

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、誤りが無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。

# H8/300L SLP シリーズ

## クロック同期式シリアルデータマスタ受信

### 要旨

本アプリケーションノートは、H8/38024 の SCI3 (Serial Communication Interface 3) モジュールを用いてクロック同期式シリアルデータマスタ受信について述べており、ユーザソフトウェア設計の際のご参考として役立てていただくようまとめたものです。

なお、本アプリケーションノートに記載されているプログラムは、動作確認は行なっておりますが、実際にご使用になる場合は、必ず動作確認の上ご使用くださいますようお願いいたします。

### 動作確認デバイス

H8/38024

### 目次

1. 仕様 .....	2
2. 使用機能説明 .....	3
3. 動作説明 .....	6
4. ソフトウェア説明 .....	7
5. フローチャート .....	10
6. プログラムリスト .....	15

### 1. 仕様

図 1 に示すように、H8/38024 の SCI3 を用いて、クロック同期式シリアルデータマスタ受信を行ないます。

本タスク例では、4 バイトのシリアルデータを受信します。

通信フォーマットは、データ長 8 ビット固定の LSB ファーストで、ビットレートは 250[kbps]です。

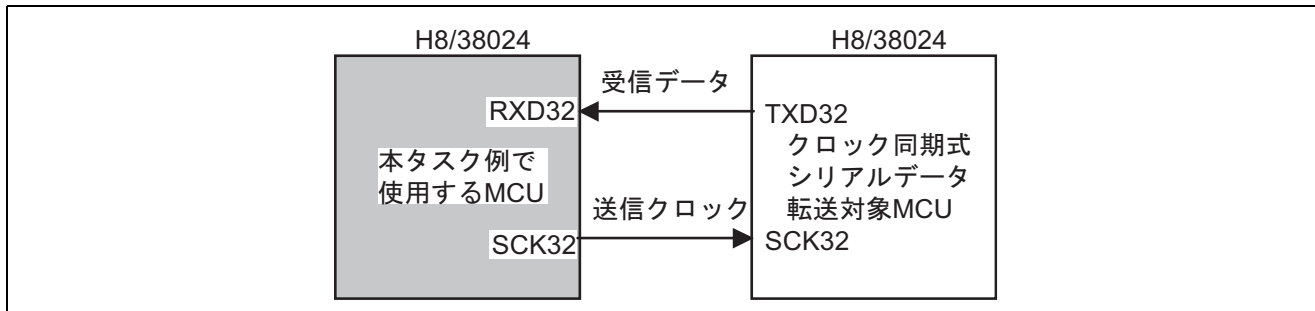


図 1 H8/38024 によるクロック同期式シリアルデータマスタ受信

2. 使用機能説明

本タスク例では、SCI3 (Serial Communication Interface 3) を用いて、クロック同期式シリアルデータマスタ受信を行ないます。図 2 に SCI3 のブロック図を示し、以下に機能説明をします。

