

# RX62T グループ

## シングルチップモードによる UART 経由での 内蔵フラッシュメモリ書き換え (スレーブ)

R01AN0640JJ0102

Rev.1.02

2015.03.04

### 要旨

本アプリケーションノートは、RX62T グループ アプリケーションノート「シングルチップモードによる UART 経由での内蔵フラッシュメモリ書き換え (マスタ)」(R01AN0639JJ) から調歩同期式シリアル通信で送信される、消去ブロック番号、書き込みデータのサイズ、および書き込みデータを使用してコード格納用フラッシュメモリ (ユーザマット) の書き込み/消去する処理について説明しています。

調歩同期式シリアル通信で消去ブロック番号、書き込みデータのサイズ、および書き込みデータを送信する処理に関しては、RX62T グループ アプリケーションノート「シングルチップモードによる UART 経由での内蔵フラッシュメモリ書き換え (マスタ)」(R01AN0639JJ) をご参考ください。また、内蔵フラッシュメモリの消去処理および書き込み処理は、アプリケーションノート「RX600 シリーズ RX600 用のシンプルフラッシュ API」Rev.2.20 (R01AN0544JU0220) を使用しています。

### 動作確認デバイス

#### RX62T グループ

RX62T グループと同様の I/O レジスタ (周辺装置制御レジスタ) を持つ他の RX ファミリでも本プログラムを使用することができます。ただし、一部の機能を機能追加等で変更している場合がありますのでマニュアルで確認してください。このアプリケーションノートをご使用に際しては十分な評価を行ってください。

### 目次

1. 仕様 .....	2
2. 動作確認環境 .....	4
3. 使用機能 .....	4
4. 動作説明 .....	5
5. ソフトウェア説明 .....	15
6. 使用上の注意事項 .....	34
7. 参考ドキュメント .....	35

## 1. 仕様

- 本アプリケーションノートでは、RX62T グループの R5F562TAADFP を用いて、シングルチップモードでユーザマットの書き込み / 消去を行います。
- スレーブは、マスタから消去ブロック番号、書き込みデータのサイズ、および書き込みデータを調歩同期式シリアル通信で受信し、ユーザマットの書き込み / 消去を行います。
- マスタとスレーブ間の調歩同期式シリアル通信は、SCI チャンネル 0（SCI0）モジュールを使用します。
- 調歩同期式シリアル通信仕様は、
  - ビットレート：31250bps
  - データ長：8 ビット
  - パリティビット：なし
  - ストップビット：1 ビット
 とします。
- 本アプリケーションノートでは、スレーブは指定された消去ブロック（EB08：16K バイト）の消去を行い、受信した 8K バイト（256 バイト × 32）の書き込みデータを消去ブロック EB08 の先頭アドレスから書き込みます。なお、本アプリケーションノートでは EB00 と EB01 にプログラムを配置し、消去 / 書き込みできない仕様になっています。したがってマスタから EB00 や EB01 の消去ブロック番号を指定された場合はエラーになります。
- スレーブとマスタは通信制御するためにハンドシェイクしています。スレーブはマスタから受信したデータに対応する処理を実行後、マスタへ [ACCEPTABLE] コマンド（55h）を送信します。マスタはスレーブから [ACCEPTABLE] コマンドを受信すると次のシリアル送信を開始します。
- スレーブは、正常にユーザマットの消去 / 書き込み処理が完了すると、I/O ポートに接続された 4 個の LED で正常終了を知らせます。また、マスタとの通信中および消去 / 書き込み処理中にエラーが発生した場合は、LED でそのエラー状態を知らせます。

図 1 に本アプリケーションノートにおける仕様を示します。

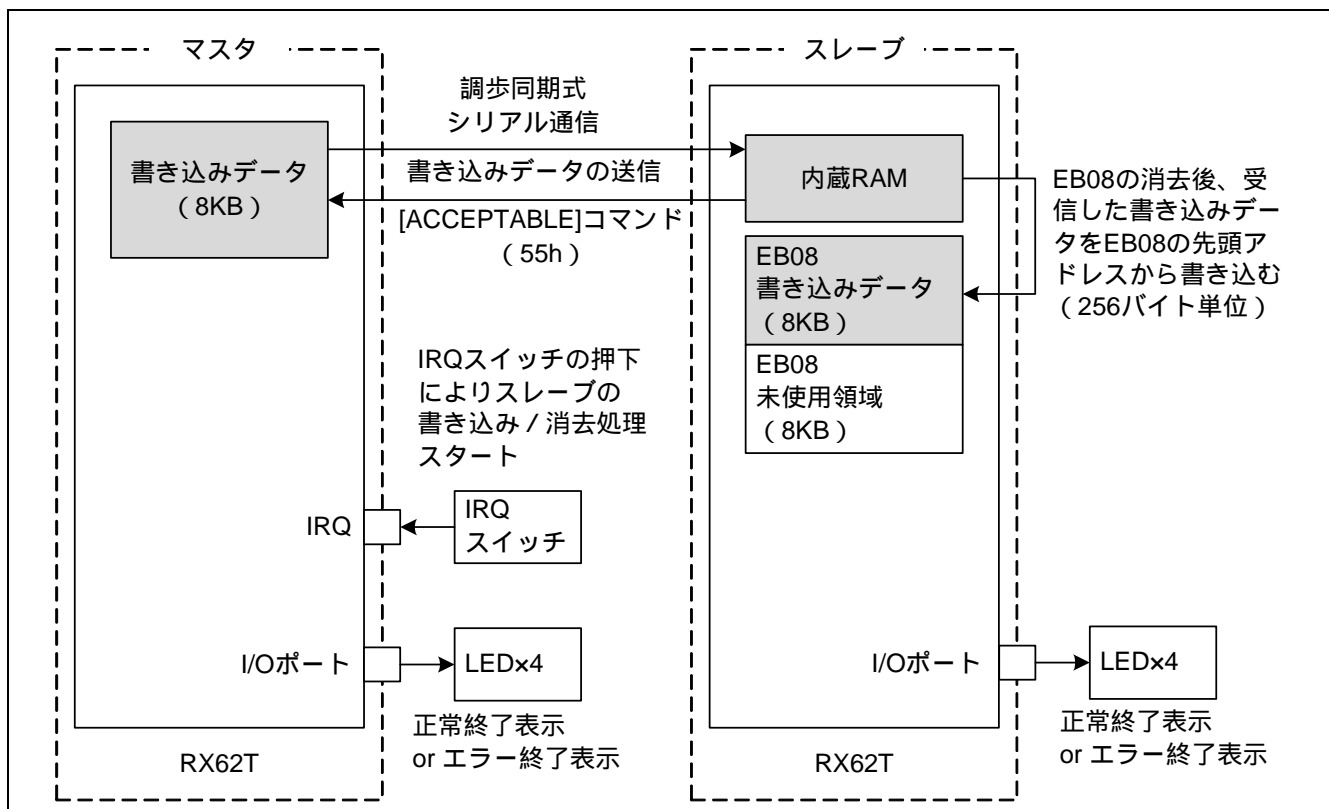


図1 仕様

図 2 に本アプリケーションノートにおけるスレーブのハードウェア構成図を示します。

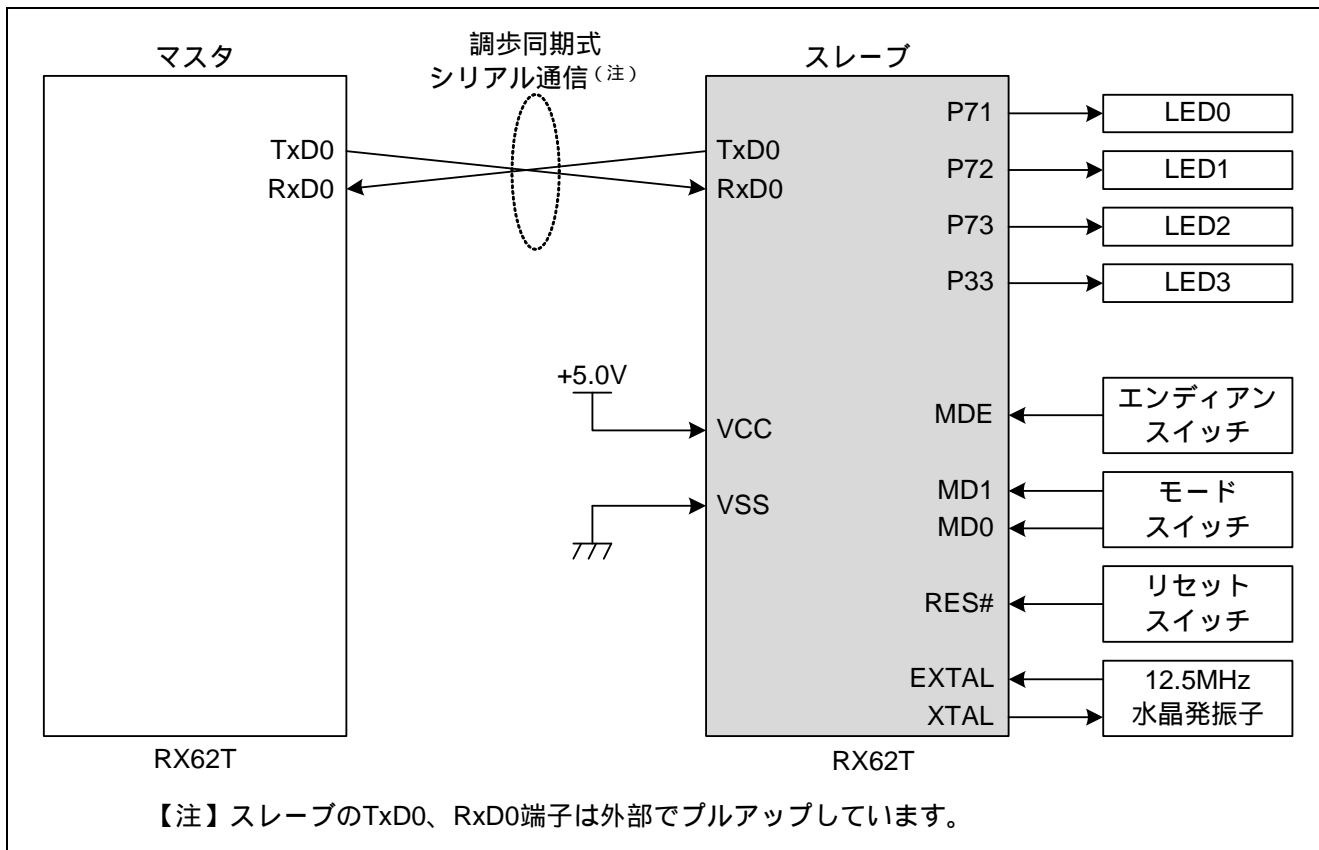


図2 スレーブのハードウェア構成図

## 2. 動作確認環境

スレーブの動作確認を行った環境を表 1 に示します。

表1 スレーブの動作確認環境

項目	内容
デバイス	RX62T グループ : R5F562TAADFP (ROM 容量 : 256K バイト、RAM 容量 : 16K バイト)
ボード	Renesas Starter Kit 【開発中 (2012/2/21 現在)】
電源電圧	5.0V
入力クロック	12.5MHz (ICLK = 100MHz、PCLK = 50MHz)
動作温度	室温
High-performance Embedded Workshop	Version 4.09.00.007
Toolchain	RX Standard Toolchain (V.1.2.0.0)
Debugger/Emulator	E1 エミュレータ
Debugger component	RX E1/E20 SYSTEM V.1.02.00
最適化リンケージエディタ (rom オプション) *1	-rom=D=R,D_1=R_1,D_2=R_2,PFRAM=RPFRAM

【注】 \*1 「6.3 rom オプションについて」を参照してください。

## 3. 使用機能

- クロック発生回路
- 低消費電力低減機能
- 割り込みコントローラ
- I/O ポート
- シリアルコミュニケーションインタフェース
- ROM (コード格納用フラッシュメモリ)

詳細は 7. 参考ドキュメント「ユーザズマニュアル」を参照してください。

## 4. 動作説明

### 4.1 動作モードの設定

本アプリケーションノートでは、スレーブのモード端子を MD1 = 1、MD0 = 1 に設定し動作モードをシングルチップモードに、システムコントロールレジスタ 0 (SYSCR0) の ROME ビットを 1 に設定し内蔵 ROM を有効に設定しています。

スレーブはシングルチップモードでユーザマットから起動します。

表 2 に本アプリケーションノートにおけるスレーブの動作モードの設定を示します。

表2 スレーブの動作モードの設定

モード端子		SYSCR0 レジスタ	動作モード	内蔵 ROM
MD1	MD0	ROME		
1	1	1	シングルチップモード	有効

【注】 SYSCR0 レジスタの ROME ビットの初期値は、SYSCR0.ROME = 1 のため、プログラム中で SYSCR0 レジスタの設定は行っておりません。

### 4.2 クロックの設定

本アプリケーションノートで使用している評価ボードには 12.5MHz の水晶発振子が搭載されています。

そのため、本アプリケーションノートではシステムクロック (ICLK) および周辺モジュールクロック (PCLK) をそれぞれ、8 通倍 (100MHz)、4 通倍 (50MHz) に設定しています。

### 4.3 エンディアンの設定

本アプリケーションノートは、ビッグエンディアン/リトルエンディアンの両方に対応しています。ハードウェア (MDE 端子) によるエンディアンの設定を表 3 に示します。なお、マスタとスレーブのエンディアンは合わせてください。

表3 エンディアン設定 (ハードウェア)

MDE 端子	エンディアン
0	リトルエンディアン
1	ビッグエンディアン

コンパイラオプションによるエンディアンの設定を表 4 に示します。

表4 エンディアン設定 (コンパイラオプション)

マイコンオプション	エンディアン
endian = little	リトルエンディアン
endian = big	ビッグエンディアン

【注】 コンパイラオプションで選択したエンディアンに合わせて、MDE 端子によるエンディアンを設定してください。

4.4 調歩同期式シリアル通信仕様

本アプリケーションノートでは、マスタとスレーブ間で調歩同期式シリアル通信により、通信コマンド ([FSTART]、[ERASE]、[WRITE])、消去ブロック番号、書き込みデータサイズ、書き込みデータの受信を行います。また、スレーブからはハンドシェイク用のステータスコマンドとして[ACCEPTABLE]コマンド (55h) を送信します。使用する SCI0 の TxD0、RxD0 端子はそれぞれ外部でプルアップしています。

