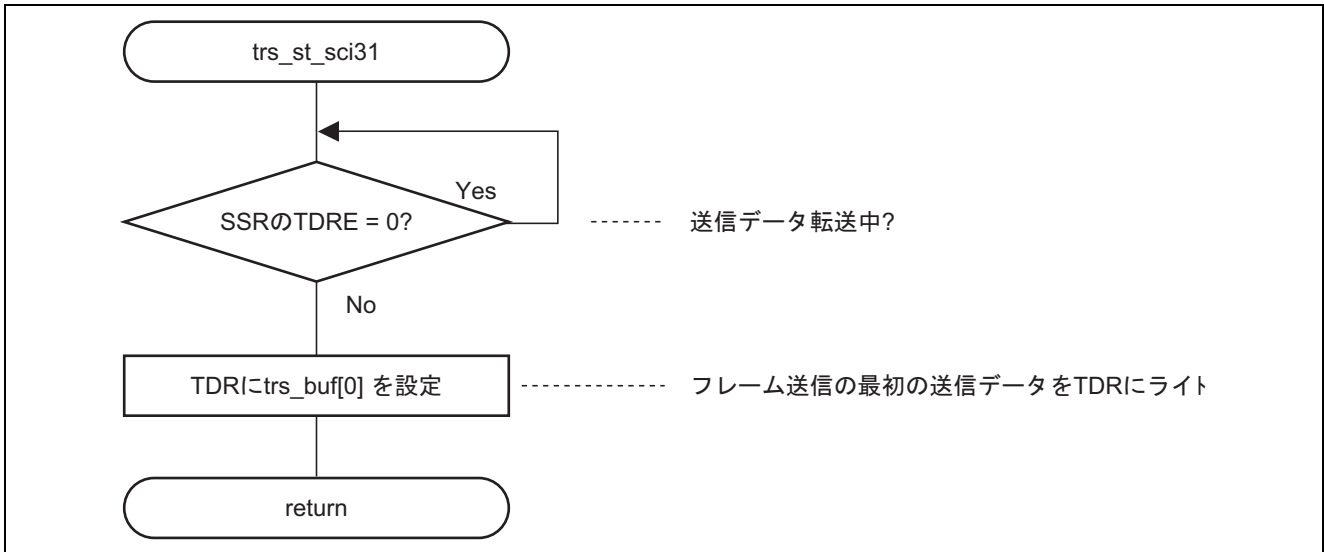
5.4 フレーム送信 DTC 設定ルーチン



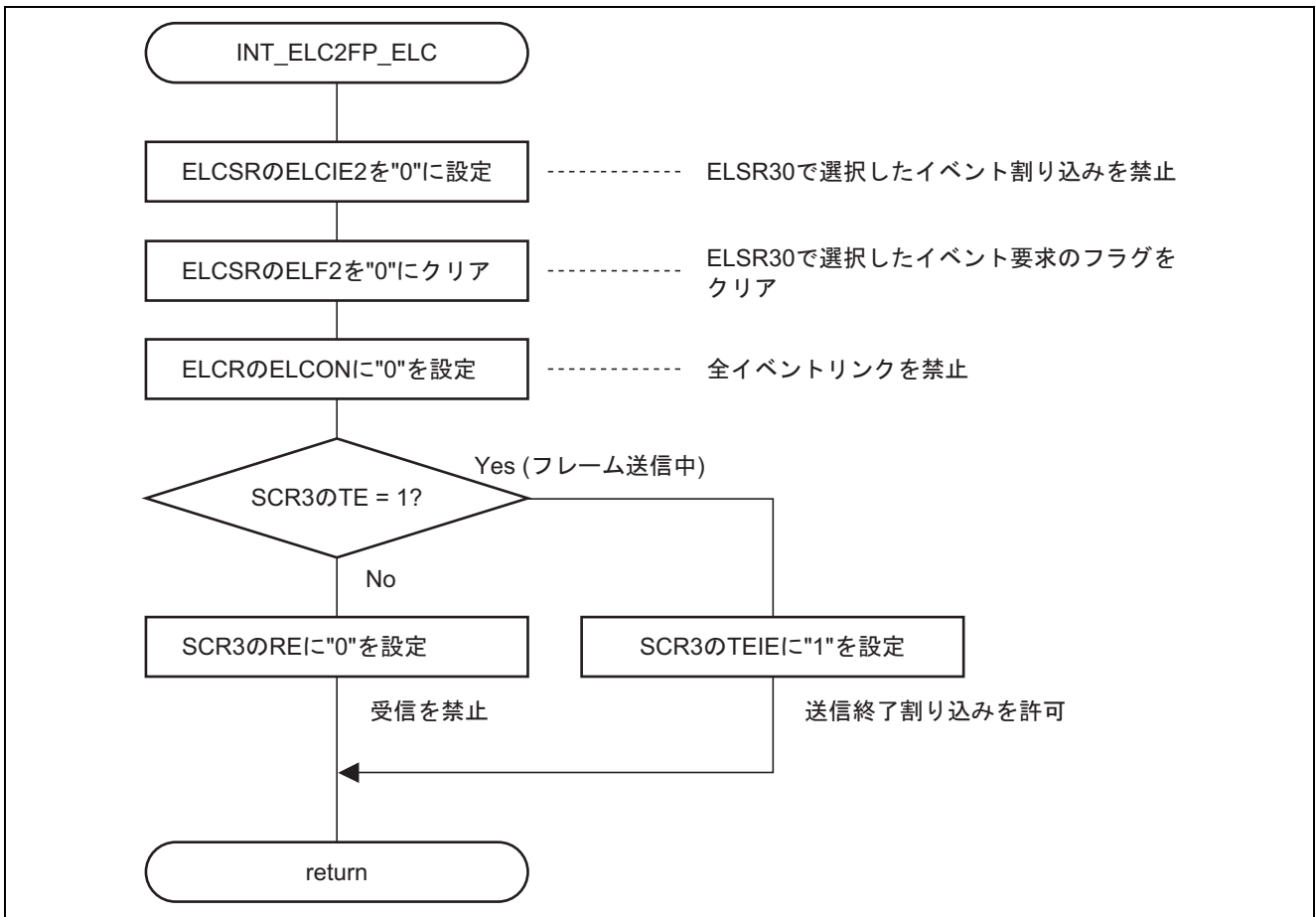
5.5 フレーム受信 DTC 設定ルーチン



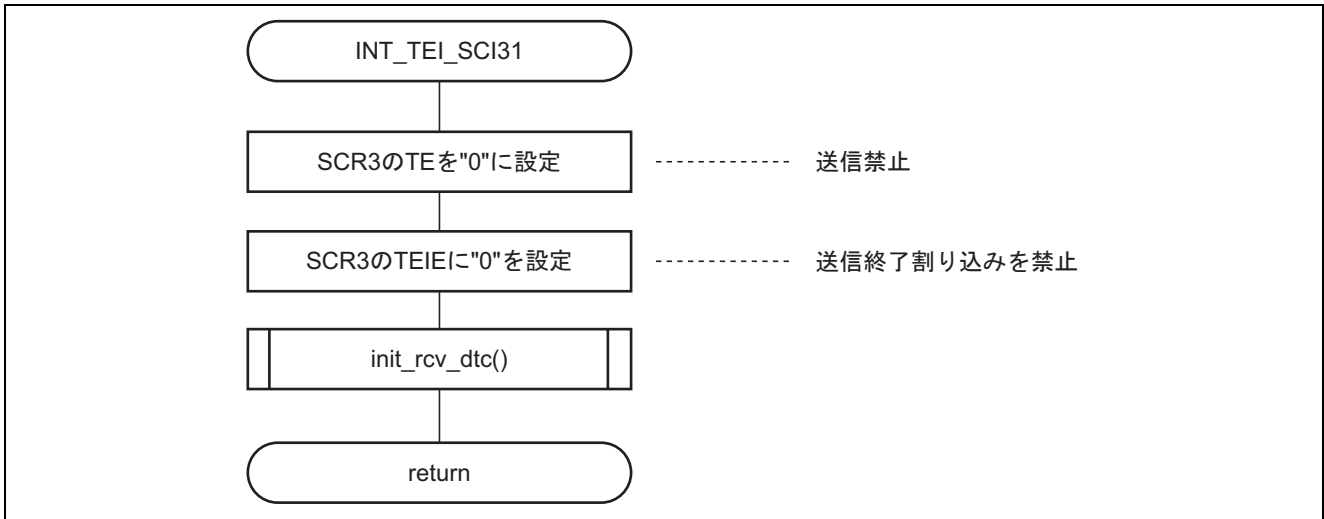
5.6 SCI3_1 送信開始ルーチン



5.7 ELSR30 イベント発生割り込みルーチン



5.8 SCI3_1 送信終了割り込みルーチン



6. プログラムリスト

```

<elc_sci3_frame.c>
/*****
/* H8S/2000 Tiny Series -H8S/20203- */
/* Application Note */
/* */
/* SCI3 frame sending and receiving by ELC */
/* */
/* Function */
/* : start DTC by SCI3_1 transmit data empty, receive data full */
/* Event Link DTC and SCI3_1 */
/* */
/* External Clock : 20MHz */
/* Internal Clock : 20MHz */
*****/
#include <machine.h>
#include "iodefine.h"

typedef struct
{
    union{
        unsigned char MRA; /* DTC mode register A */
        struct{
            unsigned long dummy1:8; /* dummy1 data (MRA address) */
            unsigned long SAR:24; /* DTC source address register */
        }SAR;
    }UN_MRA_SAR;
    union{
        unsigned char MRB; /* DTC mode register B */