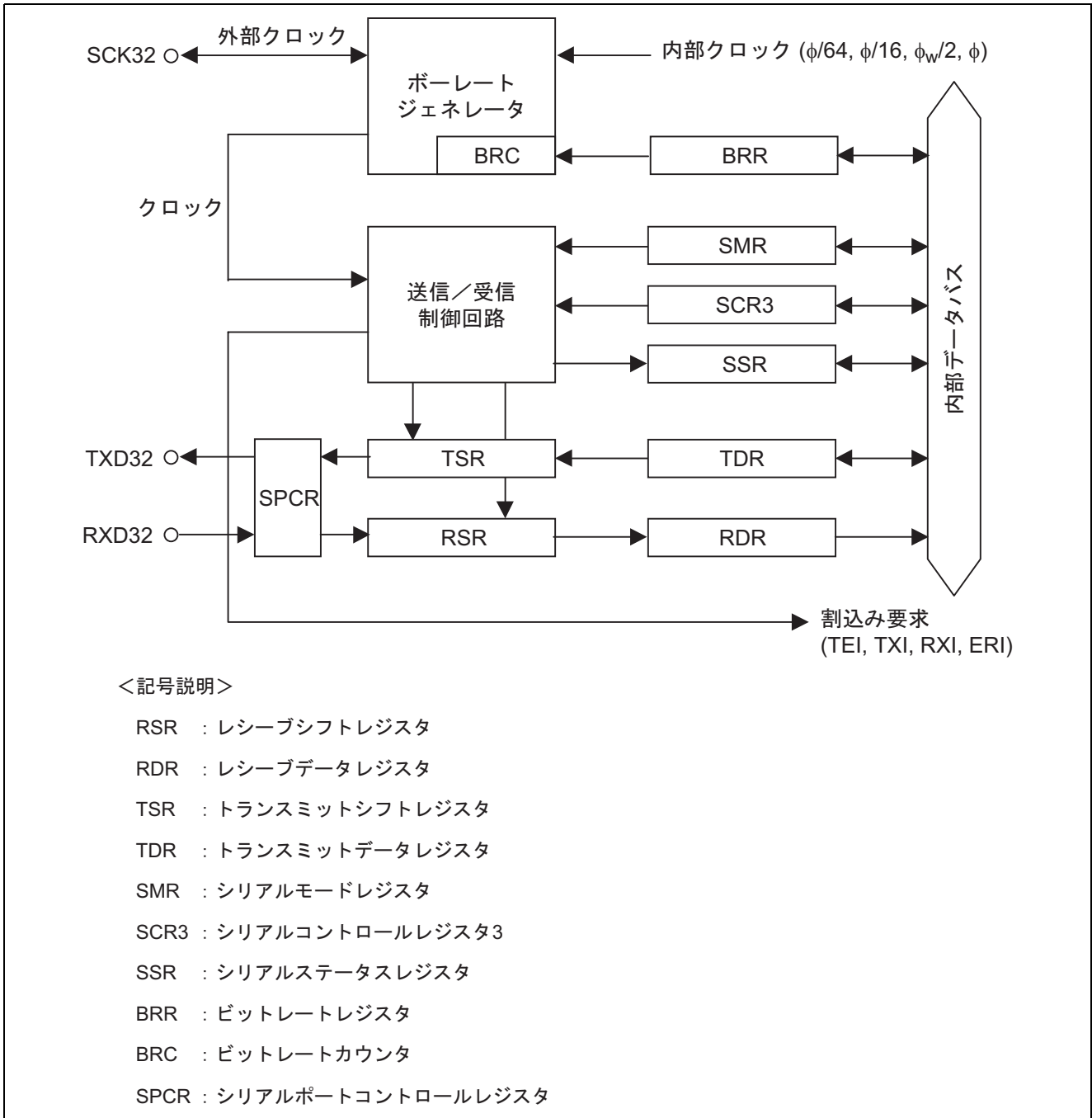


図 2 SCI3 のブロック図

- クロック同期モード  
クロックパルスに同期してデータを送受信します。クロック同期式通信機能を持つ他の LSI とのシリアルデータ通信が可能です。通信データ長は 8 ビット固定です。
- 内部クロックφ  
内蔵周辺機能を動作させるための基準クロックであり、クロック発生回路 (CPG) により生成されます。
- レシーブシフトレジスタ (RSR)  
シリアルデータを受信するためのレジスタです。RSR に RXD32 端子からシリアルデータが入力されたシリアルデータを、LSB (ビット 0) から受信した順にセットし、パラレルデータに変換します。1 フレーム分のデータを受信すると、データは自動的にレシーブデータレジスタ (RDR) へと転送されます。CPU からのアクセスは行なえません。
- レシーブデータレジスタ (RDR)  
受信したシリアルデータを格納する 8 ビットのレジスタです。1 フレーム分のデータを受信すると、自動的に RSR からデータが転送されます。RSR と RDR はダブルバッファ構造ですので、連続受信動作が可能です。RDR は、受信専用レジスタのため、CPU からのリードのみ可能です。
- シリアルモードレジスタ (SMR)  
シリアルデータ通信フォーマットの設定と、内蔵ボーレートジェネレータのクロックソースを選択するための 8 ビットのレジスタです。
- シリアルコントロールレジスタ 3 (SCR3)  
送受信制御と割込み制御、および送受信クロックソースの選択を行なうためのレジスタです。
- シリアルステータスレジスタ (SSR)  
SCI3 のステータスフラグと、送受信マルチプロセッサビットで構成されています。TDRE, RDRF, OER, FER, PER はクリアのみ可能です。
- ビットレートレジスタ (BRR)  
ビットレートを調整するための 8 ビットのレジスタです。SCI3 は、チャンネルごとにボーレートジェネレータが独立しているため、それぞれ異なるビットレートを設定可能です。設定値、実行レートの関係等、詳細はハードウェアマニュアルを参照ください。
- ポートモードレジスタ 2 (PMR2)  
P43/ $\overline{\text{IRQ0}}$  端子機能の切換えを制御します。
- IRQ エッジセレクトレジスタ (IEGR)  
 $\overline{\text{IRQ0}}$  端子の立上がり/立下がりエッジセンスを指定します。
- 割込み許可レジスタ 1 (IENR1)  
 $\overline{\text{IRQ0}}$  割込み要求の許可/禁止を制御します。
- 割込み要求レジスタ 1 (IRR1)  
 $\overline{\text{IRQ0}}$  割込み要求が発生すると、対応するフラグが 1 にセットされます。

表 1 に本タスク例の機能割付けを示します。

表 1 機能割付け

機能	分類	機能割付け
SCK32	端子	SCI3 のクロック入出力端子
TXD32	端子	SCI3 の送信データ出力端子
RXD32	端子	SCI3 の受信データ入力端子
SMR	SCI3	通信フォーマットをクロック同期式モードに設定 内蔵ボーレートジェネレータのクロックソースをφクロックに設定
SCR3	SCI3	受信動作許可
SSR	SCI3	SCI3 の動作状態を示すステータスフラグ
BRR	SCI3	通信のビットレートを設定
TSR	SCI3	シリアルデータを送信するためのレジスタ
TDR	SCI3	送信データを格納するレジスタ
RSR	SCI3	シリアルデータを受信するためのレジスタ
RDR	SCI3	受信データを格納するレジスタ
SPCR	SCI3	RXD32 の入力データを反転しない
PMR2	$\overline{\text{IRQ0}}$	$\overline{\text{IRQ0}}$ 入力端子として機能
IEGR	$\overline{\text{IRQ0}}$	$\overline{\text{IRQ0}}$ 端子入力の立下がりエッジを検出
IENR1	$\overline{\text{IRQ0}}$	$\overline{\text{IRQ0}}$ 端子の割込み要求を許可
IRR1	$\overline{\text{IRQ0}}$	$\overline{\text{IRQ0}}$ 割込み要求フラグ