表 5 に調歩同期式シリアル通信仕様を示します。

表5 調歩同期式シリアル通信仕様

項目	仕様
チャンネル	SCI チャンネル 0 ( SCI0 )
コミュニケーションモード	調歩同期式モード
ビットレート	31250bps ( PCLK=50MHz 時 )
データ長	8 ビット
パリティビット	なし
ストップビット	1 ビット
エラー	オーバランエラー、フレーミングエラー

4.4.1 通信フロー

マスタとスレーブ間の通信フローを図 3 ~ 図 6 に示します。

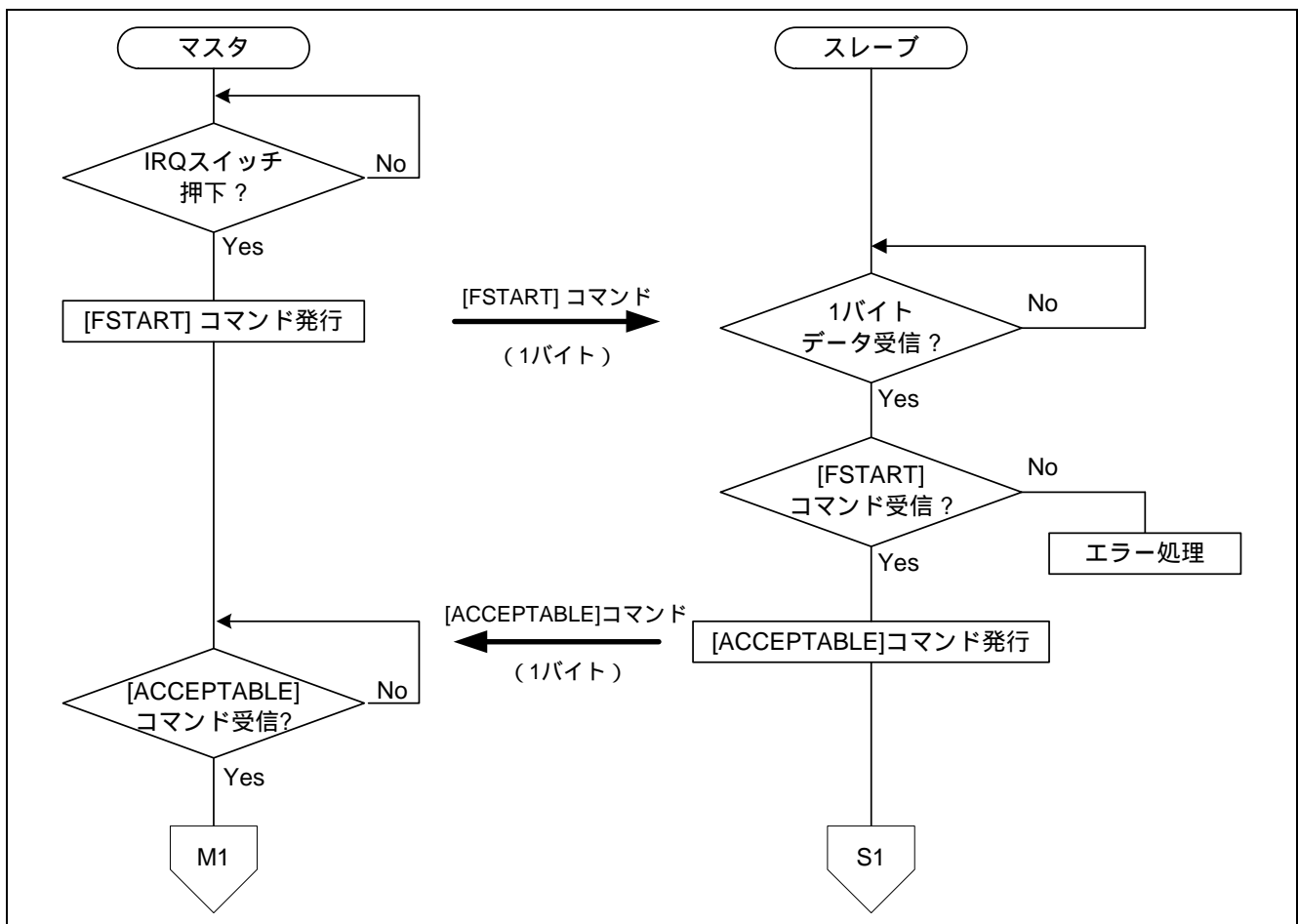


図3 通信フロー (1)

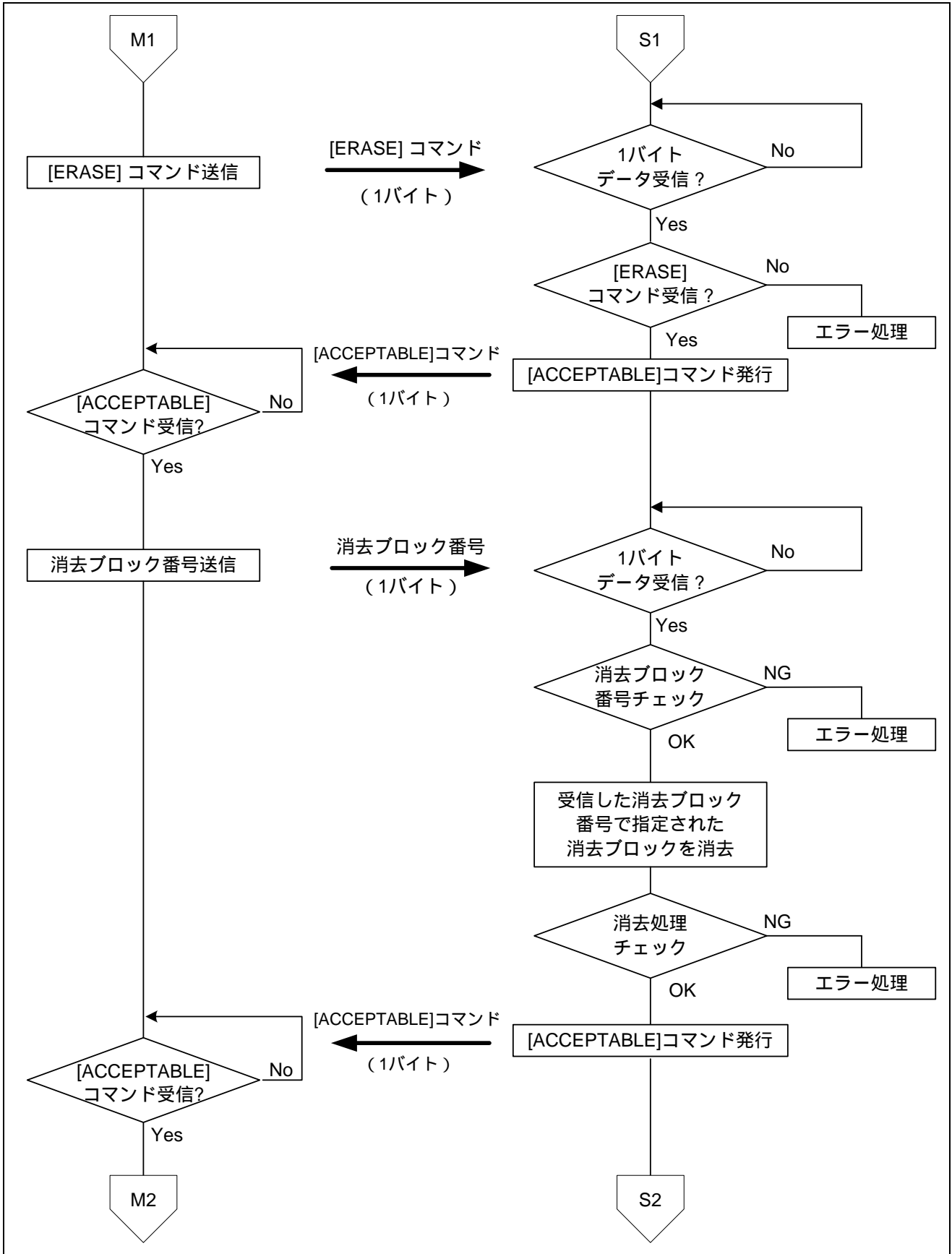


図4 通信フロー (2)

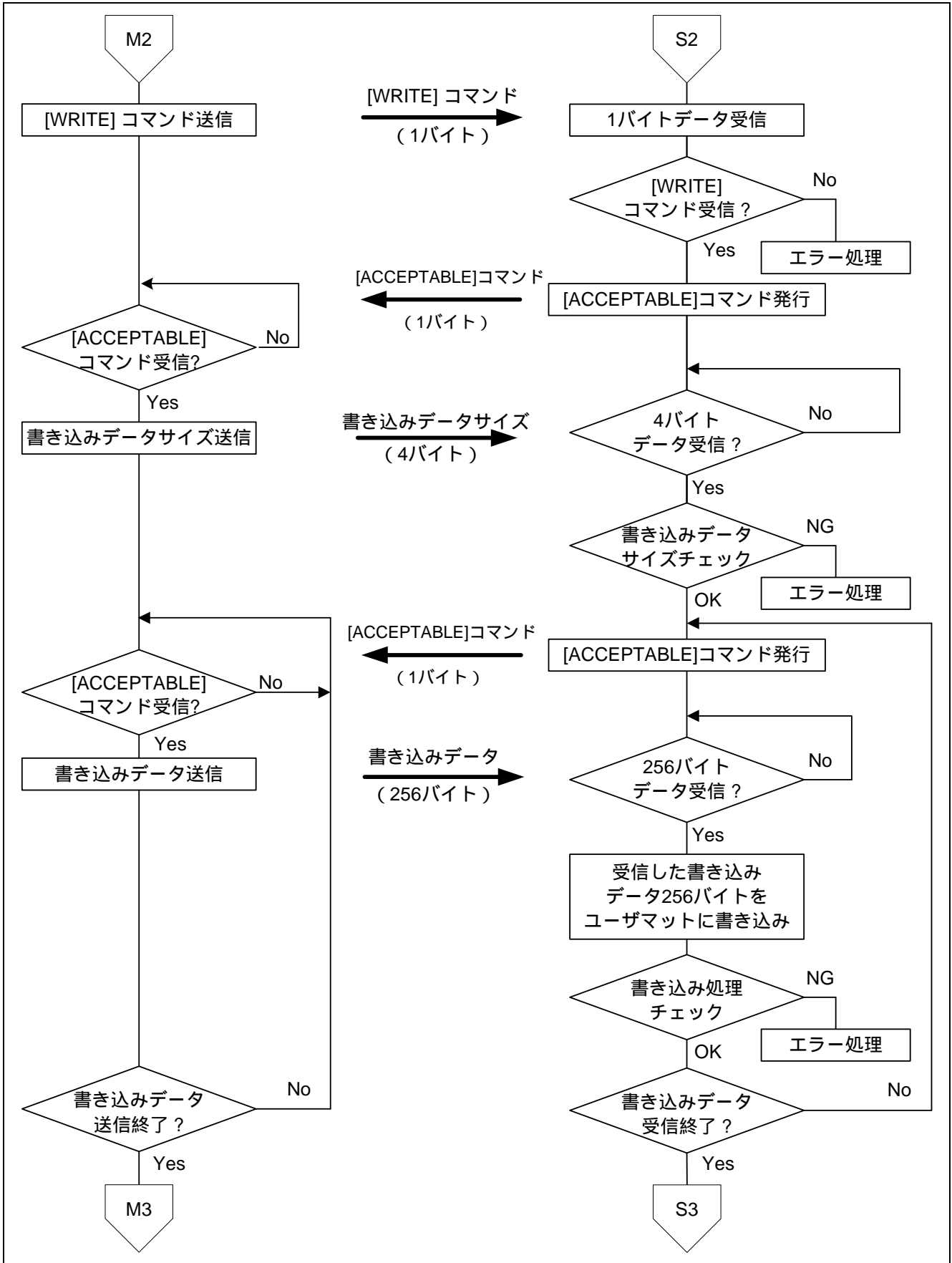


図5 通信フロー (3)



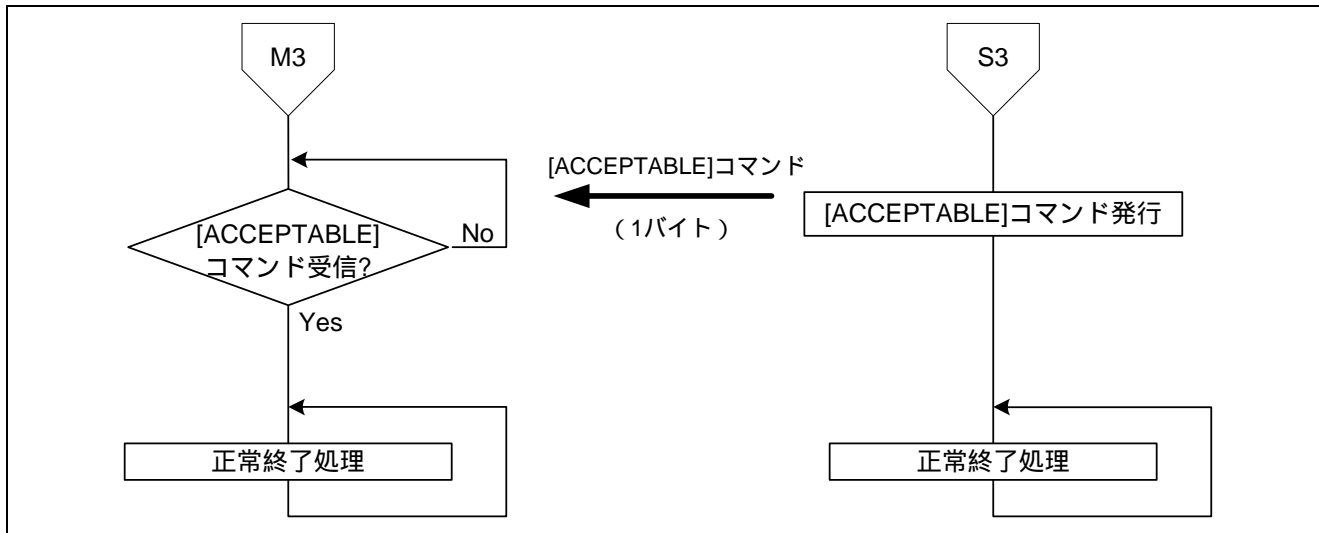


図6 通信フロー (4)

