

RX63T グループ

R01AN1422JJ0100

Rev.1.00

ユーザブートモードによる内蔵フラッシュメモリ書き換え (スレーブ)

2014.02.20

要旨

本アプリケーションノートは、RX63T グループ アプリケーションノート「ユーザブートモードによる内蔵フラッシュメモリ書き換え (マスタ)」(R01AN1421JJ) からクロック同期式シリアル通信で送信される、消去す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 (ユーザ領域) の書き込み/消去する処理について説明しています。

クロック同期式シリアル通信で消去す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送信する処理に関しては、RX63T グループ アプリケーションノート「ユーザブートモードによる内蔵フラッシュメモリ書き換え (マスタ)」(R01AN1421JJ) をご参考ください。

なお、本アプリケーションノートでは以下のアプリケーションノートのサンプルコードを使用しています。

RX63T の初期設定 :

「RX63T グループ 初期設定例」 Rev.1.00 (R01AN1252JJ0100)

内蔵フラッシュメモリの消去 / 書き込み :

「RX600 & RX200 シリーズ RX 用のシンプルフラッシュ API」 Rev.2.40 (R01AN0544JU0240)

対象デバイス

RX63T グループ 144 ピン、120 ピン、112 ピン、100 ピン

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	5
3. 関連アプリケーションノート	5
4. ハードウェア説明	6
5. ソフトウェア説明	8
6. 注意事項	34
7. サンプルコード	36
8. 参考ドキュメント	36

1. 仕様

本アプリケーションノートでは、RX63T グループの R5F563TEADFB を用いて、ユーザブートモードでユーザ領域の書き込み/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、書き込みデータサイズ、および書き込みデータをクロック同期式シリアル通信で受信し、ユーザ領域の書き込み/消去を行います。

マスタとスレーブ間のクロック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。

クロック同期式シリアル通信仕様は、以下の設定として、マスタから転送クロックを出力されます。

ビットレート 2.4Mbps

データビット 8 ビット

LSB ファースト

本アプリケーションノートでは、スレーブの通信準備が完了したときに PF2 端子の出力レベルを Low に切り替え、マスタに通信準備が完了したことを通知します。マスタ側でスレーブの通信準備完了を監視している端子と接続してください。

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 (BLOCK_30: 16K バイト) の消去を行い、受信した 8K バイト (128 バイト ×64) の書き込みデータを消去ブロック BLOCK_30 の先頭アドレスから書き込みます

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 I/O ポート (P11) を利用してピジー状態にアサート (Low)、ピジー解除時にネゲート (High) を出力します。マスタはスレーブからの出力を外部割り込み端子 (IRQ1-DS) で受け、立ち上がりエッジが入力されることにより次の送信を開始します。

スレーブは、正常にユーザ領域の消去/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書き込み/消去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。

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
ROM (コード格納用フラッシュメモリ)	ROM P/E モードによる 内蔵フラッシュメモリの書き換え
シリアルコミュニケーションインタフェース (ch0)	マスタとの通信用のクロック同期式シリアル

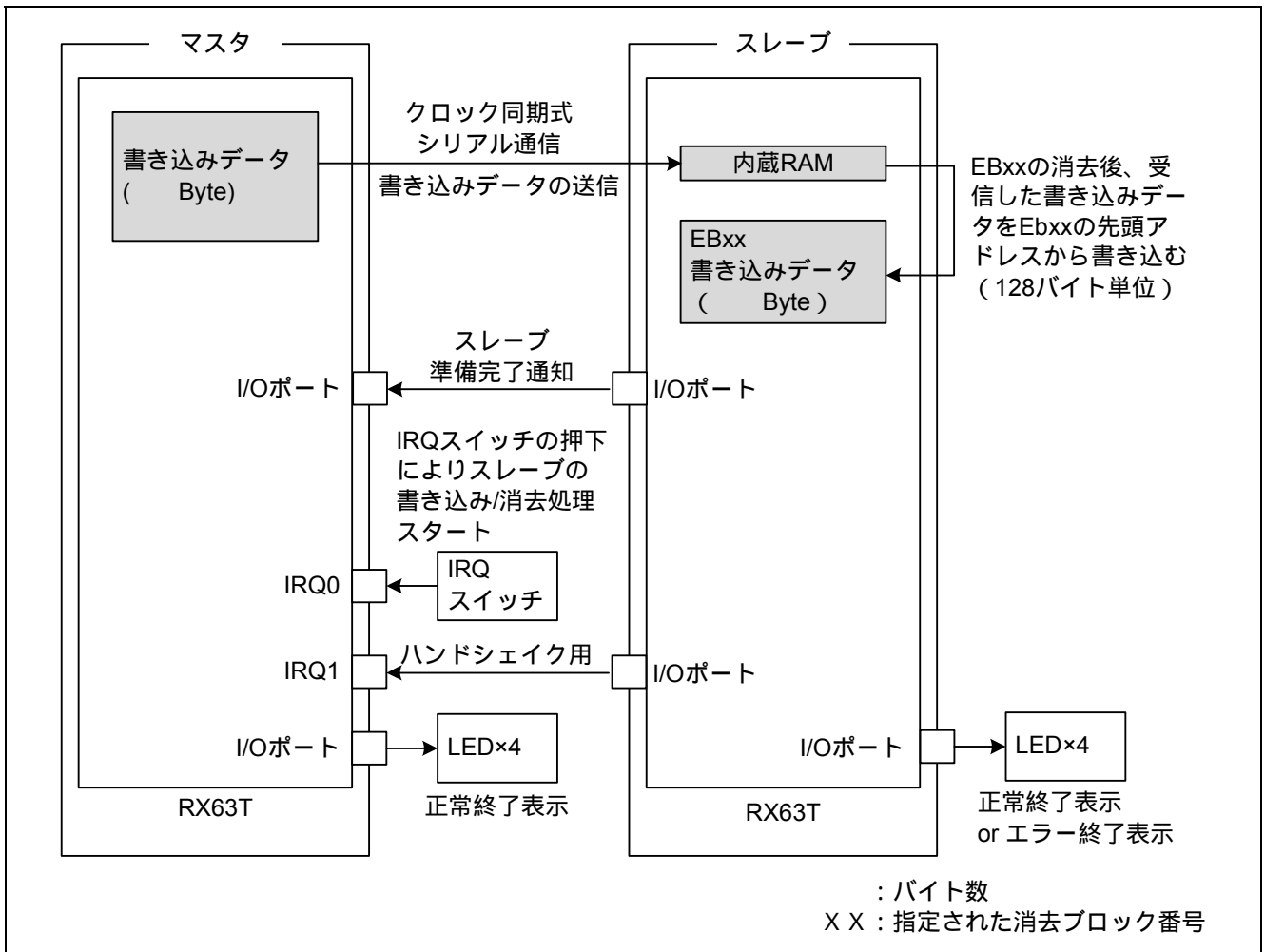


図1.1 使用例

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	RX63T グループ: R5F563TEADFB
動作周波数	メインクロック : 12.0MHz 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 (PLL2 分周) 周辺モジュールクロック B (PCLKB) : 48MHz (PLL4 分周) FlashIF クロック (FCLK) : 48MHz (PLL4 分周)
動作電圧	5.0V
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded WorkshopVersion 4.09.01.007
C コンパイラ	ルネサスエレクトロニクス製 RX Standard Toolchain (V.1.2.1.0) コンパイルオプション (統合開発環境のデフォルト設定を使用しています。)
iodefine.h のバージョン	2.00
エンディアン	リトルエンディアン
動作モード	ユーザブートモード
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 RX63T-H (R0K5563THS000BE)