        struct{
            unsigned long dummy2:8; /* dummy2 data (DAR address) */
            unsigned long DAR:24; /* DTC destination address register */
        }DAR;
    }UN_MRB_DAR;
    union{
        unsigned short CRA; /* DTC transfer count register A */
        struct{
            unsigned char CRAH; /* DTC transfer count register BH */
            unsigned char CRAL; /* DTC transfer count register BL */
        }CHAR;
    }UN_CRA;
    union{
        unsigned short CRB; /* DTC transfer count register B */
        struct{
            unsigned char CRBH; /* DTC transfer count register BH */
            unsigned char CRBL; /* DTC transfer count register BL */
        }CHAR;
    }UN_CRB;
} st_dtc_reg;

```

```

/*****/
/*Definition of const data          */
/*****/
/* SCI3_1 */
#define FRAME_SIZE          32          /* frame size */

/* transmit data */
const unsigned char trs_buf[FRAME_SIZE]={
    0x11, 0x22, 0x33, 0x44,
    0x55, 0x66, 0x77, 0x88,
    0x99, 0xAA, 0xBB, 0xCC,
    0xDD, 0xEE, 0xFF, 0xFE,
    0xDC, 0xBA, 0x98, 0x76,
    0x54, 0x32, 0x10, 0xFC,
    0xDB, 0xCA, 0xB9, 0xA8,
    0x97, 0x86, 0x75, 0x64
};

/*****/
/*Declaration of function prototype */
/*****/
void main(void);
void init_sci31(void);
void init_trs_dtc(void);
void init_rcv_dtc(void);
void trs_st_sci31(void);
void h8s_sysinit(void);

/*****/
/*Definition of RAM area          */
/*****/
#pragma section DTC
st_dtc_reg DTC_REG;          /* DTC register */

#pragma section
unsigned char rcv_buf[FRAME_SIZE];          /* receive buffer */

/*****/
/*Name:      main          */
/*Parameters: None          */
/*Returns:   None          */
/*Description: User main          */
/*****/
void main(void)
{
    unsigned char ii;

    set_ccr(0x80);          /* set CCR-Ibit */

    h8s_sysinit();          /* initialize system */

    init_sci31();          /* initialize SCI3_1 */