4.4.2 通信コマンド仕様

マスタとスレーブ間の通信コマンドの仕様を表 6 に示します。

表6 通信コマンド仕様

コマンド	値	説明	通信方向
FSTART	10h	スレーブのユーザマットの書き込み / 消去処理を開始するためのコマンド	マスタ→スレーブ
ERASE	11h	スレーブのユーザマットの消去を開始するためのコマンド	マスタ→スレーブ
WRITE	12h	スレーブのユーザマットの書き込みを開始するためのコマンド	マスタ→スレーブ
ACCEPTABLE	55h	スレーブがマスタへデータ受信可能であることを通知するためのステータスコマンド	スレーブ→マスタ

4.4.3 消去ブロック番号

スレーブは、マスタから[ERASE]コマンド受信後に 1 バイトの消去ブロック番号 (定義された 1 バイトのデータ) を受信します。

図 7 に消去ブロック番号の仕様を示します。また、消去ブロック番号の詳細に関しては「4.5.1 ユーザマットの消去処理」を参照してください。

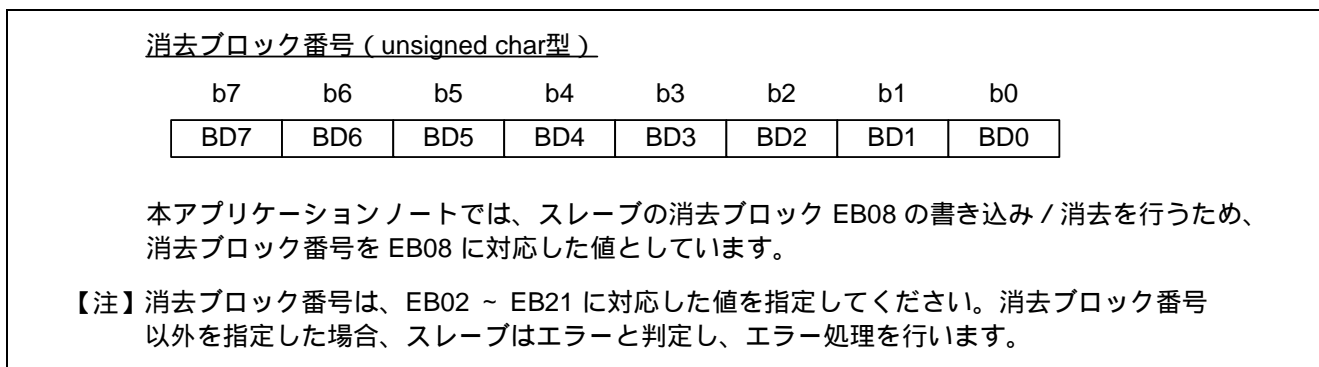


図7 消去ブロック番号仕様

#### 4.4.4 書き込みデータサイズ

スレーブは、マスタから[WRITE]コマンド受信後に 4 バイトの書き込みデータサイズを受信します。図 8 に書き込みデータサイズの仕様を示します。

##### 書き込みデータサイズ (unsigned long型)

b31	b30	b29	b28	b27	b26	b25	b24
SZ31	SZ30	SZ29	SZ28	SZ27	SZ26	SZ25	SZ24
b23	b22	b21	b20	b19	b18	b17	b16
SZ23	SZ22	SZ21	SZ20	SZ19	SZ18	SZ17	SZ16
b15	b14	b13	b12	b11	b10	b9	b8
SZ15	SZ14	SZ13	SZ12	SZ11	SZ10	SZ09	SZ08
b7	b6	b5	b4	b3	b2	b1	b0
SZ07	SZ06	SZ05	SZ04	SZ03	SZ02	SZ01	SZ00

本アプリケーションノートでは、書き込みサイズを8Kバイトにしているため、書き込みデータサイズを [0000 2000h] としています。

- 【注】 1. 書き込みデータサイズは、0より大きい値かつ消去ブロック番号で指定した消去ブロックサイズ以下としてください。0の場合もしくは消去ブロック番号で指定した消去ブロックより大きいサイズを指定した場合、スレーブはエラーと判定し、エラー処理を行います。
2. 書き込みデータの送信は256バイト固定としています。したがって、書き込みデータサイズが256バイトの倍数でない場合、マスタは256バイトごとに送信を行っていき、最後の256バイトに満たない書き込みデータに関してはFFhを追加して256バイトの書き込みデータとしてスレーブに送信します。

図8 書き込みデータサイズ仕様

#### 4.4.5 ハンドシェイク制御

スレーブは通信制御するためにマスタとハンドシェイクしています。

ハンドシェイク制御としてスレーブはマスタからのシリアル通信受信後に、受信データに対する処理を実行し、次のシリアル通信が受信可能になってから[ACCEPTABLE]コマンド (55h) をマスタへ返信します。マスタはスレーブから[ACCEPTABLE]コマンドを受信すると次のシリアル送信を開始します。

#### 4.4.6 オーバランエラー

本アプリケーションノートでは、スレーブの調歩同期式シリアル通信の受信時にオーバランエラーが発生 (SCIO.SSR.ORER ビットが 1 にセット) した場合には、エラー処理します。

#### 4.4.7 フレーミングエラー

本アプリケーションノートでは、スレーブの調歩同期式シリアル通信の受信時にフレーミングエラーが発生 (SCIO.SSR.FER ビットが 1 にセット) した場合には、エラー処理します。

## 4.5 ユーザマットの書き込み / 消去

本アプリケーションノートの書き込みおよび消去で使用している「RX600用のシンプルフラッシュ API」については、7. 参考ドキュメント「RX600用のシンプルフラッシュ API」を参照してください。

### 4.5.1 ユーザマットの消去処理

ユーザマットの消去処理は「RX600用のシンプルフラッシュ API」で提供される R\_FlashErase 関数を使用しています。したがって、消去するブロック番号も R\_FlashErase 関数に指定する値と共通にしています。消去エラーは R\_FlashErase 関数から戻る値となります。

### 4.5.2 ユーザマットの書き込み処理

ユーザマットの書き込みは「RX600用のシンプルフラッシュ API」で提供される R\_FlashWrite 関数を使用しています。書き込みエラーは R\_FlashWrite 関数から戻る値となります。

### 4.5.3 シンプルフラッシュ API の変更点

シンプルフラッシュ API から“r\_flash\_api\_rx600\_config.h”を変更しています。

表 7 に“r\_flash\_api\_rx600\_config.h”の変更点を示します。

表7 「r\_flash\_api\_rx600\_config.h」の変更点

変更項目	変更箇所
シンプルフラッシュ API の設定変更	<pre> #define IGNORE_LOCK_BITS #define COPY_CODE_BY_API #define FLASH_API_USE_R_BSP                     </pre>

### 4.5.4 割り込みを使用する場合の注意点

FLASH ROM の書き込み、および消去中は ROM へのアクセスはできません。したがって、ROM の書き込み、および消去中に発生する割り込みによる ROM へのアクセスを抑制する必要があります。本アプリケーションノートでは割り込みを使用していませんが、サンプルプログラムとして割り込みの抑止例をコメントアウトとして記載しています。

割り込みの抑止例を以下に示します。

```

/* Holds IPL of processor before flash operation */
// static unsigned char flash_pipl;

/* Save current processor IPL */
// flash_pipl = get_ipl(); /* If your system is using the interrupt, Enable this line. */

/* Set the processor IPL so that interrupts that access ROM will not occur during ROM program/erase
operations. */
// set_ipl(FLASH_READY_IPL); /* If your system is using the interrupt, Enable this line. */

/* Erasure process by using "simple API" */
fcu_status = R_FlashErase((uint8_t)target_eb);

/* Programming process by using "simple API" */
fcu_status = R_FlashWrite((uint32_t)fcu_info.p_write_adrs_now,
(uint32_t)wrd_data_buffer, BUF_SIZE); /* Program 256 byte data to the target EB by using "simpleAPI"
* (call of R_FlashWrite function in "simpleAPI") */

/* Restore processor IPL */
// set_ipl(flash_pipl); /* If your system is using the interrupt, Enable this line. */

```

## 4.6 LED 接続

スレーブの I/O ポートに接続されている LED0~LED3 の接続図を図 9 に示します。

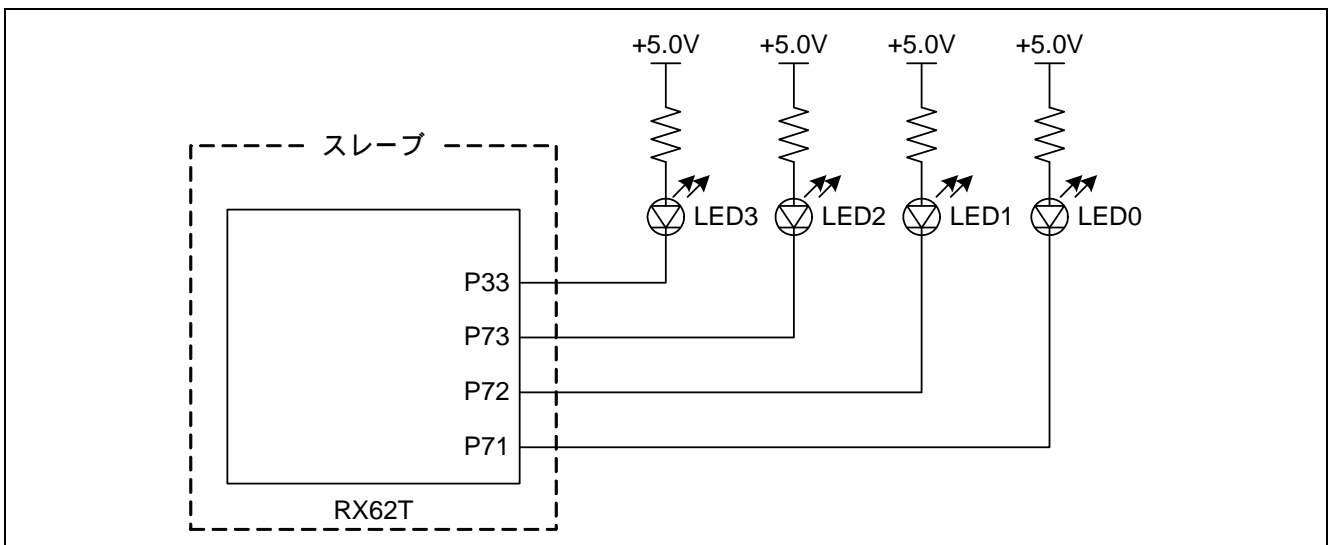


図9 スレーブのLED接続図

図9に示すように、I/Oポート(P71、P72、P73およびP33)から“High”を出力するとLED0~LED3は消灯、“Low”を出力するとLED0~LED3は点灯します。表8にI/Oポート出力とLEDの状態を示します。

表8 スレーブのI/Oポート出力とLEDの状態

I/Oポート	レジスタ設定	I/Oポート状態	LED状態	
P71	PORT7.DR.B1 = 1、PORT7.DDR.B1 = 1	“High”出力	LED0	消灯
	PORT7.DR.B1 = 0、PORT7.DDR.B1 = 1	“Low”出力		点灯
P72	PORT7.DR.B2 = 1、PORT7.DDR.B2 = 1	“High”出力	LED1	消灯
	PORT7.DR.B2 = 0、PORT7.DDR.B2 = 1	“Low”出力		点灯
P73	PORT7.DR.B3 = 1、PORT7.DDR.B3 = 1	“High”出力	LED2	消灯
	PORT7.DR.B3 = 0、PORT7.DDR.B3 = 1	“Low”出力		点灯
P33	PORT3.DR.B3 = 1、PORT3.DDR.B3 = 1	“High”出力	LED3	消灯
	PORT3.DR.B3 = 0、PORT3.DDR.B3 = 1	“Low”出力		点灯

#### 4.7 正常終了処理

スレーブは、ユーザマットの書き込み / 消去が正常に終了すると、接続されている 4 個の LED に正常終了の表示を行います。正常終了の表示は、LED0～LED3 を順番に 1 つずつ点灯する処理を繰り返します。

#### 4.8 エラー処理

本アプリケーションノートにおけるスレーブのエラー一覧を表9に示します。スレーブのエラー処理では、エラー状態を接続されている 4 個の LED に表示する処理を行います。

表9 スレーブのエラー一覧

：点灯、 ：消灯

エラー番号	内容	LED 表示			
		LED3	LED2	LED1	LED0
エラーNo.01	オーバランエラーまたはフレーミングエラーが発生した場合				
エラーNo.02	[FSTART]コマンド受信待ち状態でマスタから受信したコマンドが[FSTART]コマンドでなかった場合				
エラーNo.03	[ERASE]コマンド受信待ち状態でマスタから受信したコマンドが[ERASE]コマンドでなかった場合				
エラーNo.04	マスタから受信した消去ブロック番号が、EB02～EB21 でなかった場合				
エラーNo.05	消去ブロックの消去処理でエラーが発生した場合				
エラーNo.06	[WRITE]コマンド受信待ち状態でマスタから受信したコマンドが[WRITE]コマンドでなかった場合				
エラーNo.07	マスタから受信した書き込みデータサイズが、0 もしくは消去ブロック番号で指定されたブロックサイズより大きい場合				
エラーNo.08	データの書き込み処理でエラーが発生した場合				

## 4.9 セクション設定

スレーブのセクション設定を表 10 に示します。

表10 スレーブのセクション設定

セクション名	開始アドレス	説明	
RPFRAM	0000 0000h	[PFRAM]セクションを ROM 化支援オプションにより RAM 上にマップした領域	
B_1	0000 1000h	未初期化データ領域 (ALIGN = 1)	
R_1		[D_1]セクションを ROM 化支援オプションにより RAM 上にマップした領域	
B_2		未初期化データ領域 (ALIGN = 2)	
R_2		[D_2]セクションを ROM 化支援オプションにより RAM 上にマップした領域	
B		未初期化データ領域 (ALIGN = 4)	
R		[D]セクションを ROM 化支援オプションにより RAM 上にマップした領域	
SI		割り込みスタック領域	
PRResetPRG		FFFF E000h	プログラム領域 (PowerON_Reset_PC プログラム)
P	プログラム領域		
PIntPRG	プログラム領域 (割り込みプログラム)		
C_1	FFFF E800h	定数領域 (ALIGN = 1)	
C_2		定数領域 (ALIGN = 2)	
C		定数領域 (ALIGN = 4)	
C\$*		未初期化データ領域のセクション初期化用テーブル, 可変ベクタ領域	
D_1		初期化データ領域 (ALIGN = 1)	
D_2		初期化データ領域 (ALIGN = 2)	
D		初期化データ領域 (ALIGN = 4)	
W*		switch 文分岐テーブル領域	
PFRAM		FFFF F000h	プログラム領域 (ユーザマットの書き込み / 制御プログラム)
FIXEDVECT		FFFF FFD0h	固定ベクタ領域