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。

「RX63T グループ 初期設定例」Rev.1.00 (R01AN1252JJ0100)

「RX600&RX200 シリーズ RX 用のシンプルフラッシュ API」Rev.2.40 (R01AN0544JU0240)

4. ハードウェア説明

4.1 ハードウェア構成例

図 4.1 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。

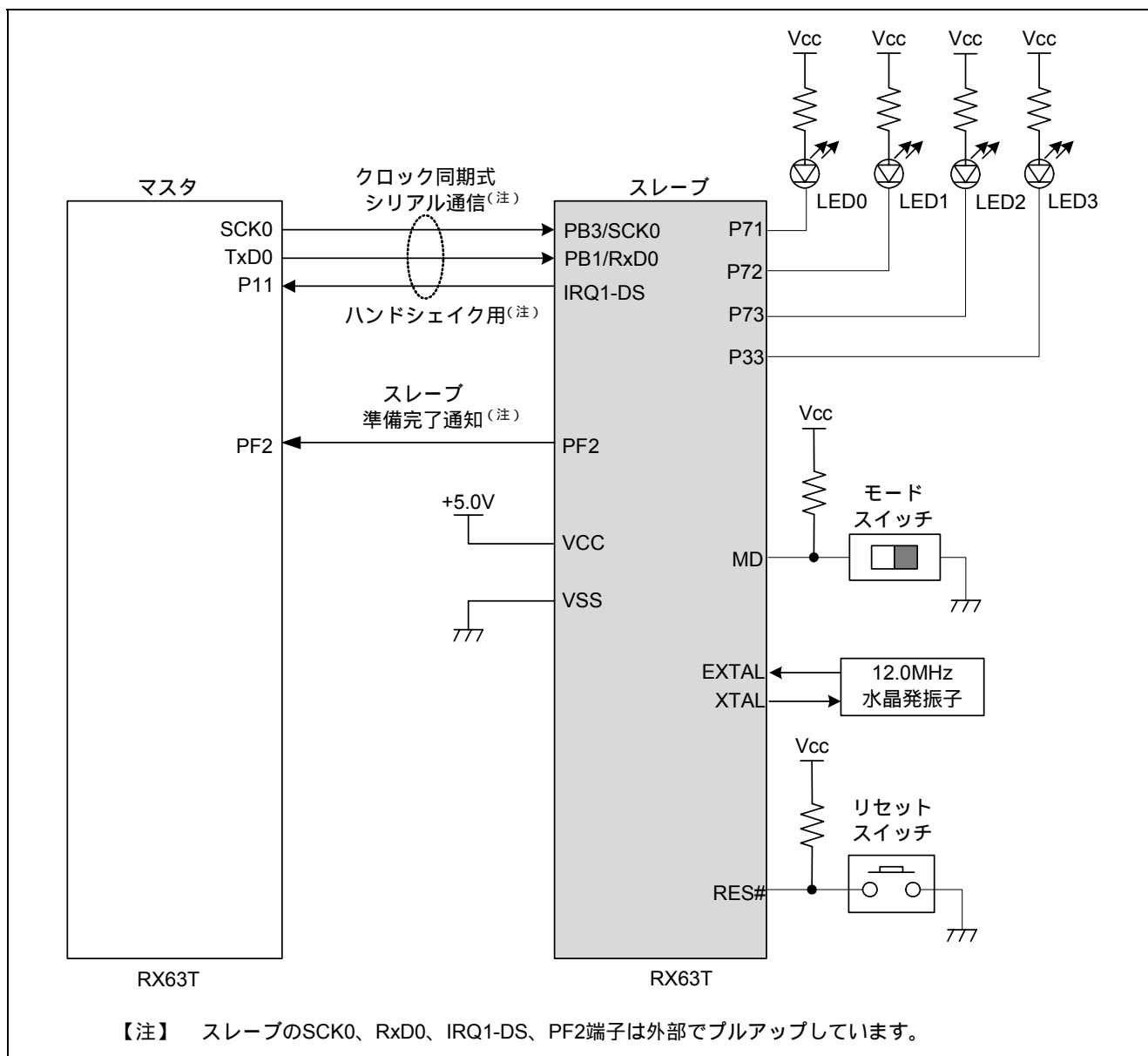


図4.1 スレーブのハードウェア構成図

4.2 使用端子一覧

表 4.1に使用端子と機能を示します。

表4.1 使用端子と機能

端子名	入出力	内容
PB1/RXD0	出力	シリアル受信端子
PB3/SCK0	出力	クロック端子
P11/IRQ1-DS	入力	ハンドシェイク用の Busy 端子
PF2/IRQ5	入力	スレーブ準備完了の確認用端子
P71	出力	LED0 接続端子（"High"出力：消灯、"Low"出力：点灯）
P72	出力	LED1 接続端子（"High"出力：消灯、"Low"出力：点灯）
P73	出力	LED2 接続端子（"High"出力：消灯、"Low"出力：点灯）
P33	出力	LED3 接続端子（"High"出力：消灯、"Low"出力：点灯）

5. ソフトウェア説明

5.1 動作概要

5.1.1 クロック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クロック同期式シリアル通信により、通信コマンド、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。転送クロックはマスタが出力します。使用する SCI0 の SCK0、RxD0 端子はそれぞれ外部でプルアップしています。

表 5.1 にクロック同期式シリアル通信仕様を示します。

表5.1 クロック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	クロック同期式モード
ビットレート	2.4Mbps (PCLKB = 48MHz 時)
データ転送方向	LSB ファースト
エラー	オーバランエラー

5.1.2 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 5.2 に示します。

表5.2 通信コマンド仕様

コマンド	値	説明	通信方向
FSTART	10h	スレーブのユーザ領域の書き込み/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	スレーブのユーザ領域の消去を開始するためのコマンド	マスタ → スレーブ
WRITE	12h	スレーブのユーザ領域の書き込みを開始するためのコマンド	マスタ → スレーブ