3. 動作説明

本タスク例におけるクロック同期モード受信時の動作を図3に示します。また、ソフトウェアおよびハードウェア処理の内容を示します。

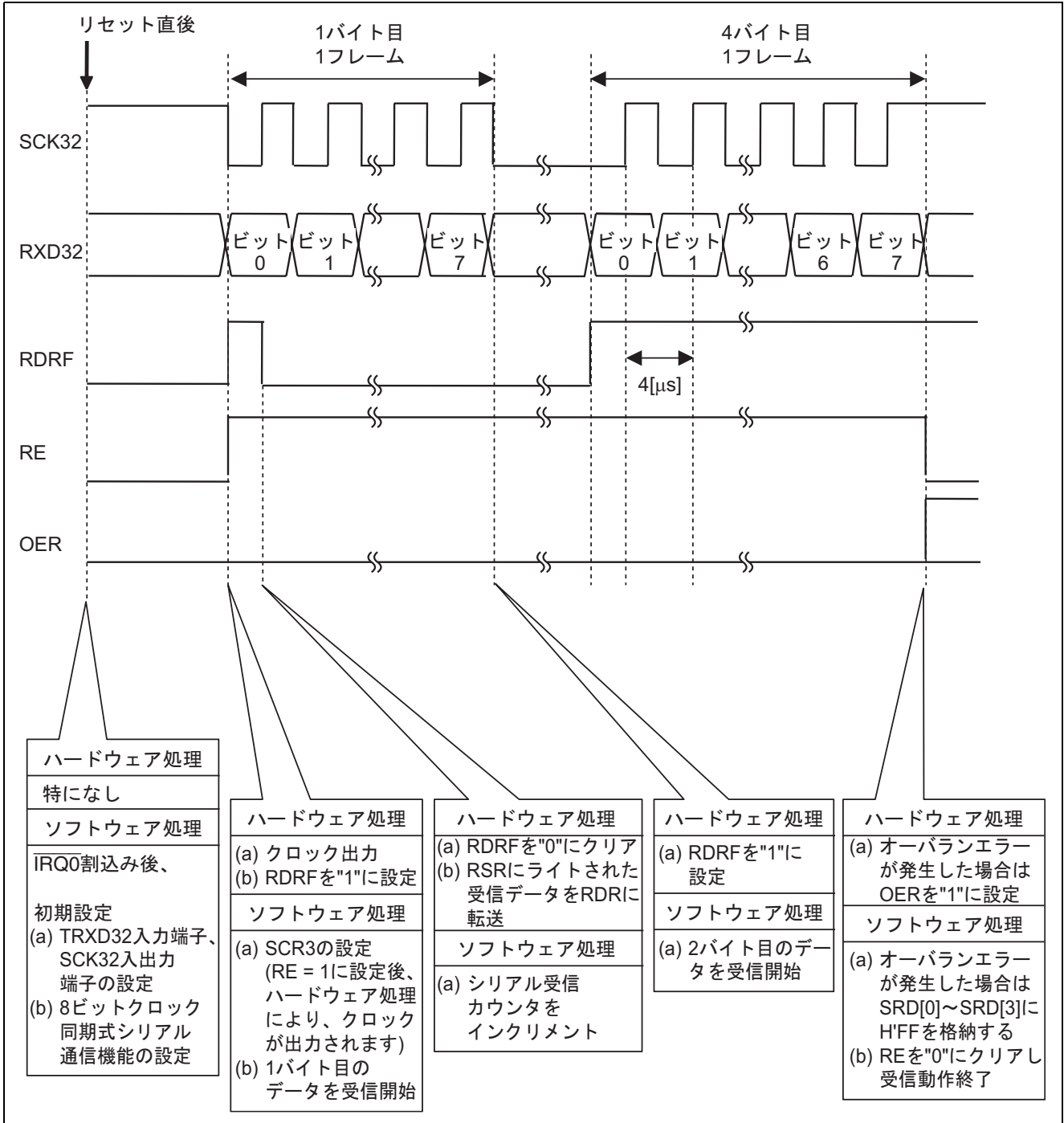


図3 データ受信時の動作



## 4. ソフトウェア説明

### 4.1 モジュール説明

表 2 に本タスク例のモジュールを示します。

表 2 モジュール説明

関数名	機能
main	クロック同期式シリアルデータマスタ受信動作, $\overline{\text{IRQ0}}$ 割込み設定
init_SCI3	SCI3 のイニシャライズ
irq0int	$\overline{\text{IRQ0}}$ 割込み処理

### 4.2 引数の説明

本タスク例では、引数を使用しません。

### 4.3 内部レジスタ説明

以下に、本タスク例で使用する内部レジスタを示します。なお、設定値は本タスク例において使用している値であり、初期値とは異なります。

- SPCR シリアルポートコントロールレジスタ アドレス：H'FF91

ビット	ビット名	設定値	R/W	機能
2	SCINV2	0	R/W	RXD32 端子入力データ反転切換え RXD32 の入力データを反転しない

- SMR シリアルモードレジスタ アドレス：H'FFA8

ビット	ビット名	設定値	R/W	機能
7	COM	1	R/W	コミュニケーションモード SCI3 をクロック同期式モードで動作
2	MP	0	R/W	マルチプロセッサモード マルチプロセッサ通信機能を禁止
1	CKS1	0	R/W	クロックセレクト 1, 0
0	CKS0	0	R/W	内蔵ポーレートジェネレータのクロックソースを $\phi$ クロックに選択

- BRR ビットレートレジスタ アドレス：H'FFA9

機能：CKS1, CKS0 で選択されるポーレートジェネレータの動作クロックと合わせて、送信/受信のビットレートを設定する 8 ビットのレジスタ

設定値：H'04

R/W：R/W

- SCR3 シリアルコントロールレジスタ 3 アドレス：H'FFAA

ビット	ビット名	設定値	R/W	機能
4	RE	1	R/W	レシーブイネーブル 受信動作を許可 (RXD32 端子はレシーブデータ端子)
1	CKE1	0	R/W	クロックイネーブル 1, 0
0	CKE0	0	R/W	クロックソースを内部クロックに選択, および SCK32 端子機能を同期クロック出力選択

## • SSR シリアルステータスレジスタ アドレス：H'FFAC

ビット	ビット名	設定値	R/W	機能
6	RDRF	—	R/(W)*	レシーブデータレジスタフル RDRF=0：RDR にデータが格納されていない RDRF=1：RDR にデータが格納されている
5	OER	0	R/(W)*	オーバランエラー OER=0：受信中，または受信を完了した OER=1：受信中にオーバランエラーが発生した
4	FER	0	R/(W)*	フレーミングエラー FER=0：受信中，または受信を完了した FER=1：受信時にフレーミングエラーが発生した
3	PER	0	R/(W)*	パリティエラー PER=0：受信中，または受信を完了した PER=1：受信時にパリティエラーが発生した