## 5. ソフトウェア説明

### 5.1 ファイル構成

スレーブのファイル構成を表 11 に示します。表 11 で示されたファイル以外は、High-performance Embedded Workshop が自動生成するファイルを使用しています。

表11 スレーブのファイル構成

ファイル名	内容
resetprg.c <sup>*1</sup>	初期設定処理
main.c	マスタとの調歩同期式シリアル通信による通信コマンドの送受信制御、消去ブロック番号、書き込みデータサイズ、および書き込みデータの受信制御、正常終了時およびエラー発生時の LED の表示制御
r_flash_api_rx600.c	シンプルフラッシュ API のプログラム <sup>*2</sup>
r_flash_api_rx600.h	シンプルフラッシュ API のプログラムの外部参照用インクルードヘッダ <sup>*2</sup>
r_flash_api_rx600_private.h	シンプルフラッシュ API のプログラムの外部参照用インクルードヘッダ <sup>*2</sup>
r_flash_api_rx600_config.h	シンプルフラッシュ API のパラメータ設定用インクルードヘッダ <sup>*2</sup>
mcu_info.h	シンプルフラッシュ API のパラメータ設定用インクルードヘッダ <sup>*2</sup>

【注】 \*1 High-performance Embedded Workshop が自動生成するファイルですが、本アプリケーションノートでは、PowerON\_Reset\_PC 関数内の HardwareSetup 関数の呼び出しのコメントアウトを解除して、main.c ファイル内の HardwareSetup 関数を PowerON\_Reset\_PC 関数から呼び出すように変更しています。

\*2 詳細はシンプルフラッシュ API のアプリケーションノートを参照してください。

### 5.2 関数構成

スレーブの関数一覧を表 12、スレーブの関数階層構造を図 10 に示します。

表12 スレーブの関数一覧

関数名	ファイル名	概要
PowerON_Reset_PC	resetprg.c	初期設定関数
HardwareSetup	main.c	MCU 初期設定関数
Main	main.c	メイン関数
Flash_Update	main.c	データ送受信制御、およびエラー処理制御関数
R_FlashErase	r_flash_api_rx600.c	ユーザマツ消去制御関数
R_FlashWrite	r_flash_api_rx600.c	ユーザマツ書き込み制御関数
Indicate_Ending_LED	main.c	正常終了処理関数
Indicate_Error_LED	main.c	エラー終了処理関数
SCI_Rcv1byte	main.c	1 バイトデータ受信関数
SCI_Rcvnbyte	main.c	n バイトデータ受信関数
SCI_Trns1byte	main.c	1 バイトデータ送信関数

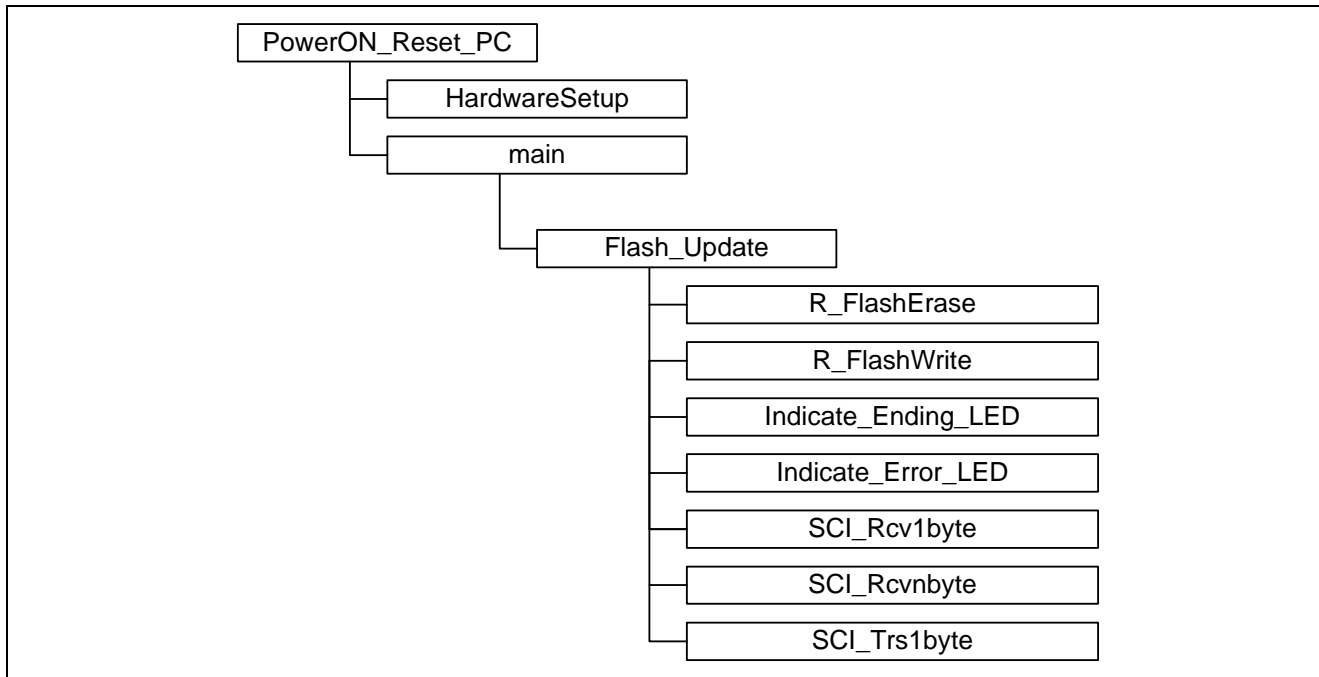


図10 スレーブの関数階層構造

### 5.3 記号定数説明

スレーブが使用する記号定数を表 13 に示します。

表13 スレーブの記号定数一覧

記号定数名	設定値	内容	使用関数
FSTART	0x10	書き込み / 消去開始コマンド	main
ERASE	0x11	消去開始コマンド	Flash_Update
WRITE	0x12	書き込み開始コマンド	Flash_Update
ACCEPTABLE	0x55	マスタへ送信するステータスコマンド	main
LED_ON	0	LED 点灯時の設定値	Indicate_Ending_LED Indicate_Error_LED
LED_OFF	1	LED 消灯時の設定値	HardwareSetup Indicate_Ending_LED Indicate_Error_LED
RSK_LED0	PORT7.DR.BIT.B1	評価ボード搭載 LED0 の点灯 / 消灯制御	HardwareSetup main Indicate_Ending_LED Indicate_Error_LED
RSK_LED1	PORT7.DR.BIT.B2	評価ボード搭載 LED1 の点灯 / 消灯制御	HardwareSetup main Indicate_Ending_LED Indicate_Error_LED
RSK_LED2	PORT7.DR.BIT.B3	評価ボード搭載 LED2 の点灯 / 消灯制御	HardwareSetup main Indicate_Ending_LED Indicate_Error_LED



表 13 スレーブの記号定数一覧 (つづき)

記号定数名	設定値	内容	使用関数
RSK_LED3	PORT3.DR.BIT.B3	評価ボード搭載 LED3 の点灯 / 消灯制御	HardwareSetup main Indicate_Ending_LED Indicate_Error_LED
RSK_LED0_DDR	PORT7.DDR.BIT.B1	評価ボード搭載 LED0 の入出力制御	HardwareSetup
RSK_LED1_DDR	PORT7.DDR.BIT.B2	評価ボード搭載 LED1 の入出力制御	HardwareSetup
RSK_LED2_DDR	PORT7.DDR.BIT.B3	評価ボード搭載 LED2 の入出力制御	HardwareSetup
RSK_LED3_DDR	PORT3.DDR.BIT.B3	評価ボード搭載 LED3 の入出力制御	HardwareSetup
WAIT_SCI1BIT	1920	SCI0 の BRR レジスタ設定後の待機時間データ	HardwareSetup
WAIT_LED	2000000	スレーブのユーザマットの書き込み / 消去が正常に終了した際に表示する LED の点灯間隔の時間データ	Indicate_Ending_LED Indicate_Error_LED
SIZE_WRITE_BLOCK	128	ユーザマットへの書き込みサイズ (ワードサイズ)	Flash_Update
BUF_SIZE	256	書き込みデータ格納領域のサイズ	Flash_Update
RxD0_ICR	PORTB.ICR.BIT.B1	RxD0 の入力バッファ設定	HardwareSetup
ERROR_NO_01	1	エラー状態を示すデータ	Flash_Update Indicate_Error_LED
ERROR_NO_02	2		
ERROR_NO_03	3		
ERROR_NO_04	4		
ERROR_NO_05	5		
ERROR_NO_06	6		
ERROR_NO_07	7		
ERROR_NO_08	8		
WRITE_ADRS_TOP_16K	0x00FC0000	書き込み / 消去用アドレス空間におけるブロックサイズ 16K バイト領域の先頭アドレス	Flash_Update
WRITE_ADRS_TOP_4K	0x00FF8000	書き込み / 消去用アドレス空間におけるブロックサイズ 4K バイト領域の先頭アドレス	
BLK_SIZE_16K	16×1024	EB08 ~ EB21 の各ブロックサイズ	
BLK_SIZE_4K	4×1024	EB00 ~ EB07 の各ブロックサイズ	

## 5.4 RAM 変数説明

スレーブが使用する RAM 変数を表 14 に示します。

表14 スレーブの RAM 変数一覧

変数名	型	内容
wrdata_buffer[BUF_SIZE]	unsigned char	スレーブから受信した 256 バイトの書き込みデータを格納する配列（256 バイト）
fcu_info	ST_FCU_INFO* <sup>1</sup>	ユーザマットの書き込み / 消去時に使用する FCU 関連のアドレス情報を格納する構造体（28 バイト）
p_write_buffer	unsigned short*	ユーザマット書き込み時の書き込みデータ格納領域のアドレス：4 バイト
p_command_adrs	unsigned char*	FCU コマンド発行先アドレス（書き込み / 消去用アドレス）：4 バイト
p_erase_adrs	unsigned short*	消去時の対象消去ブロックの先頭アドレス（書き込み / 消去用アドレス）：4 バイト
p_write_adrs_top	unsigned short*	書き込み時の対象消去ブロックの先頭アドレス（書き込み / 消去用アドレス）：4 バイト
p_write_adrs_end	unsigned short*	書き込み時の対象消去ブロックの終了アドレス（書き込み / 消去用アドレス）：4 バイト
p_write_adrs_now	unsigned short*	書き込み時の書き込み先アドレス（書き込み / 消去用アドレス）：4 バイト
eb_block_size	unsigned long	対象消去ブロックのブロックサイズ：4 バイト

【注】 \*1 ST\_FCU\_INFO 型の詳細は「5.5 構造体説明」を参照してください。

## 5.5 構造体説明

スレーブが使用する構造体 ST\_FCU\_INFO の仕様を表 15 に示します。

表15 構造体 ST\_FCU\_INFO の仕様

メンバ名	型	内容
p_write_buffer	unsigned short*	ユーザマット書き込み時の書き込みデータ格納領域のアドレス
p_command_adrs	unsigned char*	FCU コマンド発行先アドレス（書き込み / 消去用アドレス）
p_erase_adrs	unsigned short*	消去時の対象消去ブロックの先頭アドレス（書き込み / 消去用アドレス）
p_write_adrs_top	unsigned short*	書き込み時の対象消去ブロックの先頭アドレス（書き込み / 消去用アドレス）
p_write_adrs_end	unsigned short*	書き込み時の対象消去ブロックの終了アドレス（書き込み / 消去用アドレス）
p_write_adrs_now	unsigned short*	書き込み時の書き込み先アドレス（書き込み / 消去用アドレス）
eb_block_size	unsigned long	対象消去ブロックのブロックサイズ

## 5.6 使用 I/O レジスタ説明

スレーブのプログラムで使用する I/O レジスタを以下に示します。なお、設定値は本アプリケーションノートで使用している値であり、初期値とは異なります。

## (1) クロック発生回路

- システムクロックコントロールレジスタ (SCKCR) ビット数: 32 ビット アドレス: 0008 0020h

ビット	シンボル	設定値	ビット名	機能	R/W
b11-b8	PCK[3:0]	0001	周辺モジュールクロック (PCLK) 選択ビット	0001: ×4 PCLK = 50MHz	R/W
b27-b24	ICK[3:0]	0000	システムクロック (ICLK) 選択ビット	0000: ×8 ICLK = 100MHz	R/W

## (2) I/O ポート

- ポート7データレジスタ (P7.DR) ビット数: 8 ビット アドレス: 0008 C027h

ビット	シンボル	設定値	ビット名	機能	R/W
b1	B1	0	P71 出力データ格納ビット	0: 出力データ = 0	R/W
		1		1: 出力データ = 1	
b2	B2	0	P72 出力データ格納ビット	0: 出力データ = 0	R/W
		1		1: 出力データ = 1	
b3	B3	0	P73 出力データ格納ビット	0: 出力データ = 0	R/W
		1		1: 出力データ = 1	

- ポート3データレジスタ (P3.DR) ビット数: 8 ビット アドレス: 0008 C023h

ビット	シンボル	設定値	ビット名	機能	R/W
b3	B3	0	P33 出力データ格納ビット	0: 出力データ = 0	R/W
		1		1: 出力データ = 1	

- ポート7データディレクションレジスタ (P7.DDR) ビット数: 8 ビット アドレス: 0008 C007h

ビット	シンボル	設定値	ビット名	機能	R/W
b1	B1	1	P71 入力/出力指定ビット	1: 出力ポート	R/W
b2	B2	1	P72 入力/出力指定ビット	1: 出力ポート	R/W
b3	B3	1	P73 入力/出力指定ビット	1: 出力ポート	R/W