5.1.3 通信フロー

マスタとスレーブ間の通信フローを図 5.1 ~ 図 5.4 に示します。

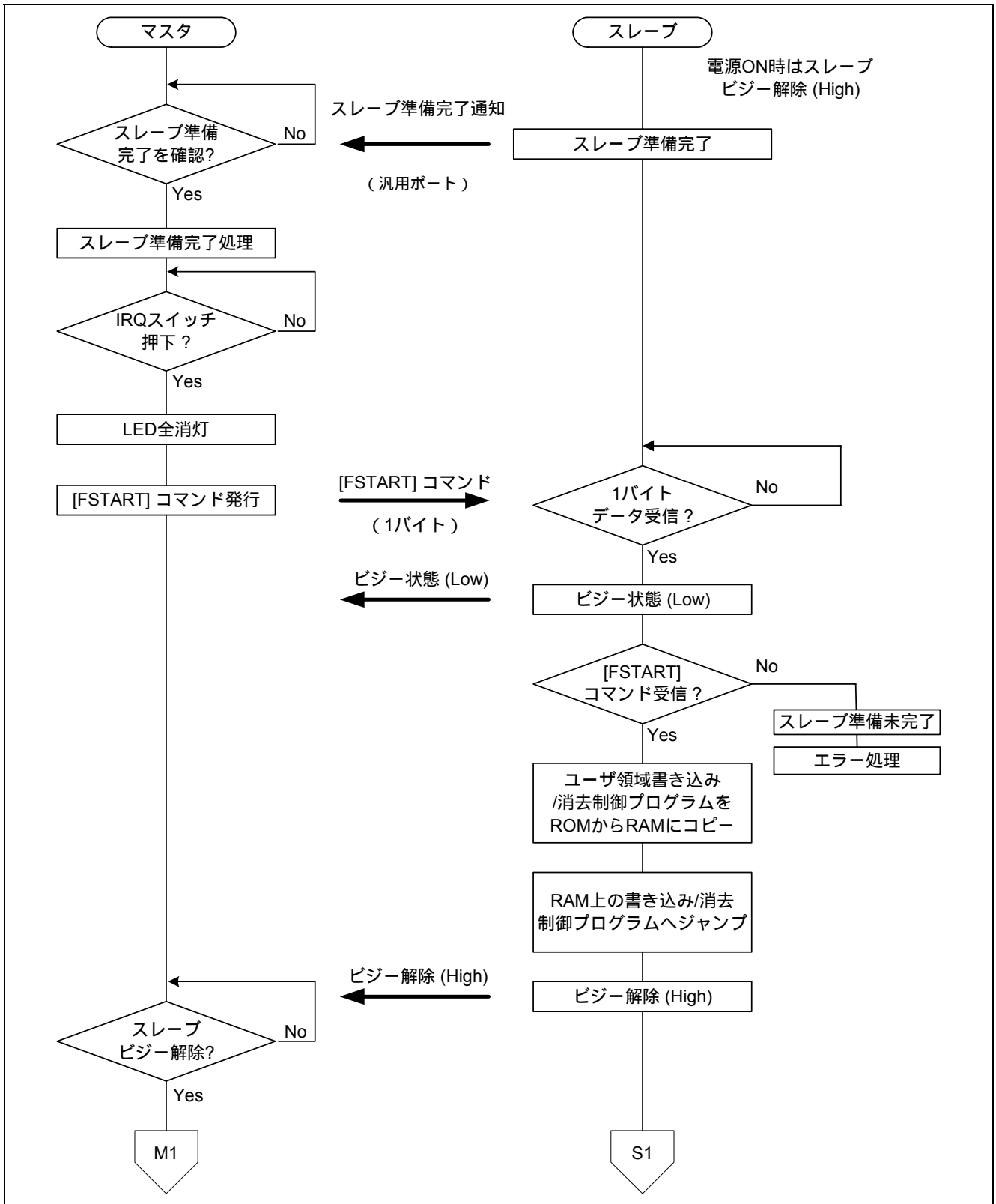


図5.1 通信フロー (1)

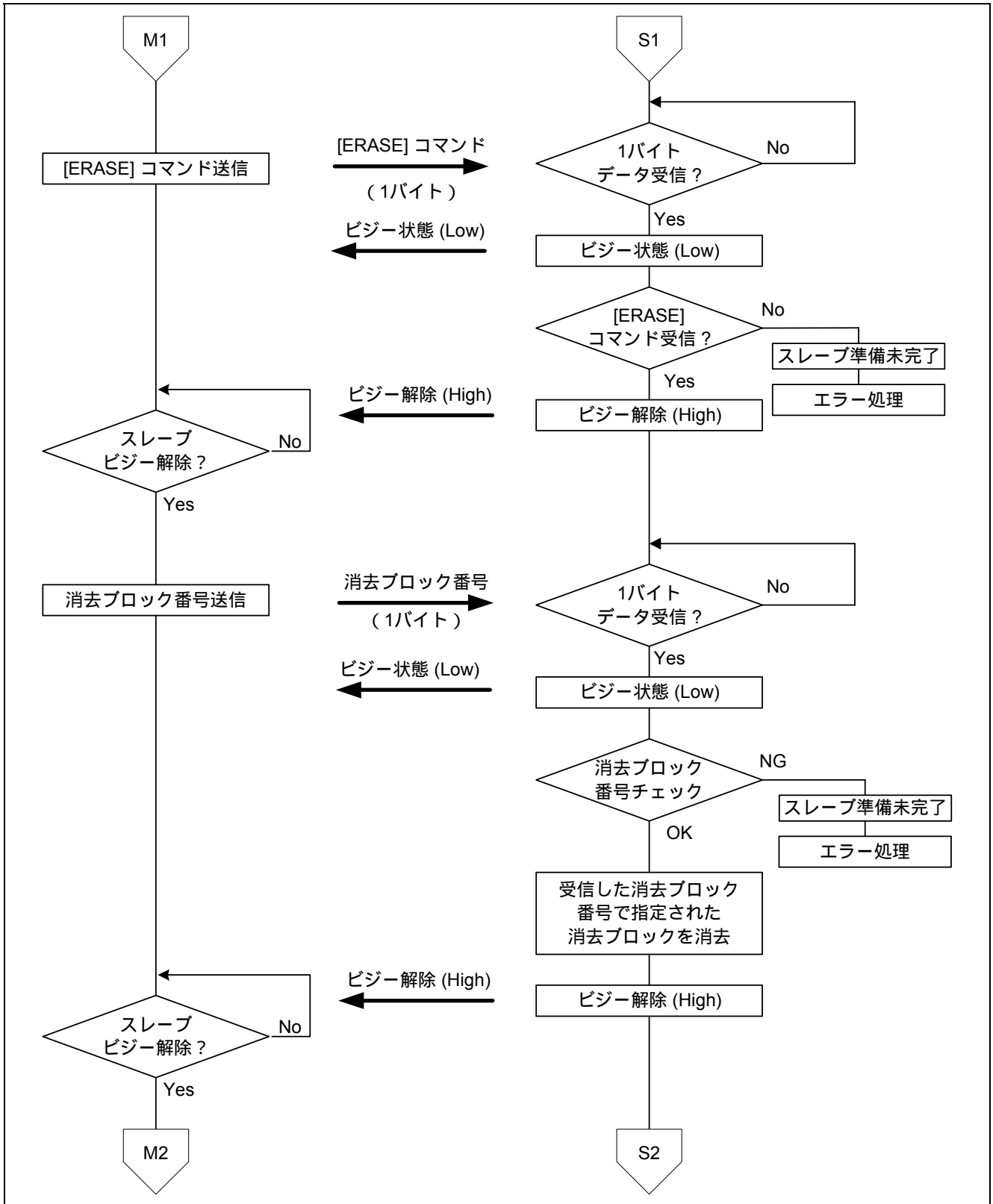


図5.2 通信フロー (2)