【注】 —: Don't care

\*: フラグをクリアするための 0 ライトのみ可能です。

## • RDR レシーブデータレジスタ アドレス：H'FFAD

機能：受信データを格納する 8 ビットのレジスタです。

設定値：—

R/W：R

## • PMR2 ポートモードレジスタ 2 アドレス：H'FFC9

ビット	ビット名	設定値	R/W	機能
0	$\overline{\text{IRQ0}}$	1	R/W	P43/ $\overline{\text{IRQ0}}$ 端子機能切換え $\overline{\text{IRQ0}} = 0$ ：P43 端子として機能 $\overline{\text{IRQ0}} = 1$ ： $\overline{\text{IRQ0}}$ 入力端子として機能

## • IEGR IRQ エッジセレクトレジスタ アドレス：H'FFF2

ビット	ビット名	設定値	R/W	機能
0	IEG0	0	R/W	$\overline{\text{IRQ0}}$ エッジセレクト IEG0 = 0： $\overline{\text{IRQ0}}$ 端子入力の立下がりエッジを検出 IEG0 = 1： $\overline{\text{IRQ0}}$ 端子入力の立上がりエッジを検出

## • IENR1 割込み許可レジスタ 1 アドレス：H'FFF3

ビット	ビット名	設定値	R/W	機能
0	IEN0	1	R/W	$\overline{\text{IRQ0}}$ 割込みイネーブル IEN0 = 0： $\overline{\text{IRQ0}}$ 端子の割込み要求を禁止 IEN0 = 1： $\overline{\text{IRQ0}}$ 端子の割込み要求を許可

## • IRR1 割込み要求レジスタ 1 アドレス：H'FFF6

ビット	ビット名	設定値	R/W	機能
0	IRRI0	0	R/(W)*	$\overline{\text{IRQ0}}$ 割込み要求フラグ [クリア条件] IRRI0 = 1 の状態で IRRI0 を 0 にライトした時 [セット条件] $\overline{\text{IRQ0}}$ 端子が割込み入力に設定されており，かつ $\overline{\text{IRQ0}}$ 端子に指定されたエッジが入力された時