- ポート3データディレクションレジスタ (P3.DDR) ビット数: 8 ビット アドレス: 0008 C003h

ビット	シンボル	設定値	ビット名	機能	R/W
b3	B3	1	P33 入力/出力指定ビット	1: 出力ポート	R/W

- ポートB入力バッファコントロールレジスタ (PB.ICR) ビット数: 8 ビット アドレス: 0008 C06Bh

ビット	シンボル	設定値	ビット名	機能	R/W
b1	B1	1	PB1 入力バッファ制御ビット	1: PB1 の入力バッファは有効	R/W

RX62T グループ

(3) 低消費電力低減機能

- モジュールストップコントロールレジスタ B (MSTPCRB) ビット数 : 32 ビット アドレス : 0008 0014h

ビット	シンボル	設定値	ビット名	機能	R/W
b31	MSTPB31	0	シリアルコミュニケーションインタフェース 0 モジュールストップ設定ビット	0: SCI0 のモジュールストップ状態の解除	R/W

(4) シリアルコミュニケーションインタフェース 0 (SCI0)

- SCI0 シリアルコントロールレジスタ (SCI0.SCR) ビット数 : 8 ビット アドレス : 0008 8242h  
(シリアルコミュニケーションインタフェースモードのとき (SCI0.SCMR.SMIF ビット = 0) )

ビット	シンボル	設定値	ビット名	機能	R/W
b1-b0	CKE[1:0]	00	クロック許可ビット	(調歩同期式の場合) 00: 内蔵ポーレートジェネレータ SCK0 端子は入出力ポート	R/W *1
b2	TEIE	0	送信完了割り込み許可ビット	0: TEI0 割り込み要求を禁止	R/W
b4	RE	0	受信許可ビット	0: シリアル受信動作を禁止	R/W *2
		1		1: シリアル受信動作を許可	
b5	TE	0	送信許可ビット	0: シリアル送信動作を禁止	R/W *2
		1		1: シリアル送信動作を許可	
b6	RIE	0	受信割り込み許可ビット	0: RXI0,ERI0 割り込み要求を禁止	R/W
		1		1: RXI0,ERI0 割り込み要求を許可	
b7	TIE	0	送信割り込み許可ビット	0: TXI0 割り込み要求を禁止	R/W
		1		1: TXI0 割り込み要求を許可	

【注】 \*1 TE ビット = 0、RE ビット = 0 の場合のみ書き込み可能です。

\*2 TE ビット = 0、RE ビット = 0 の場合のみ“1”を書き込み可能です。いったん、TE、RE ビットのいずれかを“1”に設定した後は、TE ビット = 0、RE ビット = 0 の書き込みのみ可能になります。

- SCI0 シリアルモードレジスタ (SCI0.SMR) ビット数 : 8 ビット アドレス : 0008 8240h  
(シリアルコミュニケーションインタフェースモードのとき (SCI0.SCMR.SMIF ビット = 0) )

ビット	シンボル	設定値	ビット名	機能	R/W
b1-b0	CKS[1:0]	00	クロック選択ビット	00: PCLK クロック (n=0) *1	R/W *2
b2	MP	0	マルチプロセッサモードビット	(調歩同期式モードのみ有効) 0: マルチプロセッサ通信機能を禁止	R/W *2
b3	STOP	0	ストップビット長選択ビット	(調歩同期式モードのみ有効) 0: 1 ストップビット	R/W *2
b5	PE	0	パリティ許可ビット	(調歩同期式モードのみ有効) ● 送信時 0: パリティビットなし ● 受信時 0: パリティビットなしで受信	R/W *2
b6	CHR	0	キャラクタ長ビット	(調歩同期式モードのみ有効) 0: データ長 8 ビットで送受信	R/W *2
b7	CM	0	コミュニケーションモードビット	0: 調歩同期式モードで動作	R/W *2

【注】 \*1 n の設定値については 7. 参考ドキュメント「ユーザズマニュアル」を参照してください。

\*2 SCI0.SCR.TE ビット = 0、SCI0.SCR.RE ビット = 0 (シリアル送信動作を禁止、かつシリアル受信動作を禁止) の場合のみ書き込み可能です。

シングルチップモードによる UART 経由での  
内蔵フラッシュメモリ書き換え (スレーブ)

RX62T グループ

- SCIO スマートカードモードレジスタ (SCIO.SCMR) ビット数 : 8 ビット アドレス : 0008 8246h

ビット	シンボル	設定値	ビット名	機能	R/W
b0	SMIF	0	スマートカードインタフェース モード選択ビット	0: シリアルコミュニケーション インタフェースモード	R/W *1
b3	SDIR	0	ビットオーダー選択ビット	0: LSB ファーストで送受信	R/W *1

【注】 \*1 SCIO.SCR.TE ビット = 0、SCIO.SCR.RE ビット = 0 (シリアル送信動作を禁止、かつシリアル受信動作を禁止) の場合のみ書き込み可能です。

- SCIO ビットレートレジスタ (SCIO.BRR) ビット数 : 8 ビット アドレス : 0008 8241h

ビット	シンボル	設定値	ビット名	機能	R/W
b7-b0	—	00110001 *1	—	31h: ビットレート = 31250 bps (PCLK = 50MHz 時)	R/W *2

【注】 \*1 BRR の設定値については、7. 参考ドキュメント「ユーザズマニュアル」を参照してください。

\*2 読み出しは常に可能ですが、書き込みは SCIO.SCR.TE ビット = 0、SCIO.SCR.RE ビット = 0 (シリアル送信動作を禁止、かつシリアル受信動作を禁止) の場合のみ可能です。

- SCIO シリアルステータスレジスタ (SCIO.SSR) ビット数 : 8 ビット アドレス : 0008 8244h  
(シリアルコミュニケーションインタフェースモードのとき (SCIO.SCMR.SMIF ビット = 0) )

ビット	シンボル	設定値	ビット名	機能	R/W
b2	TEND	—	送信完了フラグ	0: キャラクタ送信中 1: キャラクタ送信終了	R
b4	FER	— *1	フレーミングエラーフラグ	0: フレーミングエラーの発生 なし 1: フレーミングエラーの発生 あり	R/W *2
b5	ORER	— *1	オーバランエラーフラグ	0: オーバランエラーの発生 なし 1: オーバランエラーの発生 あり	R/W *2

【注】 \*1 本アプリケーションノートでは、FER ビットおよび ORER ビットは読み出しだけ行います。フラグをクリアするための“0”書き込みは行いません。

\*2 フラグをクリアするための“0”書き込みのみ可能です。

- SCIO トランスミットデータレジスタ (SCIO.TDR) ビット数 : 8 ビット アドレス : 0008 8243h

ビット	シンボル	設定値	ビット名	機能	R/W
b7-b0	—	— *1	—	送信データを格納	R/W

【注】 \*1 送信データを設定します。

- SCIO レシーブデータレジスタ (SCIO.RDR) ビット数 : 8 ビット アドレス : 0008 8245h

ビット	シンボル	設定値	ビット名	機能	R/W
b7-b0	—	—	—	受信データを格納	R

RX62T グループ

(5) 割り込みコントローラ (ICU)

- 割り込み要因プライオリティレジスタ 80 (IPR80) ビット数: 8 ビット アドレス: 0008 7380h

ビット	シンボル	設定値	ビット名	機能	R/W
b3-b0	IPR[3:0]	0000	SCI0 割り込み優先レベル設定 ビット	0000: レベル 0( 割り込み禁止 )	R/W

- 割り込み要求許可レジスタ 1A (IER1A) ビット数: 8 ビット アドレス: 0008 721Ah

ビット	シンボル	設定値	ビット名	機能	R/W
b7	IEN7	0	RX10 割り込み要求許可ビット 7	0: RX10 割り込み要求禁止	R/W

- 割り込み要求許可レジスタ 1B (IER1B) ビット数: 8 ビット アドレス: 0008 721Bh

ビット	シンボル	設定値	ビット名	機能	R/W
b0	IEN0	0	TX10 割り込み要求許可ビット 0	0: TX10 割り込み要求禁止	R/W

- 割り込み要求レジスタ 215 (IR215) ビット数: 8 ビット アドレス: 0008 70D7h

ビット	シンボル	設定値	ビット名	機能	R/W
b0	IR	0	RX10 割り込みステータスフラグ	0: RX10 割り込み要求なし 1: RX10 割り込み要求あり	R/W *1

【注】 \*1 フラグをクリアするための“0”書き込みのみ可能。“1”書き込みは禁止です。

- 割り込み要求レジスタ 216 (IR216) ビット数: 8 ビット アドレス: 0008 70D8h

ビット	シンボル	設定値	ビット名	機能	R/W
b0	IR	0	TX10 割り込みステータスフラグ	0: TX10 割り込み要求なし 1: TX10 割り込み要求あり	R/W *1

【注】 \*1 フラグをクリアするための“0”書き込みのみ可能。“1”書き込みは禁止です。

## 5.7 関数仕様

スレーブの各関数の仕様を以下に示します。

## (1) PowerON\_Reset\_PC 関数

## (a) 機能説明

PowerON\_Reset\_PC 関数は、スタックポインタの初期化( PowerON\_Reset\_PC 関数に対して #pragma entry を宣言することによりコンパイラが自動的に ISP/USP の初期化コードを関数先頭に生成 )、INTB の設定 ( set\_intb 関数：組み込み関数)、FPSW の初期化 ( set\_fpsw 関数：組み込み関数)、RAM 領域セクションの初期化 ( \_INITSCT 関数：標準ライブラリ関数)、HardwareSetup 関数の呼び出し、PSW の初期化 ( set\_psw 関数：組み込み関数)、プロセッサモードをユーザモードに設定します。その後、main 関数を呼び出します。

## (b) 引数

なし

## (c) 戻り値

なし

## (d) フローチャート

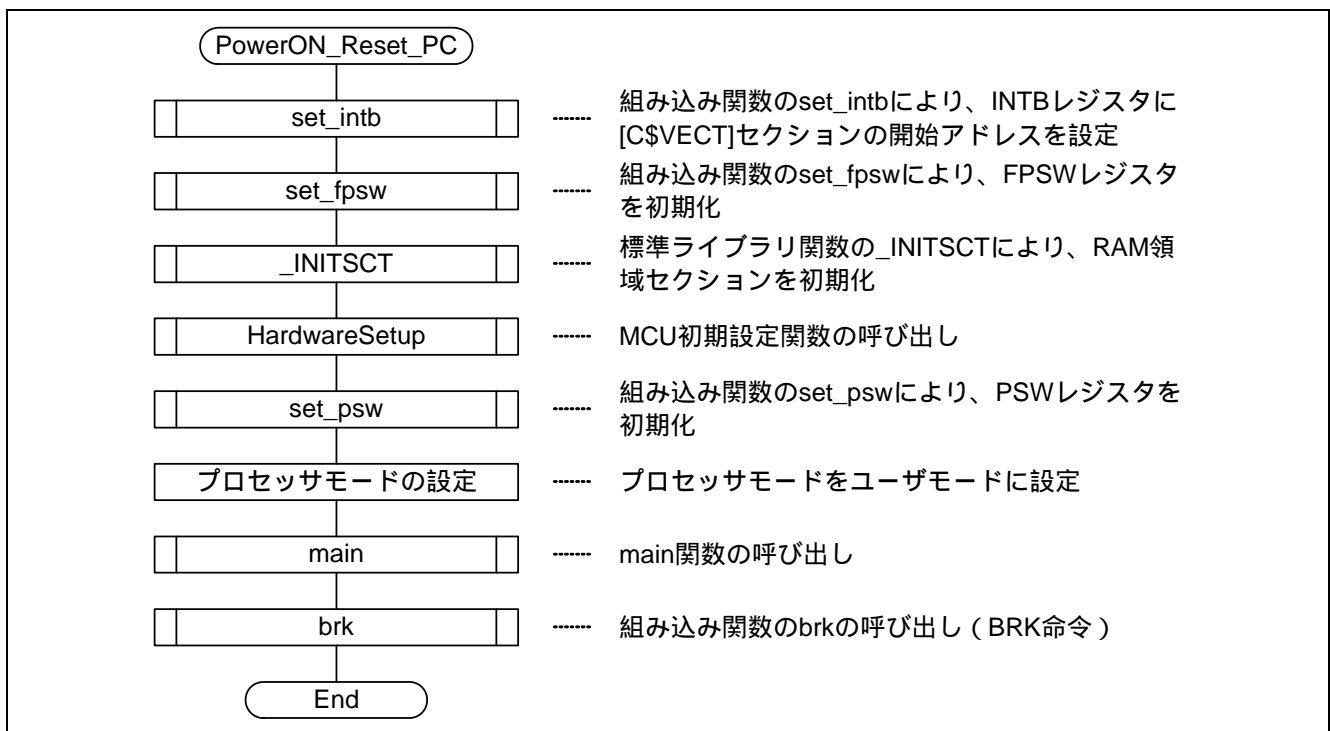


図11 フローチャート ( PowerON\_Reset\_PC ) ( スレーブ )



(2) HardwareSetup 関数

(a) 機能説明

HardwareSetup 関数は MCU の初期設定を行います。クロック (システムクロック (ICLK) および周辺モジュールクロック (PCLK)) の初期設定、LED0 ~ LED3 を接続している I/O ポート (P71、P72、P73、および P33) の初期出力設定、および SCI0 の初期設定を行います。