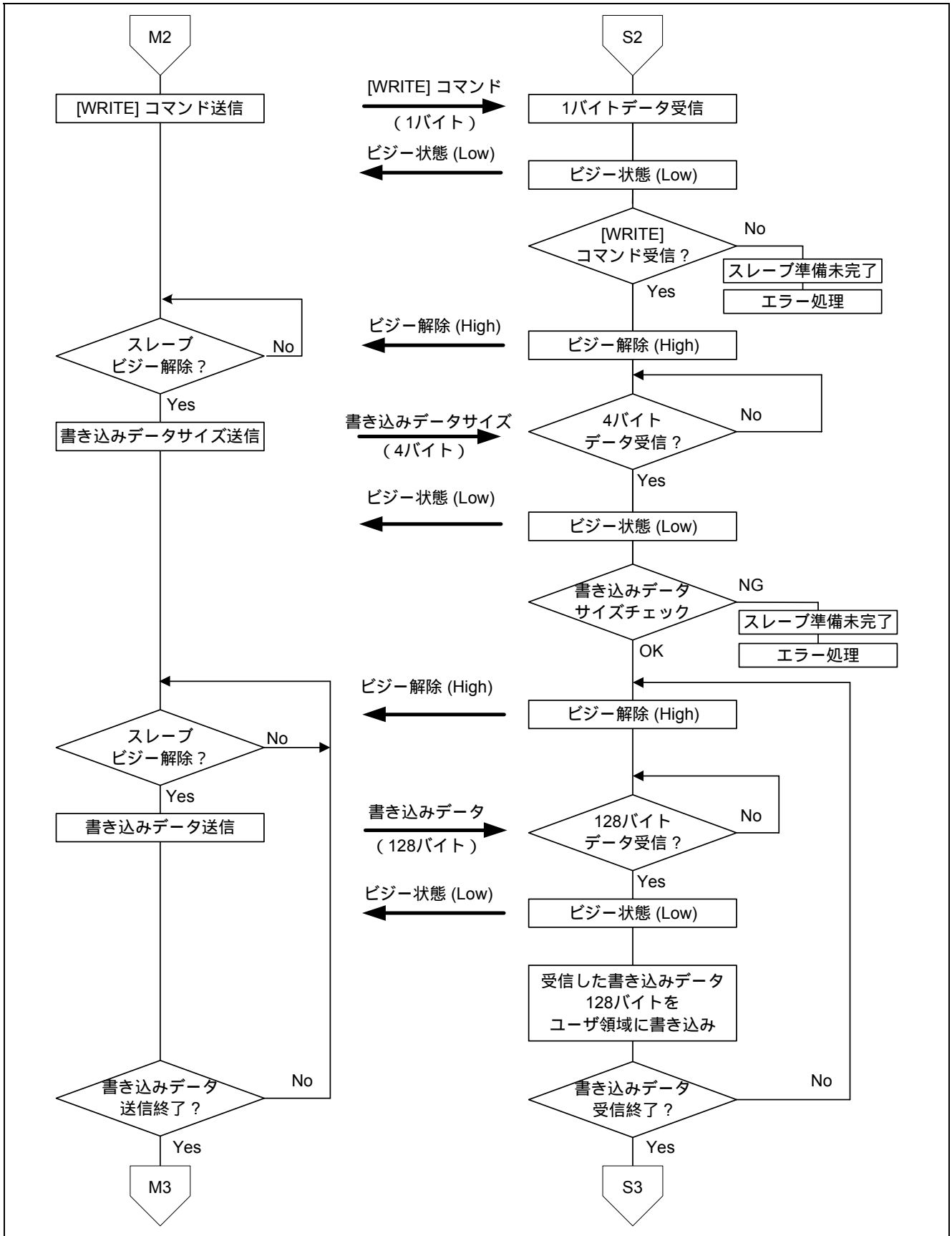


図5.3 通信フロー (3)

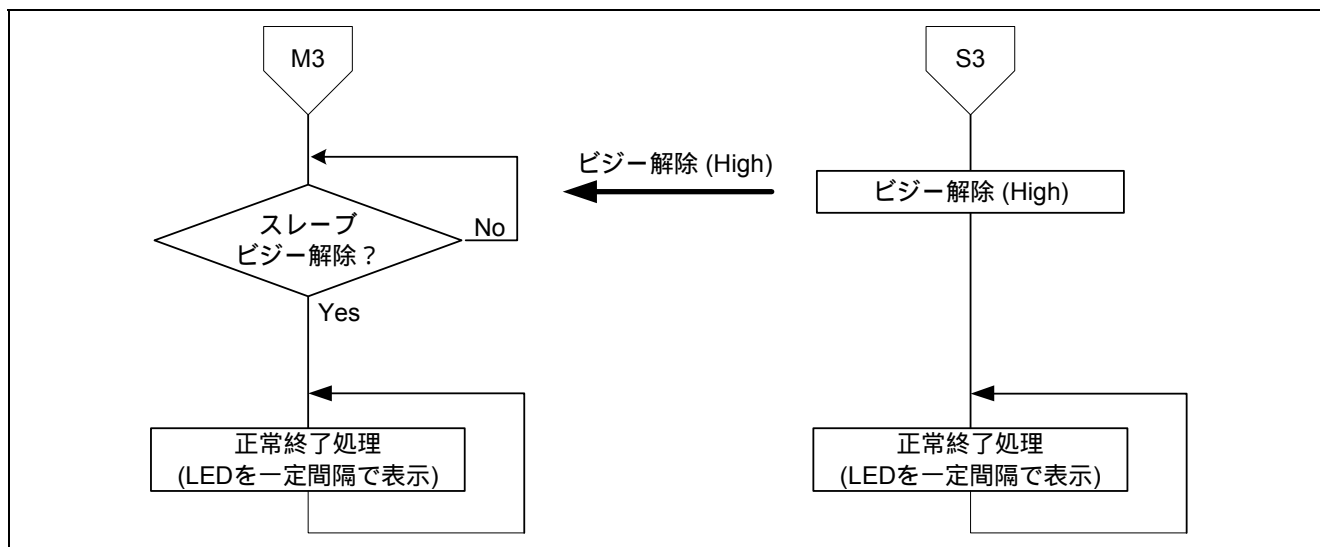


図5.4 通信フロー (4)

5.1.4 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 (記号定数で定義された 1 バイトのデータ) を受信します。図 5.5 に消去ブロック番号の仕様を示します。