【注】 \*: フラグをクリアするための 0 ライトのみ可能です。

#### 4.4 使用 RAM 説明

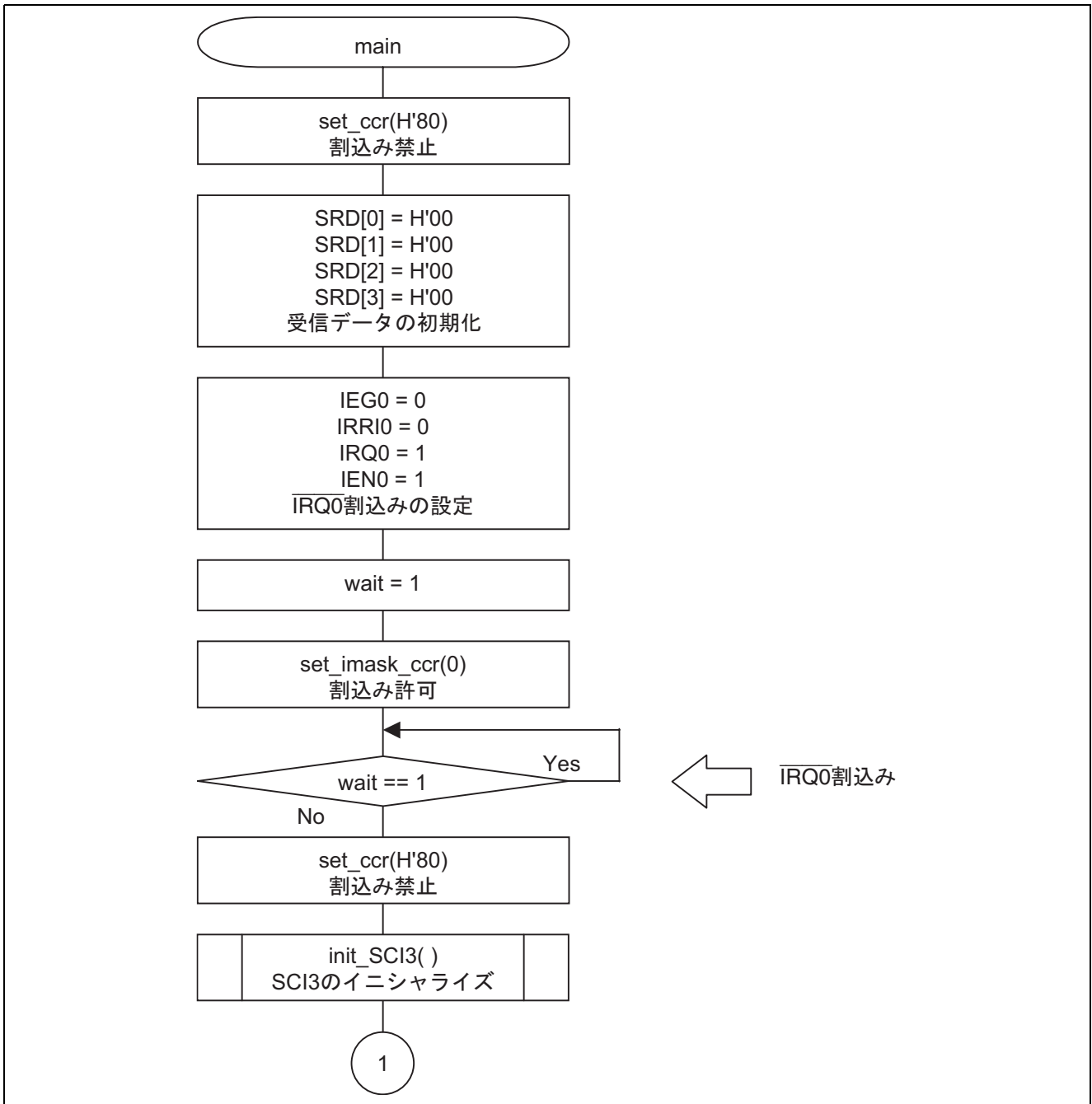
本タスク例で使用する RAM 説明を表 3 に示します。

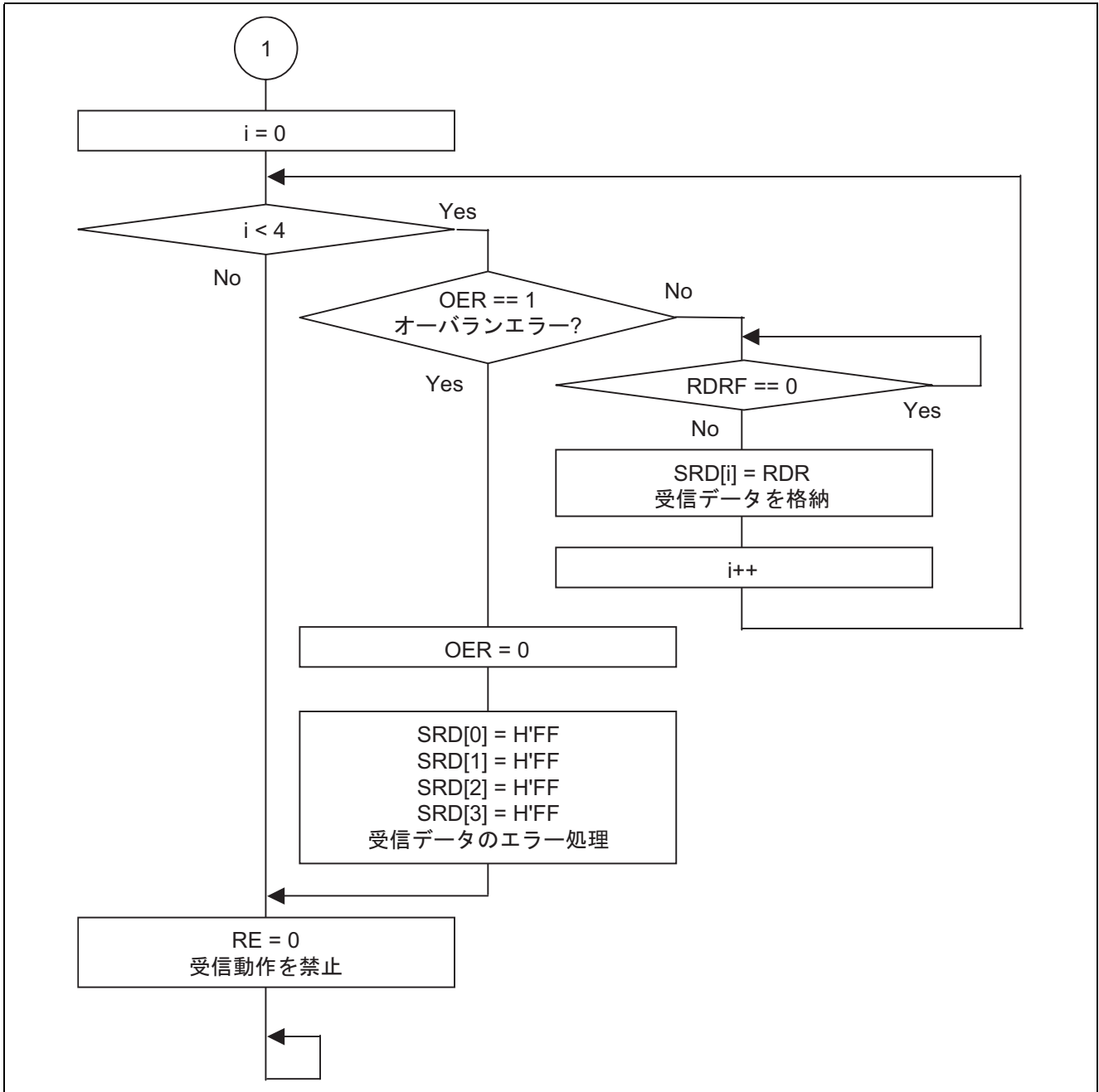
表 3 使用する RAM

定数名	説明	アドレス	メモリ消費量	使用関数名
SRD[4]	クロック同期式シリアル受信データ格納バッファ	H'FB80	4 バイト	main
wait	IRQ0 割込み待ち	H'FB84	1 バイト	main, irq0int