(b) 引数

なし

(c) 戻り値

なし

(d) フローチャート

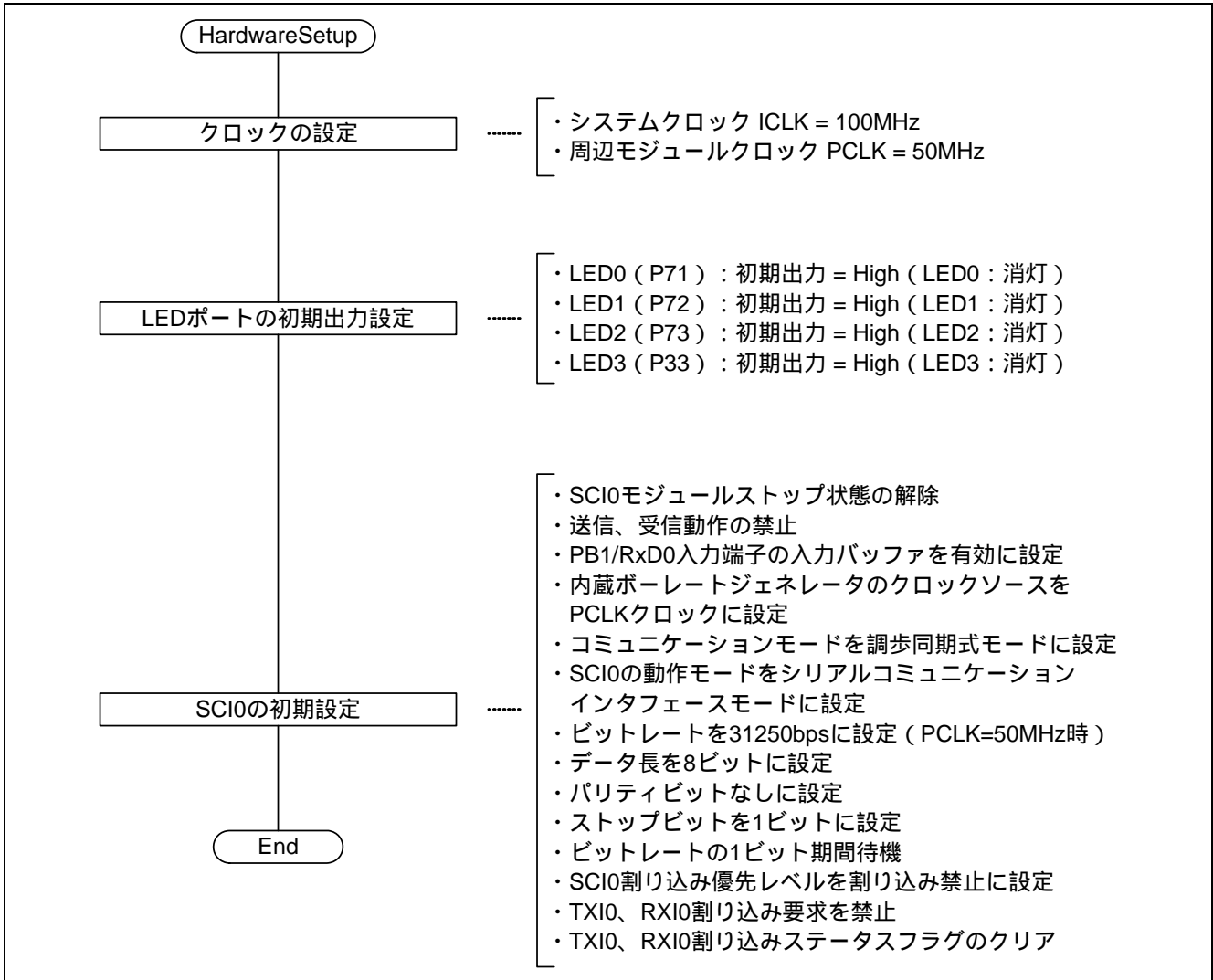


図12 フローチャート (HardwareSetup) (スレーブ)



(3) main 関数

(a) 機能説明

main 関数は、マスタからの 1 バイトデータの受信制御、エラー発生時に Indicate\_Error\_LED 関数の呼び出し、内蔵 RAM 上のユーザマツト書き込み / 制御プログラム (Flash\_Update 関数) の呼び出しを行います。

(b) 引数

なし

(c) 戻り値

なし

(d) フローチャート

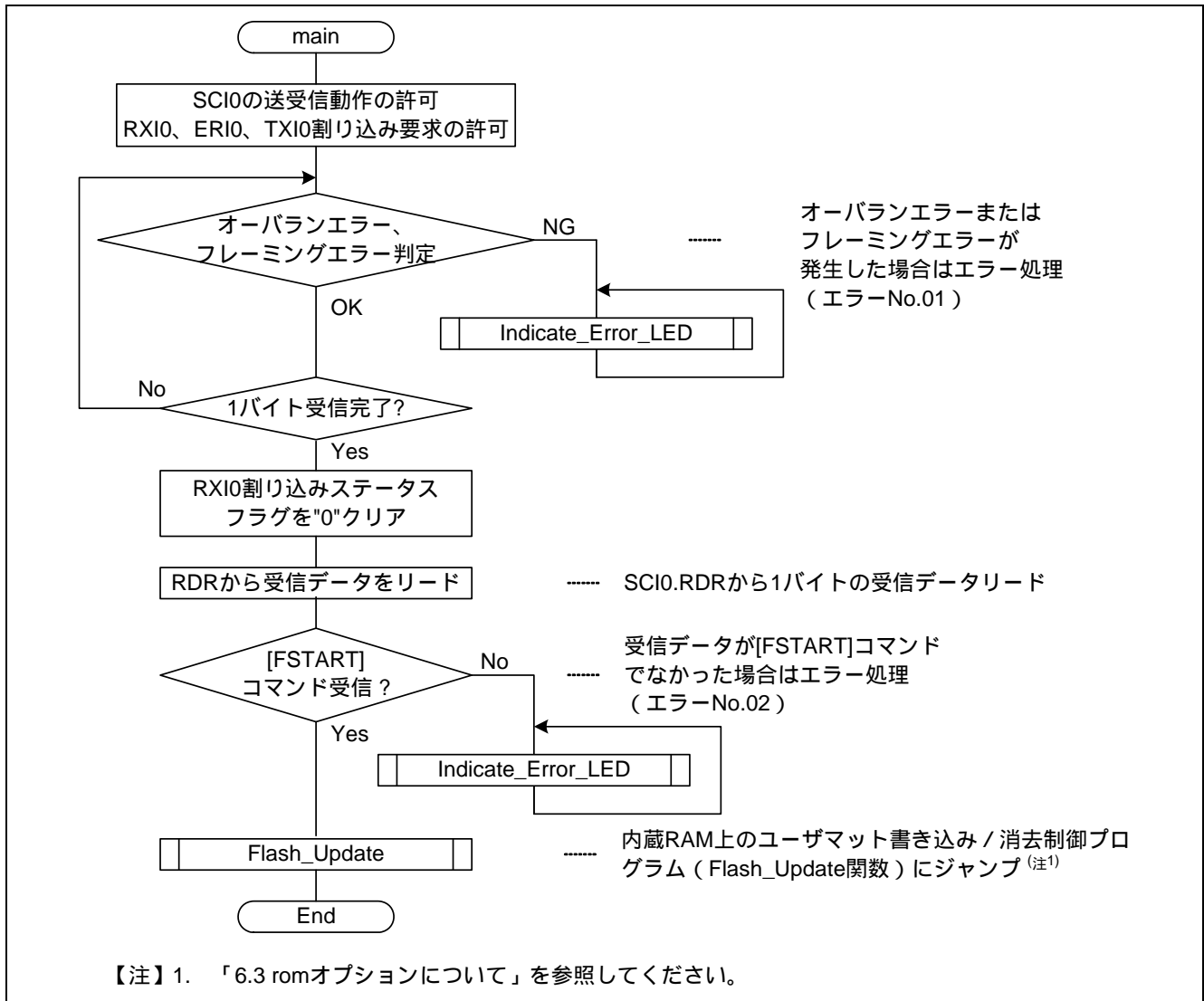


図13 フローチャート (main) (スレーブ)

(4) Flash\_Update 関数

(a) 機能説明

Flash\_Update 関数は、マスタとの調歩同期式シリアル通信による通信コマンドの受信制御、消去ブロック番号の受信制御、書き込みデータサイズの受信制御、書き込みデータの受信制御、[ACCEPTABLE] コマンドの送信制御、ユーザマットの消去時に R\_FlashErase 関数の呼び出し、ユーザマットの書き込み時に R\_FlashWrite 関数の呼び出し、ユーザマットの書き込み / 消去の正常終了時に Indicate\_Ending\_LED 関数の呼び出し、エラー終了時に Indicate\_Error\_LED 関数の呼び出しを行います。

(b) 引数

なし

(c) 戻り値

なし

(d) フローチャート

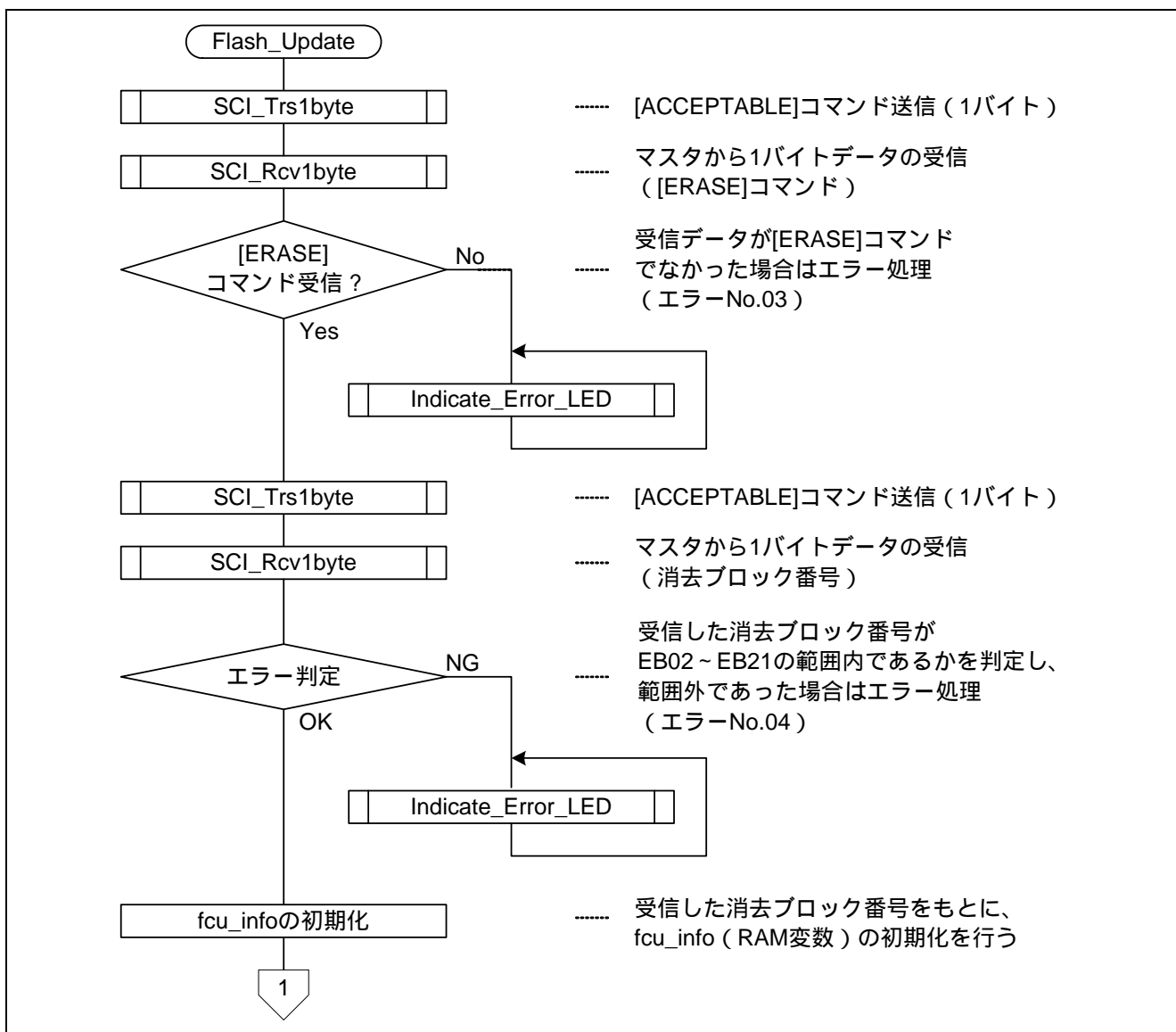


図14 フローチャート (Flash\_Update) (1) (スレーブ)

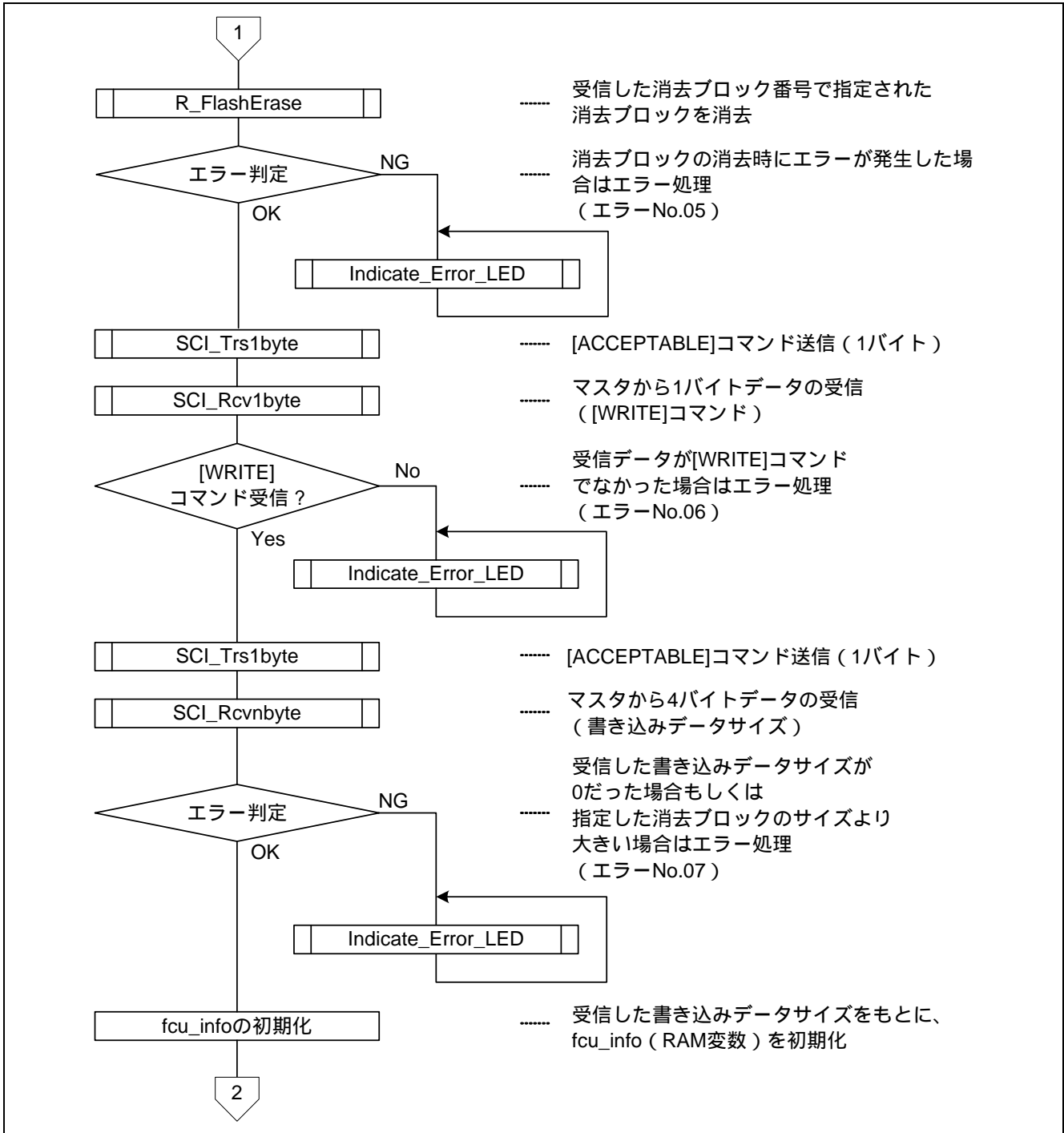


図15 フローチャート (Flash\_Update) (2) (スレーブ)