```

```

init_trs_dtc(); /* initialize DTC (SCI3_1 transmit) */

trs_st_sci31(); /* transmit start SCI3_1 */

set_imask_ccr(0); /* interrupt enable */

while(1);
}

/*****/
/*Name:      init_sci31          */
/*Parameters: None              */
/*Returns:    None              */
/*Description: initialize SCI3_1 */
/*****/
void init_sci31(void)
{
    unsigned short wait;

    /* port pull up */
    PUCR2.BYTE |= 0x02; /* pull up P21 */

    SCI3_1.SCR3.BYTE = 0x00; /* clear TE, RE */
    /* internal baud rate generator */
    SCI3_1.SMR.BYTE = 0x20; /* select asynchronous mode */
    /* even parity, 1 stop bit */
    SCI3_1.BRR = 64; /* bitrate => 9600 bit/s */

    /* 1bit wait */
    for( wait=0; wait<420; wait++ );

    SCI3_1.SCR3.BYTE |= 0x30; /* set TE, RE */
    PMR2.BYTE |= 0x06; /* P21=>RXD, P22=>TXD */
}

/*****/
/*Name:      init_trs_dtc        */
/*Parameters: None              */
/*Returns:    None              */
/*Description: initialize DTC (SCI3_1 transmit) */
/*****/
void init_trs_dtc(void)
{
    DTC_REG.UN_MRA_SAR.MRA = 0x80; /* normal mode, SAR increment, DAR hold */
    /* transfer 1byte size */

    DTC_REG.UN_MRA_SAR.SAR.SAR = (unsigned long)&trs_buf[1]; /* Forwarding former address */
    DTC_REG.UN_MRB_DAR.DAR.DAR = (unsigned long)&SCI3_1.TDR; /* Address at forwarding destination */
    DTC_REG.CRA = FRAME_SIZE - 1; /* Set transfer counter */

    DTC.DTCERB.BIT.ELC2FP = 1; /* DTC start by ELSR30 event */
    DTC_REG.UN_MRB_DAR.MRB = 0x00; /* disable chain, interrupt transfer end */
}

```

```

/* Set event link, SCI3_1 transmit data empty */
ELC.ELSR30.BYTE = 0x4A;
INTC.ELCSR.BYTE &= ~0x02;          /* clear ELF2 */
INTC.ELCSR.BIT.ELCIE2 = 1;        /* ELF2 interrupt enable */
ELC.ELCR.BIT.ELCON = 1;          /* event link enable */
}

/*****/
/*Name:      init_rcv_dtc          */
/*Parameters: None                */
/*Returns:   None                 */
/*Description: initialize DTC (SCI3_1 receive) */
/*****/
void init_rcv_dtc(void)
{
    DTC_REG.UN_MRA_SAR.MRA = 0x20;          /* normal mode, SAR hold, DAR increment */
                                          /* transfer 1byte size */

    DTC_REG.UN_MRA_SAR.SAR.SAR = (unsigned long)&SCI3_1.RDR; /* Forwarding former address */
    DTC_REG.UN_MRB_DAR.DAR.DAR = (unsigned long)rcv_buf;    /* Address at forwarding destination */
    DTC_REG.CRA = FRAME_SIZE;              /* Set transfer counter */

    DTC.DTCERB.BIT.ELC2FP = 1;             /* DTC start by ELSR30 event */
    DTC_REG.UN_MRB_DAR.MRB = 0x00;        /* disable chain, interrupt transfer end */

    /* Set event link, SCI3_1 receive data full */
    ELC.ELSR30.BYTE = 0x4C;
    INTC.ELCSR.BYTE &= ~0x02;          /* clear ELF2 */
    INTC.ELCSR.BIT.ELCIE2 = 1;        /* ELF2 interrupt enable */
    ELC.ELCR.BIT.ELCON = 1;          /* event link enable */
}

/*****/
/*Name:      trs_st_sci31         */
/*Parameters: None                */
/*Returns:   None                 */
/*Description: start transmit SCI3_1 */
/*****/
void trs_st_sci31(void)
{
    while ( SCI3_1.SSR.BIT.TDRE == 0 ); /* wait when transmit not empty */

    SCI3_1.TDR = trs_buf[0];
}