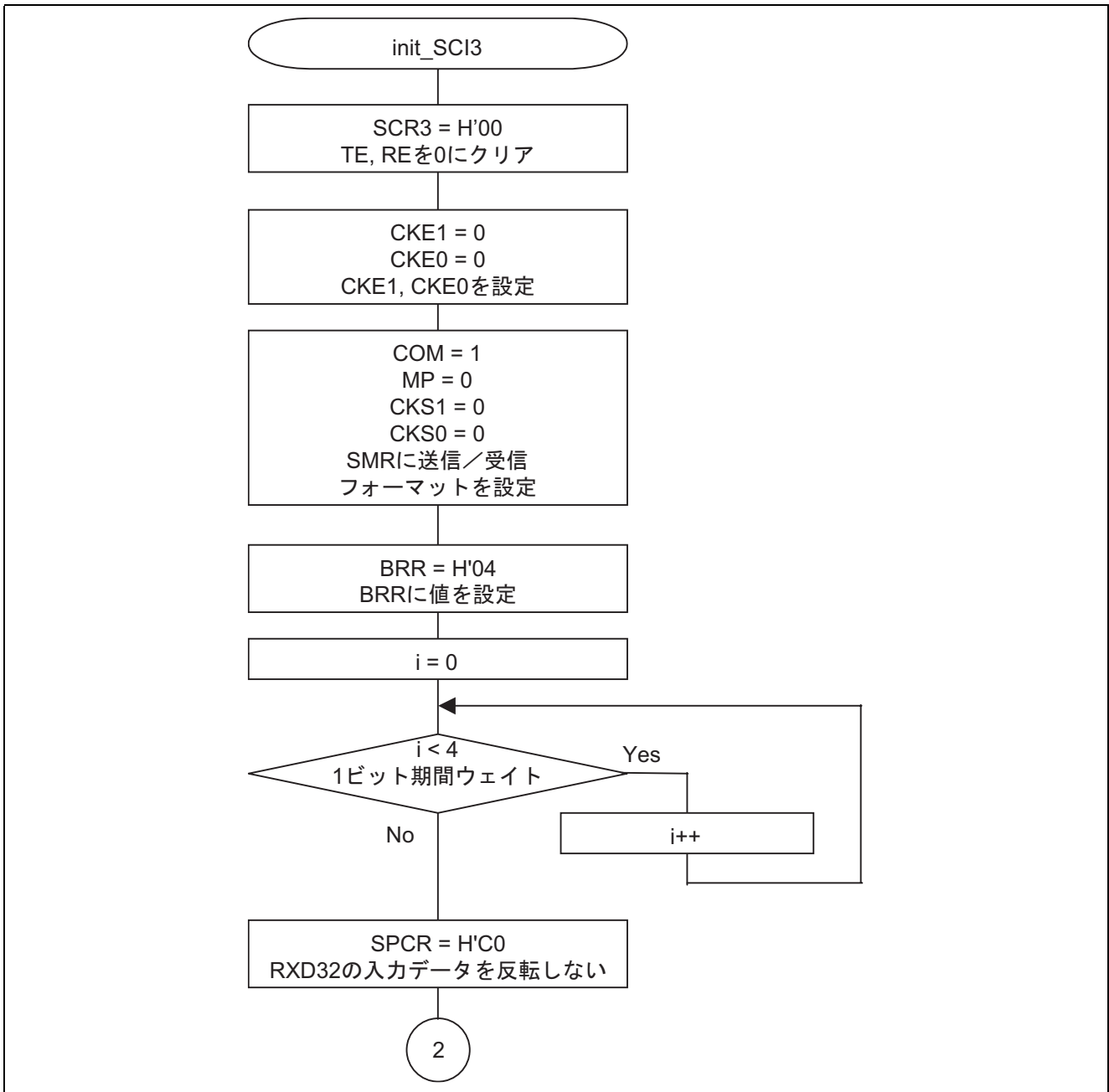
5. フローチャート

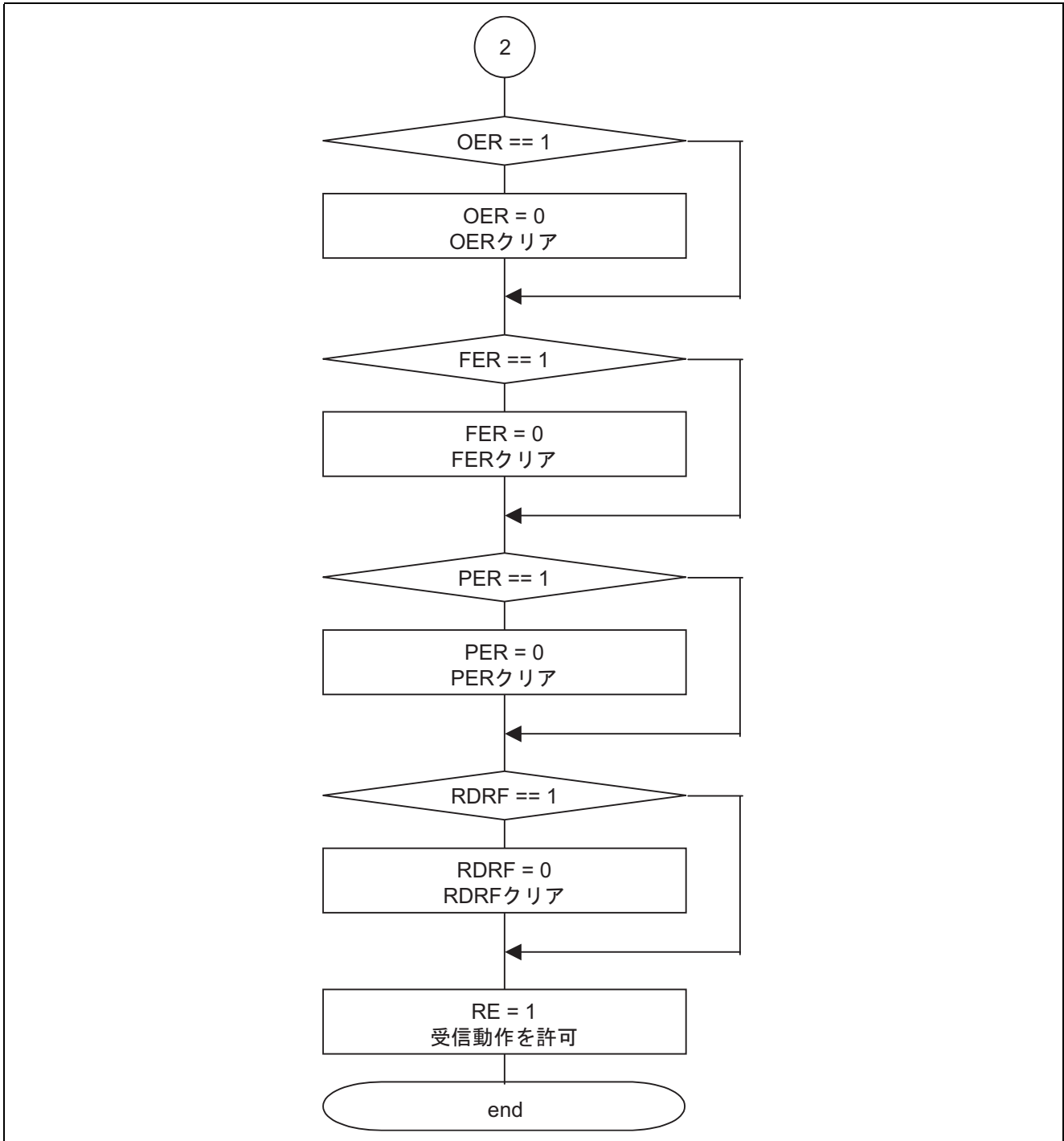
5.1 main



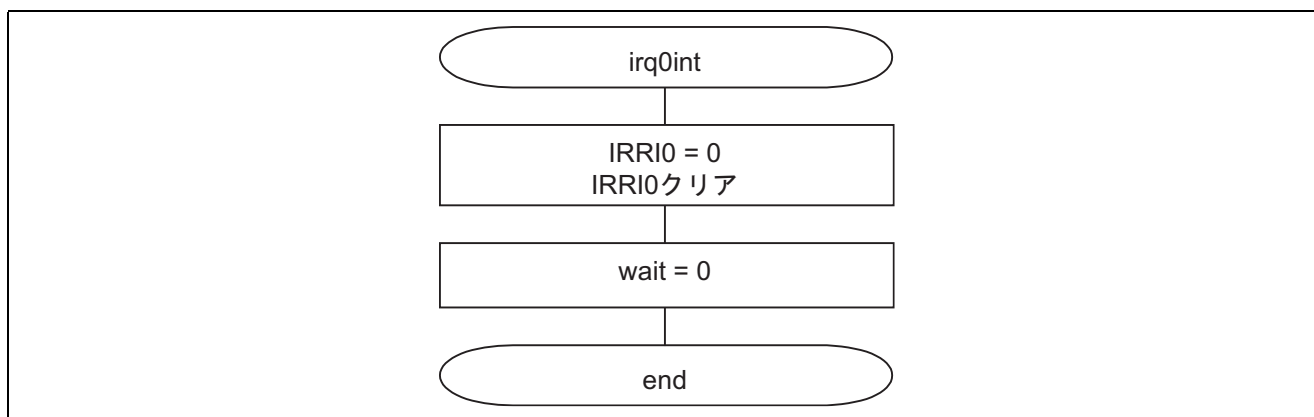


5.2 init\_SCI 3





5.3 irq0int



5.4 リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'0008
P	H'0100
B	H'FB80



## 6. プログラムリスト

```

/*****/
/*                                     */
/* H8/300L Super Low Power Series      */
/*   -H8/38024 Series-                 */
/* Application Note                     */
/*                                     */
/* 'Clock Synchronous Serial Data Reception' */
/*                                     */
/* Function                             */
/* : Serial Communication Interface     */
/*   Clock Synchronous Serial Interface */
/*   -Master Receiving                 */
/*                                     */
/* External Clock : 10MHz              */
/* Internal Clock : 5MHz               */
/* Sub Clock      : 32.768kHz          */
/*                                     */
/*****/

#include <machine.h>

/*****/
/* Symbol Definition                   */
/*****/
struct BIT {
    unsigned char  b7:1;    /* bit7 */
    unsigned char  b6:1;    /* bit6 */
    unsigned char  b5:1;    /* bit5 */
    unsigned char  b4:1;    /* bit4 */
    unsigned char  b3:1;    /* bit3 */
    unsigned char  b2:1;    /* bit2 */
    unsigned char  b1:1;    /* bit1 */
    unsigned char  b0:1;    /* bit0 */
};

#define SPCR      *(volatile unsigned char *)0xFF91 /* Transmit Data Register */