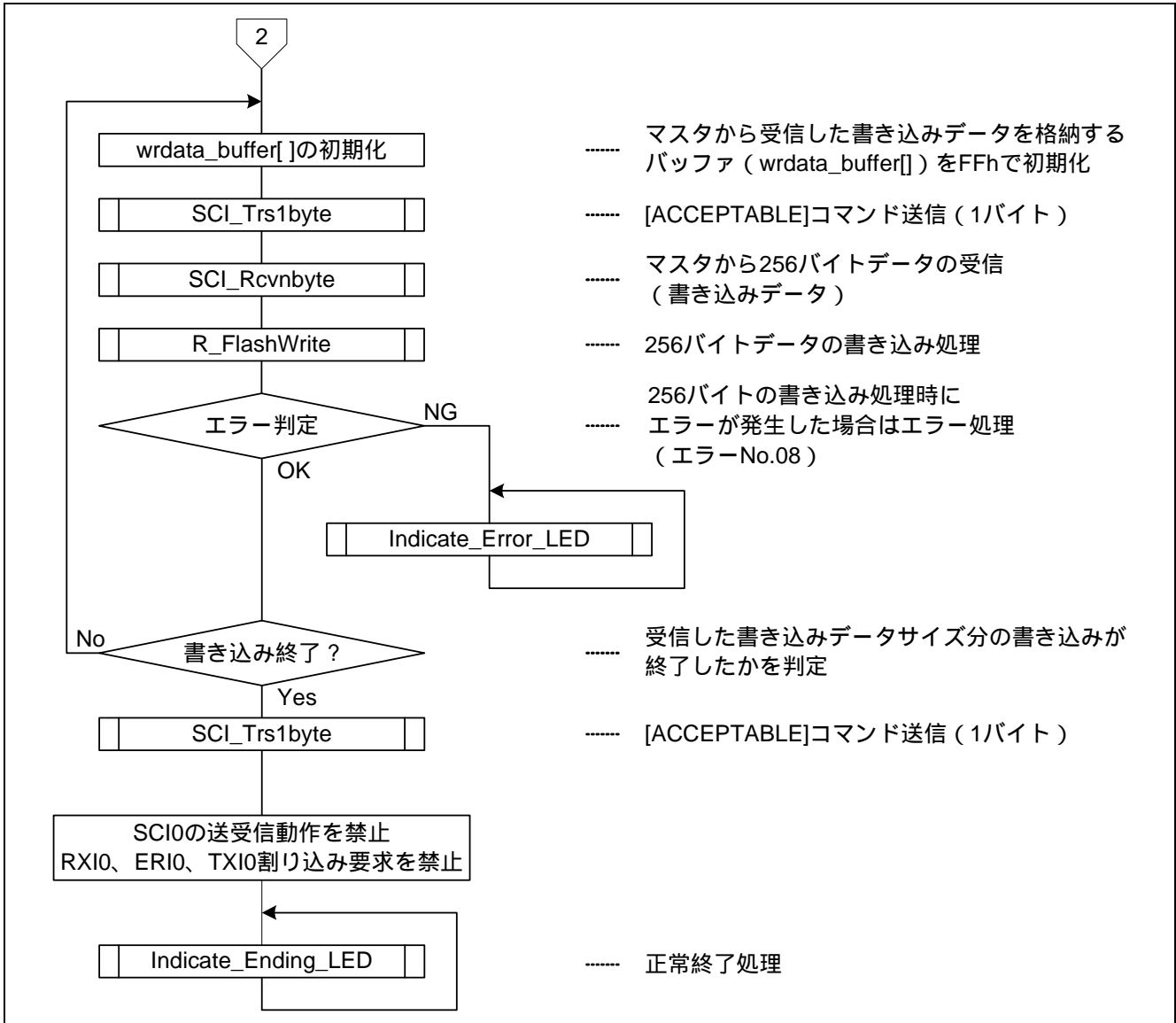


図16 フローチャート (Flash\_Update) (3) (スレーブ)

(5) Indicate\_Ending\_LED 関数

(a) 機能説明

Indicate\_Ending\_LED 関数は、書き込み / 消去が正常に終了した場合に、LED0 ~ LED3 に正常終了を示す表示を行います。LED0 ~ LED3 を順番に 1 つずつ点灯させます。

(b) 引数

なし

(c) 戻り値

なし

(d) フローチャート

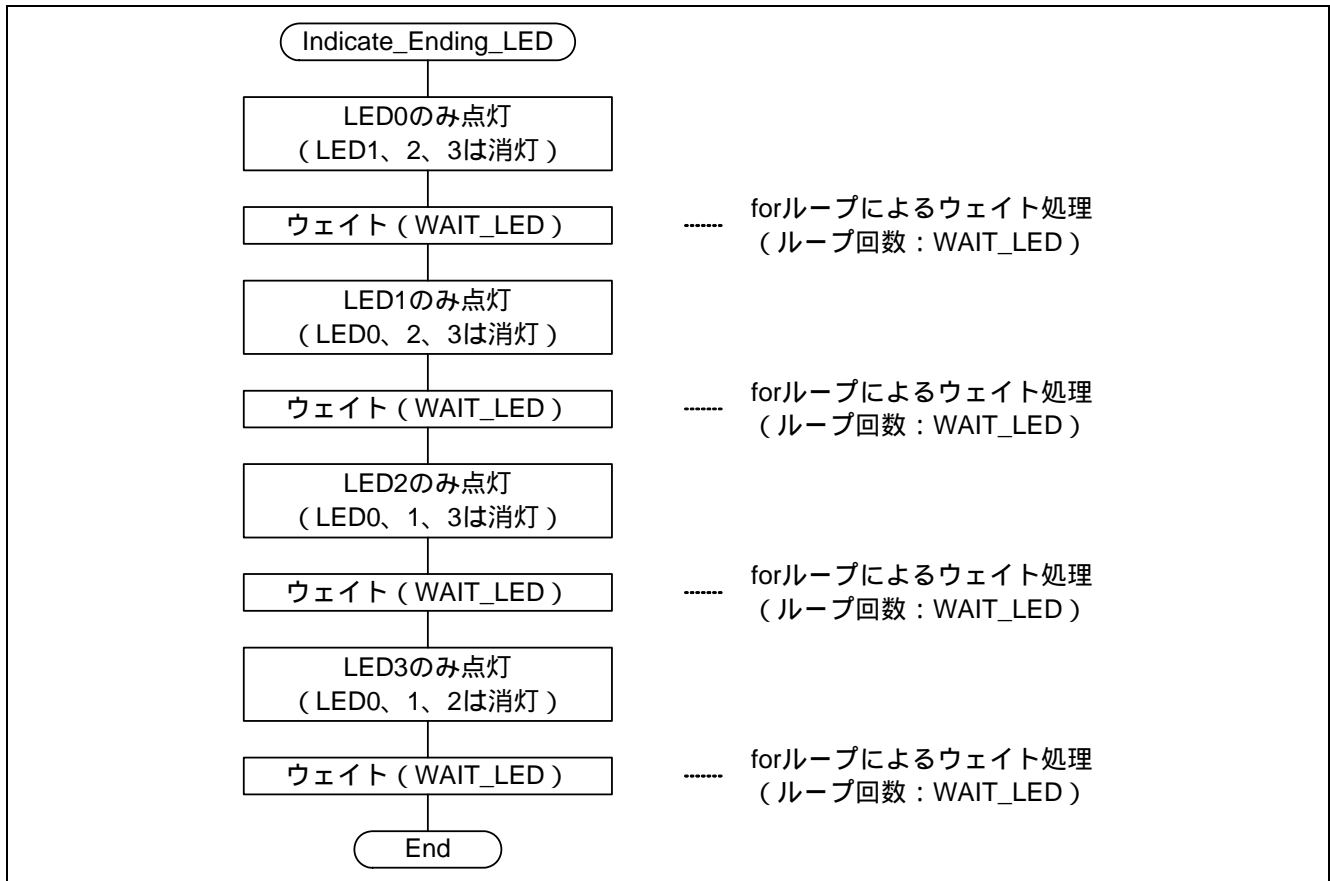


図17 フローチャート (Indicate\_Ending\_LED) (スレーブ)

(6) Indicate\_Error\_LED 関数

(a) 機能説明

Indicate\_Error\_LED 関数は、ユーザマットの書き込み / 消去の処理中にエラーが発生した場合に、LED0 ~ LED3 に発生したエラー番号の表示を行います。表示はエラー番号の表示と全消灯を繰り返します。

(b) 引数

表 16 に本関数で使用する引数を示します。

表16 Indicate\_Error\_LED 関数の引数一覧

引数	型	説明
第 1 引数	unsigned char	ユーザマットの書き込み / 消去中に発生したエラー番号* <sup>1</sup>

【注】 \*1 エラー番号は、「4.8 エラー処理」を参照してください。

(c) 戻り値

なし

(d) フローチャート

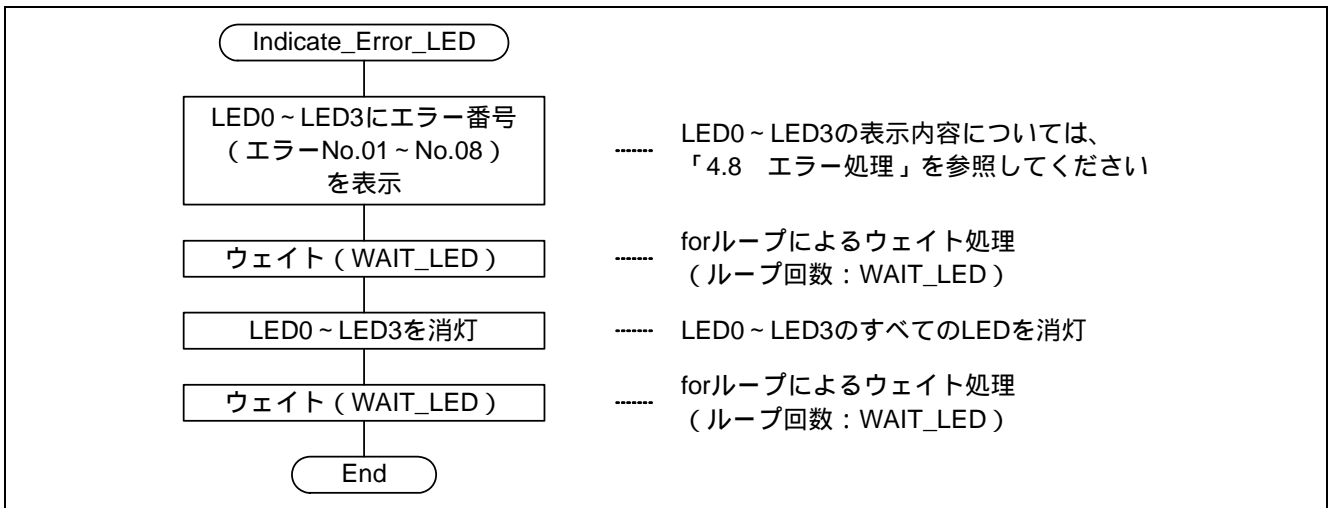


図18 フローチャート (Indicate\_Error\_LED) (スレーブ)