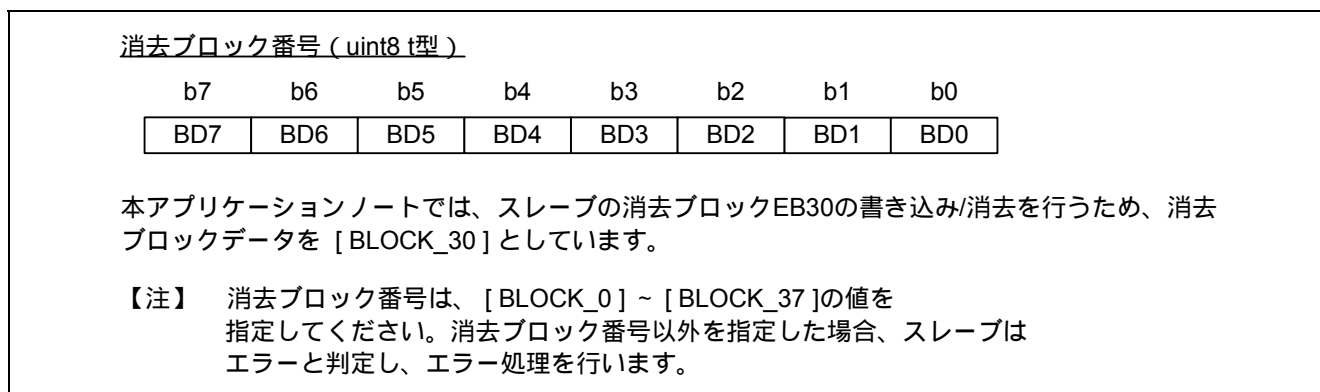


図5.5 消去ブロック番号仕様

5.1.5 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5.6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 (uint32 t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16Kバイトにしているため、書き込みデータサイズを [0000 4000h] 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。
- 【注2】 書き込みデータの送信は128バイト固定とし、128バイトごとに書き込みデータの受信を行います。書き込みデータサイズが128バイトの倍数でない場合、最後の128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、スレーブのユーザマツトに書き込みます。

図5.6 書き込みデータサイズ仕様

5.1.6 オーバランエラー

本アプリケーションノートでは、スレーブのクロック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（SCI0.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。

5.2 ユーザ領域の書き込み/消去

本アプリケーションノートにおけるユーザ領域の書き込み/消去について、以下に説明します。詳細については 8. 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。

5.2.1 RX63T グループ（R5F563TE）のメモリ領域構成

R5F563TEのコード格納用フラッシュメモリは、512Kバイトのユーザ領域と 16Kバイトのユーザブート領域から構成されています。図 5.7にR5F563TEのユーザ領域とユーザブート領域のアドレスを示します。

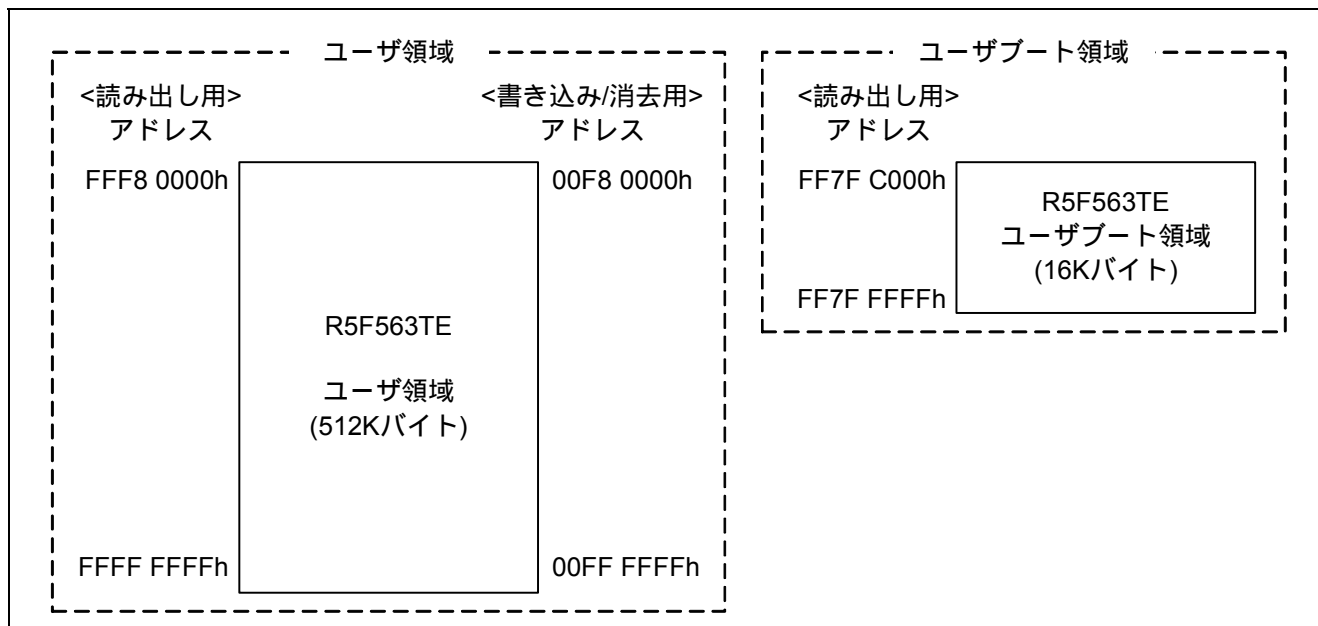


図5.7 R5F563TE のメモリ領域構成

5.3 スレーブ準備完了処理

スレーブは通信準備が完了すると、PF2 から Low レベルを出力します。PF2 は、マスタのスレーブの通信準備完了を監視している端子と接続してください。

PF2 から High レベルが出力している時はスレーブ準備未完了状態を示し、ユーザ領域の書き込み / 消去を受け付けません。

5.4 LED 表示

本サンプルコードの動作状態におけるLED表示を図 5.8に示します。

エラー番号	動作状態	LED表示				順序
		LED3	LED2	LED1	LED0	
	正常終了時 (一定間隔でシフト表示)	●	●	●	○	↑ ↓
		●	●	○	●	
		●	○	●	●	
		○	●	●	●	
エラーNo.01	オーバランエラー発生時	●	●	●	●	
エラーNo.02	[FSTART]コマンドエラー発生時	●	●	●	●	
エラーNo.03	[ERASE]コマンドエラー発生時	●	●	●	●	
エラーNo.04	消去ブロック番号エラー発生時	●	●	●	●	
エラーNo.05	消去処理エラー発生時	●	●	●	●	
エラーNo.06	[WRITE]コマンドエラー発生時	●	●	●	●	
エラーNo.07	書き込みデータサイズ エラー発生時	●	●	●	●	
エラーNo.08	書き込み処理エラー発生時	●	●	●	●	

点灯： ○ 点滅： ● 消灯： ●

図5.8 LED 表示

5.5 ハンドシェイク制御

5.5.1 Busy ポート

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おり、ハンドシェイク用の信号を Busy ポート (IRQ1-DS) から出力しています。

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 Busy ポートをビジー状態にアサート (Low) します。なお、次のシリアル通信が受信可能になってから Busy ポートをビジー解除にネゲート (High) します。