#define SPCR_BIT (*(struct BIT *)0xFF91)          /* Port Mode Register 1 */
#define SCINV2    SPCR_BIT.b2                    /* RXD Input data reversal switching */
#define SMR_BIT   (*(struct BIT *)0xFFA8)         /* Serial Mode Register */
#define COM       SMR_BIT.b7                     /* Communication Mode */
#define CHR       SMR_BIT.b6                     /* Character Length */
#define PE        SMR_BIT.b5                     /* Parity Enable */
#define PM        SMR_BIT.b4                     /* Parity Mode */
#define STOP      SMR_BIT.b3                     /* Stop Bit Length */
#define MP        SMR_BIT.b2                     /* Multiprocessor Mode */
#define CKS1      SMR_BIT.b1                     /* Clock Select 1 */
#define CKS0      SMR_BIT.b0                     /* Clock Select 0 */
#define BRR       *(volatile unsigned char *)0xFFA9 /* Bit Rate Register */
#define SCR3      *(volatile unsigned char *)0xFFAA /* Serial Control Register 3 */
#define SCR3_BIT  (*(struct BIT *)0xFFAA)         /* Serial Control Register 3 */
#define TIE       SCR3_BIT.b7                    /* Transmit Interrupt Enable */
#define RIE       SCR3_BIT.b6                    /* Receive Interrupt Enable */
#define TE        SCR3_BIT.b5                    /* Transmit Enable */
#define RE        SCR3_BIT.b4                    /* Receive Enable */
#define MPIE      SCR3_BIT.b3                    /* Multiprocessor Interrupt Enable */