(7) SCI\_Rcv1byte 関数

(a) 機能説明

SCI\_Rcv1byte 関数は、SCI0 の調歩同期式シリアル通信による 1 バイトデータの受信制御を行います。

(b) 引数

なし

(c) 戻り値

表 17 に本関数で使用する戻り値を示します。

表17 SCI\_Rcv1byte 関数の戻り値一覧

型	説明
unsigned char	SCI0 の調歩同期式シリアル通信による 1 バイト受信データ

(d) フローチャート

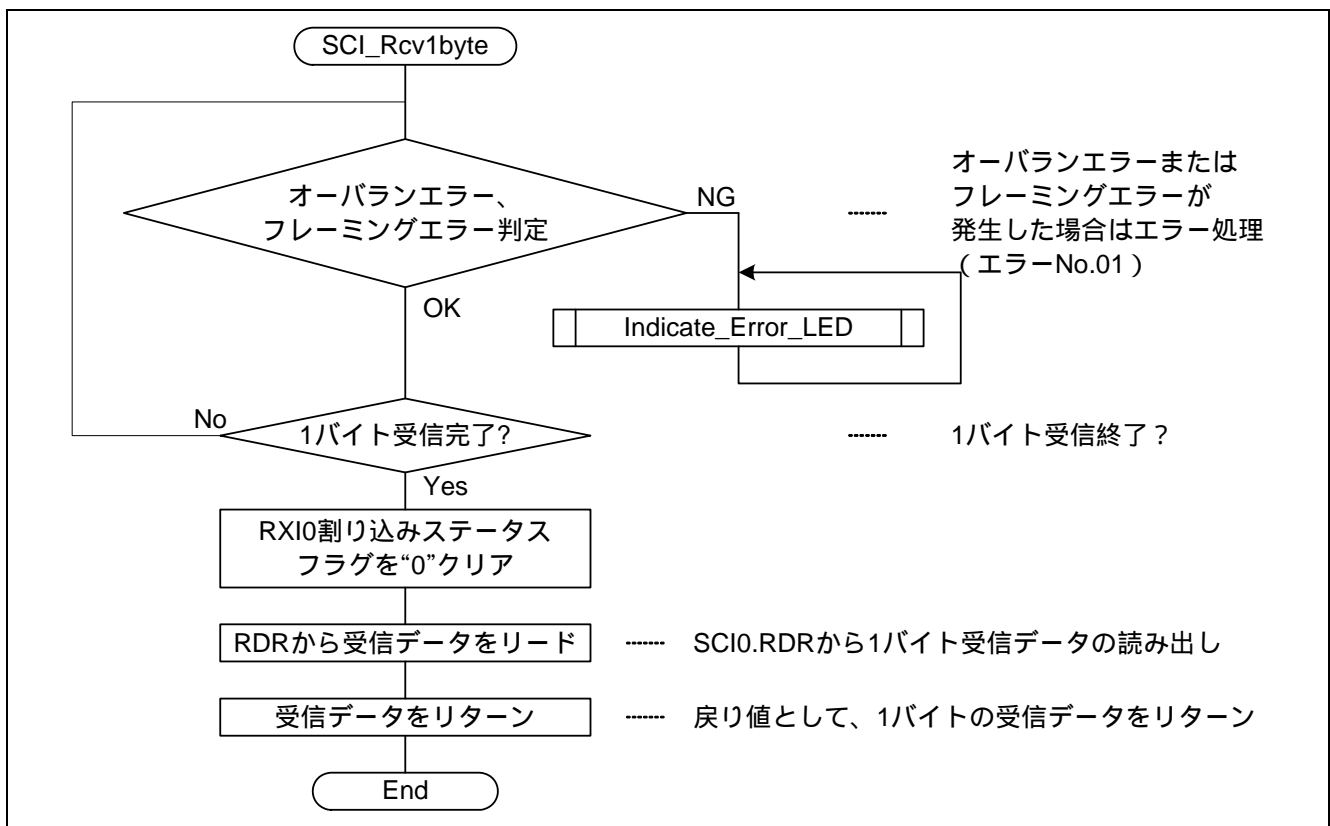


図19 フローチャート (SCI\_Rcv1byte) (スレーブ)

(8) SCI\_Rcvnbyte 関数

(a) 機能説明

SCI\_Rcvnbyte 関数は、SCI0 の調歩同期式シリアル通信による n バイトデータ (n は unsigned short 型の第 1 引数) の受信制御を行います。

(b) 引数

表 18 に本関数で使用する引数を示します。

表18 SCI\_Rcvnbyte 関数の引数一覧

引数	型	説明
第 1 引数	unsigned short	SCI0 の調歩同期式シリアル通信による受信データバイト数
第 2 引数	unsigned char*	受信データ格納場所の先頭アドレス

(c) 戻り値

なし

(d) フローチャート

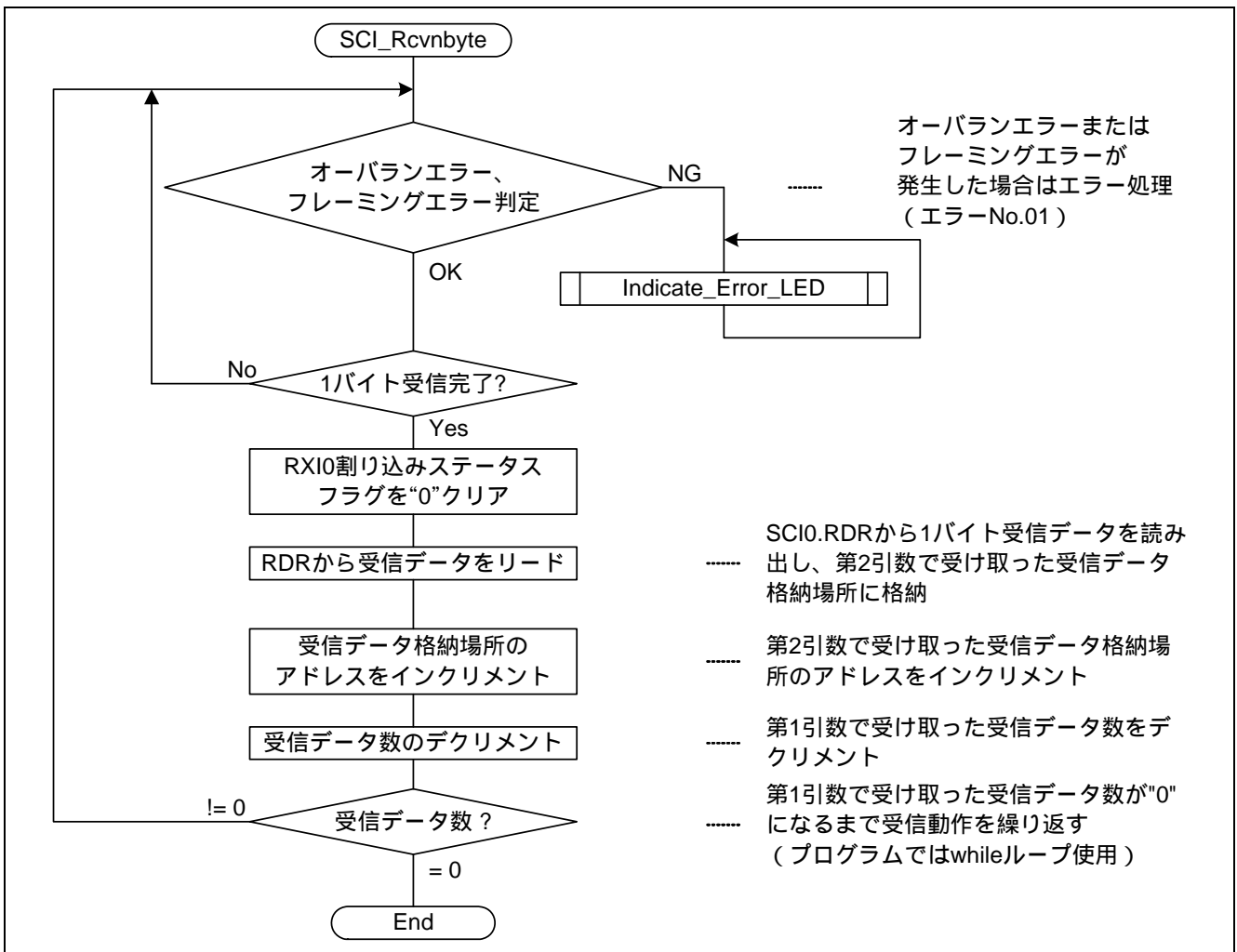


図20 フローチャート (SCI\_Rcvnbyte) (スレーブ)



(9) SCI\_Trns1byte 関数

(a) 機能説明

SCI\_Trns1byte 関数は、SCI0 の調歩同期式シリアル通信による 1 バイトデータの送信制御を行います。

(b) 引数

表 19 に本関数で使用する引数を示します。

表19 SCI\_Trns1byte 関数の引数一覧

引数	型	説明
第 1 引数	unsigned char	SCI0 の調歩同期式シリアル通信による 1 バイト送信データ

(c) 戻り値

なし

(d) フローチャート

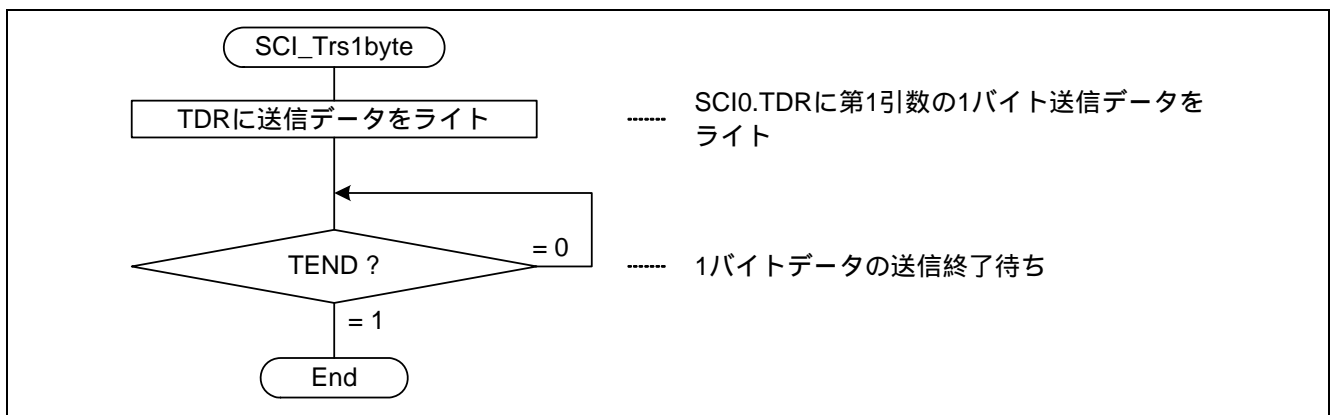


図21 フローチャート ( SCI\_Trns1byte ) ( スレーブ )

## 6. 使用上の注意事項

### 6.1 SCI0 初期化時のビットレートに対する 1 ビット期間の待機時間について

本アプリケーションノートでは、SCI 初期化時のビットレートレジスタ (SCI0.BRR) 設定後のビットレートに対する 1 ビット期間の待機時間はソフトウェアタイマを使用して時間を計測しています。SCI0 の調歩同期式シリアル通信のビットレートは 31250 bps に設定しているため、

$$\text{ビットレート } 31250 \text{ bps に対する } 1 \text{ ビット期間} = 32[\mu\text{s}]$$

と計算できます。

本アプリケーションノートでは、ビットレートに対する 1 ビット期間の待機時間は、記号定数 WAIT\_SCI1BIT で定義したループ回数を while ループで回す処理を行っています。while ループを 1 回まわるサイクル数を 5 サイクル\*1 (コンパイラが出力するアセンブリ言語で確認できます) とすると、

$$\text{while ループの回る回数} = \text{待機時間} / (\text{while ループを 1 回まわるサイクル数} \times \text{ICLK サイクル時間})$$

で計算できます。なお、CPU の命令処理時間はパイプライン処理によって変動するため、上記の while ループを 1 回まわるサイクル数 (5 サイクル) は概算の命令処理時間となります。

本アプリケーションノートでは、マージンをとって待機時間を 96[μs]として計算しているため、

$$\text{while ループの回る回数} = \text{WAIT\_SCI1BIT} = 96[\mu\text{s}] / (5 \times 10 [\text{ns}]) = 1920 \text{ (ICLK} = 100\text{MHz 時)}$$

となります。したがって、WAIT\_SCI1BIT を 1920 と定義して使用しています。

ユーザにて本アプリケーションノートをご使用の際には、CPU の命令処理時間の評価を十分に行ってください。タイマを使用して時間を計測するようにしてください。

【注】 \*1 サイクル数は参考値です。顧客の条件によって変わります。

### 6.2 消去ブロック EB00 の書き換え時の注意事項

消去ブロック EB00(書き込み/消去用アドレス:00FF F000h~00FF FFFF、読み出し用アドレス:FFFF F000h~FFFF FFFFh)には、固定ベクタ (FFFF FF80h~FFFF FFFFh)、ID コードプロテクト (FFFF FFA0h~FFFF FFAFh) などが配置されています。

EB00 の書き込み/消去を行うと、上記の固定ベクタ、ID コードプロテクトは、一度消去されてしまいます。したがって、EB00 の消去後に再度、固定ベクタ、ID コードプロテクトの設定を行う必要がありますのでご注意ください。

ID コードプロテクトは、ホストからの読み出し/書き込み/消去を禁止するための機能で、ROM 上に書かれている制御コードおよび ID コードを使い、ID コードプロテクトの判定を行います。ID コードプロテクトの詳細については、7. 参考ドキュメント「ユーザーズマニュアル」を参照してください。

### 6.3 rom オプションについて

本アプリケーションノートでは、最適化リンケージエディタの rom オプションを使用して、PFRAM (ROM セクション) 内の定義シンボルを RPFRAM (RAM セクション) 内のアドレスにリロケーションしています。これにより、Flash\_Update 関数コール後の実行アドレスはリロケーション先である RAM 上のアドレスとなります。

rom オプションについての詳細は、7. 参考ドキュメント「RX ファミリー用 C/C++コンパイラパッケージ ユーザーズマニュアル」を参照してください。

## 7. 参考ドキュメント

- ユーザーズマニュアル  
RX62T グループ ユーザーズマニュアル ハードウェア編 Rev.1.00  
(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。)
- RX ファミリー ユーザーズマニュアル ソフトウェア編 Rev.1.00  
(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。)
- 開発環境マニュアル  
RX ファミリー用 C/C++コンパイラパッケージ ユーザーズマニュアル V.1.0.1.0  
(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。)
- アプリケーションノート  
RX62T グループ  
シングルチップモードによる UART 経由での内蔵フラッシュメモリ書き換え (マスタ) (R01AN0639JJ)  
(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。)
- アプリケーションノート  
RX600 シリーズ  
RX600 用のシンプルフラッシュ API Rev.2.20 (R01AN0544JU)  
(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  
(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/>

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。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2011.06.20	—	初版発行
1.01	2012.03.27	1 4 11  14 15 16 37 -	要旨 R01AN0639JJ のバージョンを削除、および R01AN0544JU0220 にバージョンを変更 表 1 スレーブの動作確認環境 評価環境を更新 4.5.3 シンプルフラッシュ API の変更点「RX600 用のシンプルフラッシュ API」の変更を反映 4.5.4 割り込みを使用する場合の注意点 追加 表 10 スレーブのセクション設定 設定内容を変更 表 11 スレーブのファイル構成「RX600 用のシンプルフラッシュ API」の説明を変更 表 12 スレーブの関数一覧「RX600 用のシンプルフラッシュ API」の変更を反映 7.参考ドキュメント「RX600 用のシンプルフラッシュ API」のバージョンを変更、および R01AN0639JJ のバージョンを削除 HEW workspace を表 1 の環境で生成
1.02	2015.03.04	4 4 25 34 34	表 1 最適化リンケージエディタ 追加 表 1 注 1 追加 図 13 注 1 追加 6.1 注 1 追加 6.3 rom オプションについて 追加

## 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. リザーブアドレスのアクセス禁止

【注意】リザーブアドレス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、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違くと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。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。

## 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。
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も、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て、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用用途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/>