5.5.2 RX63T グループ（R5F563TE）の消去ブロック構成

R5F563TE のユーザ領域は 16K バイト（30 ブロック）、4K バイト（8 ブロック）に分割されており、消去はこのブロック単位で行います。

また、ユーザ領域への書き込みは、下位アドレスが 00h で始まる 128 バイト単位で行います。

表 5.3 に R5F563TE のユーザ領域の消去ブロック構成を示します。

表 5.3 R5F563TE の消去ブロック構成

消去ブロック	読み出し用		書き込み/消去用		サイズ (バイト)
	開始アドレス	終了アドレス	開始アドレス	終了アドレス	
BLOCK_37	FFF8 0000h	FFF8 3FFFh	00F8 0000h	00F8 3FFFh	16K
BLOCK_36	FFF8 4000h	FFF8 7FFFh	00F8 4000h	00F8 7FFFh	16K
BLOCK_35	FFF8 8000h	FFF8 BFFFh	00F8 8000h	00F8 BFFFh	16K
BLOCK_34	FFF8 C000h	FFF8 FFFFh	00F8 C000h	00F8 FFFFh	16K
BLOCK_33	FFF9 0000h	FFF9 3FFFh	00F9 0000h	00F9 3FFFh	16K
BLOCK_32	FFF9 4000h	FFF9 7FFFh	00F9 4000h	00F9 7FFFh	16K
BLOCK_31	FFF9 8000h	FFF9 BFFFh	00F9 8000h	00F9 BFFFh	16K
BLOCK_30	FFF9 C000h	FFF9 FFFFh	00F9 C000h	00F9 FFFFh	16K
BLOCK_29	FFFA 0000h	FFFA 3FFFh	00FA 0000h	00FA 3FFFh	16K
BLOCK_28	FFFA 4000h	FFFA 7FFFh	00FA 4000h	00FA 7FFFh	16K
BLOCK_27	FFFA 8000h	FFFA BFFFh	00FA 8000h	00FA BFFFh	16K
BLOCK_26	FFFA C000h	FFFA FFFFh	00FA C000h	00FA FFFFh	16K
BLOCK_25	FFFB 0000h	FFFB 3FFFh	00FB 0000h	00FB 3FFFh	16K
BLOCK_24	FFFB 4000h	FFFB 7FFFh	00FB 4000h	00FB 7FFFh	16K
BLOCK_23	FFFB 8000h	FFFB BFFFh	00FB 8000h	00FB BFFFh	16K
BLOCK_22	FFFB C000h	FFFB FFFFh	00FB C000h	00FB FFFFh	16K
BLOCK_21	FFFC 0000h	FFFC 3FFFh	00FC 0000h	00FC3FFFFh	16K
BLOCK_20	FFFC 4000h	FFFC 7FFFh	00FC 4000h	00FC 7FFFh	16K
BLOCK_19	FFFC 8000h	FFFC BFFFh	00FC 8000h	00FC BFFFh	16K
BLOCK_18	FFFC C000h	FFFC FFFFh	00FC C000h	00FC FFFFh	16K
BLOCK_17	FFFD 0000h	FFFD 3FFFh	00FD 0000h	00FD 3FFFh	16K
BLOCK_16	FFFD 4000h	FFFD 7FFFh	00FD 4000h	00FD 7FFFh	16K
BLOCK_15	FFFD 8000h	FFFD BFFFh	00FD 8000h	00FD BFFFh	16K
BLOCK_14	FFFD C000h	FFFD FFFFh	00FD C000h	00FD FFFFh	16K
BLOCK_13	FFFE 0000h	FFFE 3FFFh	00FE 0000h	00FE 3FFFh	16K
BLOCK_12	FFFE 4000h	FFFE 7FFFh	00FE 4000h	00FE 7FFFh	16K
BLOCK_11	FFFE 8000h	FFFE BFFFh	00FE 8000h	00FE BFFFh	16K
BLOCK_10	FFFE C000h	FFFE FFFFh	00FE C000h	00FE FFFFh	16K
BLOCK_09	FFFF 0000h	FFFF 3FFFh	00FF 0000h	00FF 3FFFh	16K
BLOCK_08	FFFF 4000h	FFFF 7FFFh	00FF 4000h	00FF 7FFFh	16K
BLOCK_07	FFFF 8000h	FFFF 8FFFh	00FF 8000h	00FF 8FFFh	4K
BLOCK_06	FFFF 9000h	FFFF 9FFFh	00FF 9000h	00FF 9FFFh	4K
BLOCK_05	FFFF A000h	FFFF AFFFh	00FF A000h	00FF AFFFh	4K
BLOCK_04	FFFF B000h	FFFF BFFFh	00FF B000h	00FF BFFFh	4K
BLOCK_03	FFFF C000h	FFFF CFFFh	00FF C000h	00FF CFFFh	4K
BLOCK_02	FFFF D000h	FFFF DFFFh	00FF D000h	00FF DFFFh	4K
BLOCK_01	FFFF E000h	FFFF EFFFh	00FF E000h	00FF EFFFh	4K
BLOCK_00	FFFF F000h	FFFF FFFFh	00FF F000h	00FF FFFFh	4K

5.6 セクション設定

スレーブのセクション設定を表 5.4に示します。

表5.4 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	
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	
RPFRAM		[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	
PResetPRG	FF7F C000h	プログラム領域 (PowerON_Reset_PC プログラム)	
C_1	FF7F C100h	定数領域 (ALIGN = 1)	
C_2		定数領域 (ALIGN = 2)	
C		定数領域 (ALIGN = 4)	
C\$*		定数領域	
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)	
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)	
W*		switch 文分岐テーブル領域 (ALIGN = 4)	
PFRAM		プログラム領域 (ユーザ領域の書き込み/制御プログラム)	
FIXEDVECT		FF7F FFFCh	固定ベクタ領域 (リセットベクタ)

5.7 必要メモリサイズ

表 5.5に 必要メモリサイズを示します。

表5.5 必要メモリサイズ

使用メモリ	サイズ	備考
ROM	6345 バイト	
RAM	4413 バイト	
最大使用ユーザスタック	92 バイト	
最大使用割り込みスタック	0 バイト	割り込みを使用していないため

【注】 必要メモリサイズはC コンパイラのバージョンやコンパイルオプションにより異なります。

5.8 ファイル構成

表 5.6にサンプルコードで使用するファイルを示します。なお、統合開発環境で自動生成されるファイルのうち、内容を変更していないファイルは除きます。

表5.6 サンプルコードで使用するファイル

ファイル名	概要	備考
r_init_module_stop.c	RX63T 初期設定例のプログラム	詳細は RX63T 初期設定例のアプリケーションノートを参照してください。
r_init_clock.c		
r_init_non_existent_port.c		
r_init_module_stop.h	RX63T 初期設定例のパラメータ設定用インクルードヘッダ	
r_init_clock.h		
r_init_non_existent_port.h		
vecttbl.c (* ¹)	固定ベクタテーブル定義	
vect.h (* ²)	割り込み処理関数定義	
intprg.c (* ³)	割り込み処理	
main.c	マスタとのクロック同期式シリアル通信による通信コマンド、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、ユーザ領域のブロック消去および書き込み制御、正常終了時およびエラー発生時の LED の表示制御	
r_flash_api_rx.c	シンプルフラッシュ API のプログラム	詳細は RX600&RX200 シリーズ RX 用の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。
locking.c		
mcu_locks.c	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	
r_flash_api_rx_if.h		
locking.h		
mcu_locks.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	
r_flash_api_rx_private.h		
r_flash_api_rx_config.h		
r_bsp.h		
r_bsp_config.h		
r_flash_api_rx63t.h		
mcu_info.h		