```

```

#define TEIE      SCR3_BIT.b2          /* Transmit End Interrupt Enable */
#define CKE1      SCR3_BIT.b1          /* Clock Enable 1 */
#define CKE0      SCR3_BIT.b0          /* Clock Enable 0 */
#define TDR       *(volatile unsigned char *)0xFFAB /* Transmit Data Register */
#define SSR       *(volatile unsigned char *)0xFFAC /* Serial Status Register */
#define SSR_BIT   (*(struct BIT *)0xFFAC) /* Serial Status Register */
#define TDRE      SSR_BIT.b7          /* Transmit Data Register Empty */
#define RDRF      SSR_BIT.b6          /* Receive Data Register Full */
#define OER       SSR_BIT.b5          /* Overrun Errorr */
#define FER       SSR_BIT.b4          /* Framing Errorr */
#define PER       SSR_BIT.b3          /* Parity Errorr */
#define TEND      SSR_BIT.b2          /* Transmit End */
#define MPBR      SSR_BIT.b1          /* Multiprocessor Bit Receive */
#define MPBT      SSR_BIT.b0          /* Multiprocessor Bit Transfer */
#define RDR       *(volatile unsigned char *)0xFFAD /* Receive Data Register */

#define PMR2      *(volatile unsigned char *)0xFFC9 /* Port Mode Register 2 */
#define PMR2_BIT  (*(struct BIT *)0xFFC9)
#define IRQ0      PMR2_BIT.b0          /* P43/IRQ0 Select */
#define IEGR      *(volatile unsigned char *)0xFFFF2 /* Interrupt Edge Select Register 1 */
#define IEGR_BIT  (*(struct BIT *)0xFFFF2)
#define IEG0      IEGR_BIT.b0          /* IEG0 Edge Select */
#define IENR1     *(volatile unsigned char *)0xFFFF3 /* Interrupt Enable Register 1 */
#define IENR1_BIT (*(struct BIT *)0xFFFF3)
#define IEN0      IENR1_BIT.b0          /* IRQ0 Interrupt Enable */
#define IRR1      *(volatile unsigned char *)0xFFFF6 /* Interrupt Request Register 1 */
#define IRR1_BIT  (*(struct BIT *)0xFFFF6)
#define IRRIO     IRR1_BIT.b0          /* IRQ0 Interrupt Request Register */

#pragma interrupt (irq0int)
/*****
/* Function define */
*****/
void main ( void );
void init_SCI3( void );
void irq0int( void );

/*****
/* RAM define */
*****/
unsigned char SRD[4], wait;

/*****
/* Vector Address */
*****/
#pragma section V1 /* VECTOR SECTOIN SET */
void (*const VEC_TBL1[])(void) = {
    main /* 00 Reset */
};

#pragma section V2 /* VECTOR SECTOIN SET */
void (* const VEC_TBL2[])(void) = {
    irq0int /* 08 IRQ0 Interrupt */
};

#pragma entry main(sp=0xFF80)
#pragma section /* P */

```

```

/*****/
/* Main Program */
/*****/
void main ( void )
{
    unsigned char i;

    set_ccr(0x80); /* interrupt disable */

    SRD[0] = 0x00; /* Initialize Serial Receiving Data 0 */
    SRD[1] = 0x00; /* Initialize Serial Receiving Data 1 */
    SRD[2] = 0x00; /* Initialize Serial Receiving Data 2 */
    SRD[3] = 0x00; /* Initialize Serial Receiving Data 3 */

    IEG0 = 0; /* Initialize IRQ0 Terminal Input Edge */
    IRRIO = 0; /* Initialize IRQ0 Interrupt Request Flag */
    IRQ0 = 1;
    IEN0 = 1; /* IRQ0 Interrupt Enable */

    wait = 1;

    set_imask_ccr(0); /* interrupt enable */

    while(wait);

    set_ccr(0x80); /* interrupt disable */

    init_SCI3(); /* initilaize SCI3 */

    for(i=0; i<4; i++){ /* Serial Transmitting Data Counter 4 Loop */

        if (OER == 1){ /* Overrun Erorr Flag = 1 ? */
            OER = 0;
            SRD[0] = 0xFF; /* Overrun Erorr 0 */
            SRD[1] = 0xFF; /* Overrun Erorr 1 */
            SRD[2] = 0xFF; /* Overrun Erorr 2 */
            SRD[3] = 0xFF; /* Overrun Erorr 3 */
            break;
        }

        while(RDRF == 0); /* Receive ? */

        SRD[i] = RDR; /* Save Serial Receiving Data */
    }

    RE = 0; /* Serial Receiving disable */

    while(1);
}

/*****/
/* SCI3 initialize */
/*****/
void init_SCI3( void )
{
    volatile unsigned char i;

```

```

SCR3 = 0x00;          /* initialize SCR3          */

CKE1 = 0;             /* Initialize Clock Enable 1 Output */
CKE0 = 0;             /* Initialize Clock Enable 0 Output */

COM = 1;              /* Initialize Communication Mode     */
MP = 0;               /* Initialize Multiprocessor Mode    */
CKS1 = 0;             /* Initialize Clock Select 1        */
CKS0 = 0;             /* Initialize Clock Select 0        */

BRR = 4;
for(i=0; i<4; i++);  /* Serial Transmitting Data Counter 4 Loop */

SPCR = 0xC0;

if(OER == 1)
    OER = 0;          /* Clear OER          */
if(FER == 1)
    FER = 0;          /* Clear FER          */
if(PER == 1)
    PER = 0;          /* Clear PER          */

if(RDRF == 1)
    RDRF = 0;        /* Clear RDRF        */

RE = 1;              /* Serial Receiving enable */
}

/*****/
/* IRQ0 interrupt          */
/*****/
void irq0int( void )
{
    IRRIO = 0;        /* Initialize IRQ0 Interrupt Request Flag */

    wait = 0;
}

```

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2004.07.28	—	初版発行

### 安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

### 本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ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ルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾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の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 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全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ(<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。
4. 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。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位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。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。