```

```

/*****/
/*Name:      h8s_sysinit          */
/*Parameters: None                */
/*Returns:   None                 */
/*Description: initialize H8S/20203 */
/*****/
void h8s_sysinit(void)
{
    MSTCR1.BIT.MSTWDT = 0;          /* WDT module standby off */

    /* stop WDT */
    WDT.TCSRWD.BYTE = 0x97;        /* write enable TMWLOCK, TMWI */
    WDT.TCSRWD.BYTE = 0xA3;        /* write enable TMWD */
    WDT.TMWD.BYTE = 0xF7;          /* Not select clock source */
    WDT.TMWD.BYTE = 0xF8;          /* write bit inversion */
    WDT.TCSRWD.BYTE = 0x87;        /* write disable TMWLOCK, TMWI */

    CPG.OSCCSR.BYTE = 0x0E;        /* wait over 6.5ms, Phi_osc=20MHz */
    PMRJ.BYTE = 0x03;             /* select OSC1,OSC2 */

    CPG.SYSCCR.BYTE = (CPG.SYSCCR.BYTE & 0x7F) | 0x40; /* WI=0, WE=1 */
    CPG.SYSCCR.BYTE = 0x60;        /* high=Phi_osc, Phi_low=Phi_loco */
    CPG.SYSCCR.BYTE = CPG.SYSCCR.BYTE & 0x3F;        /* WI=0, WE=0 */

    CPG.LPCR1.BYTE = (CPG.LPCR1.BYTE & 0x7F) | 0x40; /* WI=0, WE=1 */
    CPG.LPCR1.BYTE = 0x41;        /* PSC on, Phi_base=Phi_high */
    CPG.LPCR1.BYTE = CPG.LPCR1.BYTE & 0x3F;        /* WI=0, WE=0 */

    CPG.LPCR2.BYTE = (CPG.LPCR2.BYTE & 0x7F) | 0x40; /* WI=0, WE=1 */
    CPG.LPCR2.BYTE = 0x40;        /* select system clock */
    CPG.LPCR2.BYTE = CPG.LPCR2.BYTE & 0x3F;        /* WI=0, WE=0 */

    CPG.LPCR3.BYTE = (CPG.LPCR3.BYTE & 0x7F) | 0x40; /* WI=0, WE=1 */
    CPG.LPCR3.BYTE = 0x40;        /* select clock of bus master */
    CPG.LPCR3.BYTE = CPG.LPCR3.BYTE & 0x3F;        /* WI=0, WE=0 */

    /* module standby off */
    MSTCR1.BIT.MSTDTC = 0;        /* DTC module standby off */
    MSTCR2.BIT.MSTSCI3_1 = 0;    /* SCI3_1 module standby off */
}

```

```

<intprg.c>
/*****
/*Extern Declaration of function prototype */
/*****
extern void init_rcv_dtc(void);

// vector 36 ELSR30 event ELC
__interrupt(vect=36) void INT_ELC2FP_ELC(void) {
    INTC.ELCSR.BIT.ELCIE2 = 0;          /* ELF2 interrupt disable */
    DTC.DTCERB.BIT.ELC2FP = 0;        /* clear ELC2FP */
    ELC.ELCR.BIT.ELCON = 0;           /* event link disable */

    /* transmit ? */
    if ( SCI3_1.SCR3.BIT.TE != 0 ){
        SCI3_1.SCR3.BIT.TEIE = 1;     /* trasmit end interrupt enable */
    }
    /* receive */
    else{
        SCI3_1.SCR3.BIT.RE = 0;       /* receive disable */
    }
}

// vector 40 TEI SCI31
__interrupt(vect=40) void INT_TEI_SCI31(void) {
    SCI3_1.SCR3.BIT.TE = 0;           /* trasmit disable */
    SCI3_1.SCR3.BIT.TEIE = 0;        /* transmit end interrupt disable */

    init_rcv_dtc();                  /* initialize DTC (SCI3_1 receive) */
}

```

6.1 リンクアドレス指定

セクション名	アドレス
CDTC_VECT	H'000400
PRresetPRG, PIntPRG	H'000500
P, C, C\$DSEC, C\$BSEC, D	H'000800
BDTC, B, R	H'FFDF80
S	H'FFFD80

ホームページとサポート窓口

ルネサス テクノロジホームページ

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

お問合せ先

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

csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2008.12.24	—	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。

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の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、あるいはその他軍事事業の目的で使用しないでください。また、輸出に際しては、「外国為替および外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりますは、事前に弊社営業窓口で最新の情報をご確認いただきますとともに、弊社ホームページ (<http://www.renesas.com>) などを通じて公開される情報に常にご注意ください。
5. 本資料に記載した情報は、正確を期すため慎重に制作したものです。万一本資料の記述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、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会ください。なお、上記用途に使用されたことにより発生した損害等については弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないでください。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行うもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて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、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に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。弊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、社会的損害など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、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断りいたします。
13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会ください。

D039444