【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、固定ベクタテーブルの特権命令例外、未定義命令例外、浮動小数点例外、ノンマスクブル割り込み、予約領域の Dummy 関数の定義をコメントアウトして、リセットベクタだけの定義に変更しています。

*2 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、Excep_SuperVisorInst 関数、Dummy 関数、Excep_UndefinedInst 関数、Excep_FloatingPoint 関数、NonMaskableInterrupt 関数の定義をコメントアウトしています。

*3 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、Excep_SuperVisorInst 関数、Dummy 関数、Excep_UndefinedInst 関数、Excep_FloatingPoint 関数、NonMaskableInterrupt 関数の記述をコメントアウトしています。

5.9 オプション設定メモリ

表 5.7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。なお、本サンプルコードではマスタ側とスレーブ側でエンディアンを合わせてご使用ください。

表5.7 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、IWDTC は停止 リセット後、WDT は停止
OFS1	FFFF FF8Bh - FFFF FF88h	FFFF FFFFh	リセット後、電圧監視リセット 0 無効
MDEB (*1)	FF7F FFF8h - FFFF FFFBh	FFFF FFFFh FFFF FFF8h	(ユーザブートモード時) リトルエンディアン ビッグエンディアン

【注】 *1 本サンプルコードの設定はリトルエンディアンです。エンディアンの切り替えは 6.4 エンディアンの設定を参照ください。

5.10 定数一覧

表 5.8にサンプルコードで使用する定数を示します。

表5.8 サンプルコードで使用する定数

定数名	設定値	内容
FSTART	0x10	書き込み/消去開始コマンド
ERASE	0x11	消去開始コマンド
WRITE	0x12	書き込み開始コマンド
LED_ON	0	LED 点灯時の設定値
LED_OFF	1	LED 消灯時の設定値
RSK_LED0	PORT7.PODR.BIT.B1	評価ボード搭載 LED0 の点灯/消灯制御
RSK_LED1	PORT7.PODR.BIT.B2	評価ボード搭載 LED1 の点灯/消灯制御
RSK_LED2	PORT7.PODR.BIT.B3	評価ボード搭載 LED2 の点灯/消灯制御
RSK_LED3	PORT3.PODR.BIT.B3	評価ボード搭載 LED3 の点灯/消灯制御
RSK_LED0_PDR	PORT7.PDR.BIT.B1	評価ボード搭載 LED0 の入出力制御
RSK_LED1_PDR	PORT7.PDR.BIT.B2	評価ボード搭載 LED1 の入出力制御
RSK_LED2_PDR	PORT7.PDR.BIT.B3	評価ボード搭載 LED2 の入出力制御
RSK_LED3_PDR	PORT3.PDR.BIT.B3	評価ボード搭載 LED3 の入出力制御
RxD0_PMR	PORTB.PMR.BIT.B1	RxD0 のポートモードレジスタ設定
TxD0_PMR	PORTB.PMR.BIT.B3	TxD0 のポートモードレジスタ設定
WRITE_ADRS_TOP_16K	0x00F80000	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
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 ~ EB37 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
NOT_READY	1	スレーブ準備未完了の設定値
READY	0	スレーブ準備完了の設定値
SLAVE_READY_PODR	PORTF.PODR.BIT.B2	スレーブ準備完了通知のポート出力レジスタ設定

定数名	設定値	内容
SLAVE_READY_PDR	PORTF.PDR.BIT.B2	スレーブ準備完了通知のポート方向レジスタ設定
NON_BUSY	0	Busy ポートアサート時の設定値
BUSY	1	Busy ポートネゲート時の設定値
SLAVE_BUSY	PORT1.PODR.BIT.B1	Busy ポートの出力制御
SLAVE_BUSY_PDR	PORT1.PDR.BIT.B1	Busy ポートの入出力制御
WAIT_LED	2000000	スレーブのユーザ領域の書き込み/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
ROM_PROGRAM_SIZE	128 (*2)	対象デバイスに応じた、ユーザ領域への書き込み単位が設定されます。r_flash_api_rx63t.h に記述されており、インクルードすることで参照するようにしています。
ERROR_NO_01	1	エラー状態を示すデータ
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	

【注】 *2 RX63T グループを対象デバイスにした場合の値になります。

5.11 構造体/共用体一覧

図 5.9 にサンプルコードで使用する構造体を示します。

```

typedef struct
{
    uint8_t *p_write_adrs_top;           /* 書き込み時の対象消去ブロックの先頭アドレス */
    uint8_t *p_write_adrs_end;         /* 書き込み時の対象消去ブロックの終了アドレス */
    uint8_t *p_write_adrs_now;        /* 書き込み時の書き込み先アドレス */
    uint32_t eb_block_size;           /* 対象消去ブロックのブロックサイズ */
}st_fcu_info_t;

```

図5.9 サンプルコードで使用する構造体

5.12 変数一覧

スレーブが使用する グローバル変数を 表 5.9 に示します。

表5.9 グローバル変数

型	変数名	内容	使用関数
uint8_t	wrdata_buffer [ROM_PROGRA M_SIZE]	マスタから受信した 128 バイトの書き込みデータを格納する配列 (128 バイト)	Flash_Update
st_fcu_info_t	fcu_info	ユーザ領域の書き込み / 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)	Flash_Update

5.13 関数一覧

表 5.10に関数を示します。

表5.10 関数

関数名	概要
main	メイン関数
Flash_Update	ユーザ領域書き込み/消去制御関数
Indicate_Ending_LED	正常終了処理関数
Indicate_Error_LED	エラー終了処理関数
SCI_Rcv1byte	1バイトデータ受信関数
SCI_Rcvnbyte	nバイトデータ受信関数
sci0_init	SCI 初期設定関数
lcu_init	ICU 初期設定関数
mpc_init	MPC 初期設定関数
pmr_init	PMR 初期設定関数

5.14 関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン関数
ヘッダ	なし
宣言	void main (void)
説明	<p>main 関数は、以下の処理をします。</p> <ul style="list-style-type: none"> 存在しないポートの初期設定 クロック設定（システムクロック（ICLK）、周辺モジュールクロック（PCLKB）など） LED0～LED3 を接続している I/O ポート（P71、P72、P73、P33）の初期出力設定、Busy ポート（P11）の初期出力設定 SCI0 の初期設定 SCI のポート設定マルチファンクションピンコントローラ（MPC）の設定 IRQ のポート設定ポートモードレジスタ（PMR）の設定 スレーブ準備完了の通知 マスタからの 1 バイトデータの受信制御 ユーザ領域書き込み / 消去制御プログラムをユーザ領域（PF_UPDATE_FUNC セクション）から内蔵 RAM（RF_UPDATE_FUNC セクション）にコピー 内蔵 RAM 上のユーザ領域書き込み/制御プログラム（Flash_Update 関数）の呼び出し
引数	なし
リターン値	なし

Flash_Update	
概要	ユーザ領域の書き込み / 消去処理関数
ヘッダ	なし
宣言	static void Flash_Update (void)
説明	Flash_Update 関数は、マスタから消去ブロック番号、書き込みデータサイズ、書き込みデータをクロック同期式シリアル通信による通信で受信し、ユーザ領域の書き込み/消去を行います。 ユーザ領域の書き込み/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。
引数	なし
リターン値	なし

Indicate_Ending_LED	
概要	正常終了処理関数
ヘッダ	なし
宣言	static void Indicate_Ending_LED (void)
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。
引数	なし
リターン値	なし

Indicate_Error_LED	
概要	エラー終了処理関数
ヘッダ	なし
宣言	static void Indicate_Error_LED (uint8_t error_no)
説明	Indicate_Error_LED 関数は、ユーザ領域の書き込み / 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。
引数	uint8_t error_no : ユーザ領域の書き込み / 消去中に発生したエラー番号 (*1)
リターン値	なし

【注】 *1 エラー番号は、「5.4 LED表示」を参照してください。

SCI_Rcv1byte	
概要	1 バイトデータ受信関数
ヘッダ	なし
宣言	static uint8_t SCI_Rcv1byte (void)
説明	SCI_Rcv1byte 関数は、SCI0 のクロック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。
引数	なし
リターン値	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

SCI_Rcvnbyte	
概要	n バイトデータ受信関数
ヘッダ	なし
宣言	static void SCI_Rcvnbyte (uint16_t size, uint8_t *rcv_buffer)
説明	SCI_Rcvnbyte 関数は、SCI0 のクロック同期式シリアル通信による n バイトデータ (n は uint16_t 型の第 1 引数) の受信制御を行います。
引数	uint16_t size : SCI1 のクロック同期式シリアル通信による受信データ バイト数 uint8_t *rcv_buffer : 受信データ格納場所の先頭アドレス
リターン値	なし
sci0_init	
概要	SCI 初期設定関数
ヘッダ	なし
宣言	static void sci0_init (void)
説明	SCI の初期設定を行います。 ポート設定は含みません。
引数	なし
リターン値	なし
icu_init	
概要	ICU 初期設定関数
ヘッダ	なし
宣言	static void icu_init (void)
説明	ICU の初期設定を行います。 シリアルの割り込み設定を行います。
引数	なし
リターン値	なし
mpc_init	
概要	MPC 初期設定
ヘッダ	なし
宣言	static void mpc_init (void)
説明	MPC を下記の機能に選択します <ul style="list-style-type: none"> • PD4 → SCK0 • PD5 → RXD0
引数	なし
リターン値	なし
pmr_init	
概要	PMR 初期設定
ヘッダ	なし
宣言	static void pmr_init (void)
説明	PMR の初期設定を行います。 <ul style="list-style-type: none"> • PD3、PD4、PD5 を周辺機能として使用
引数	なし
リターン値	なし

5.15 フローチャート

5.15.1 メイン関数

図 5.10にメイン関数のフローチャートを示します。

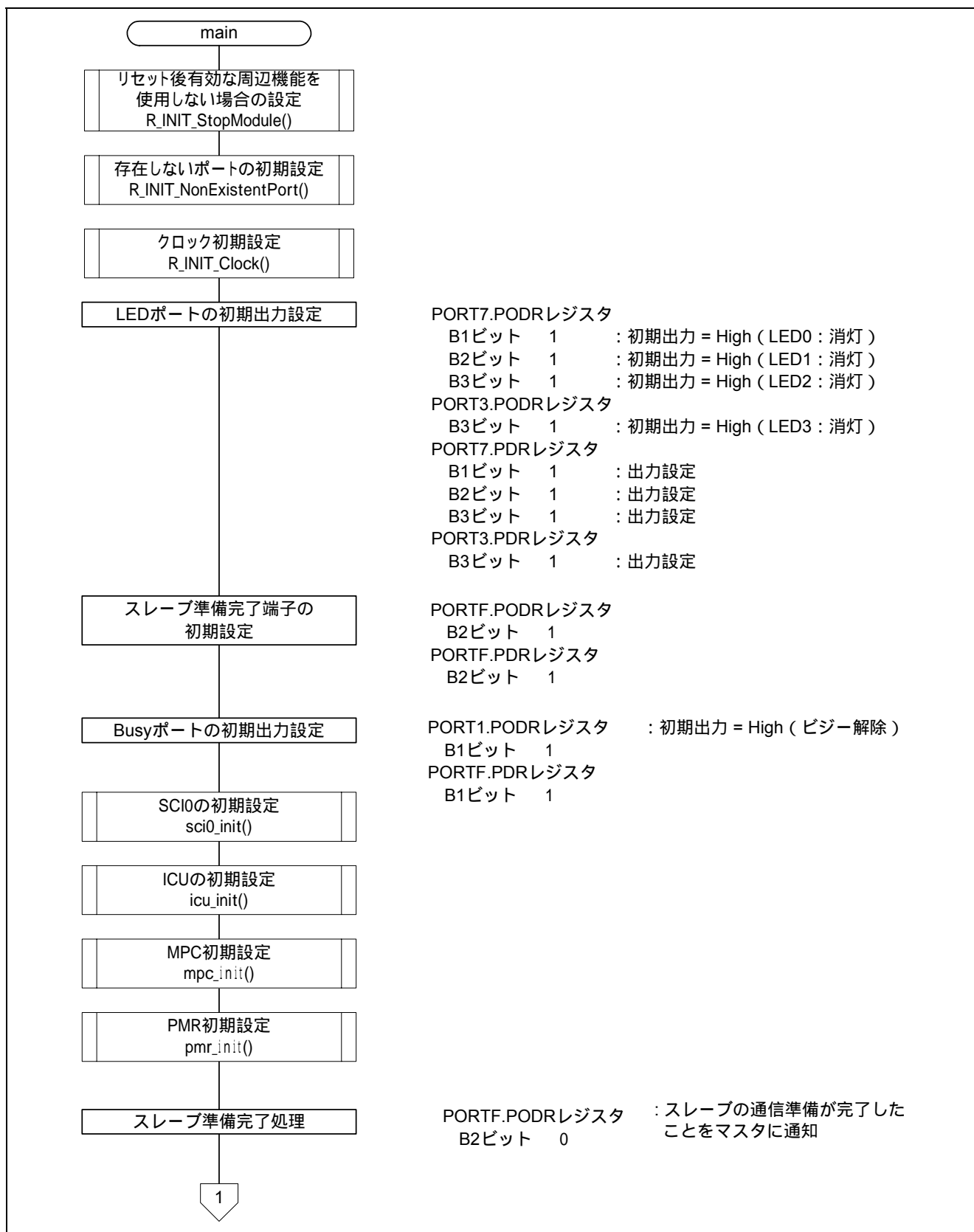


図5.10 メイン関数 (1)

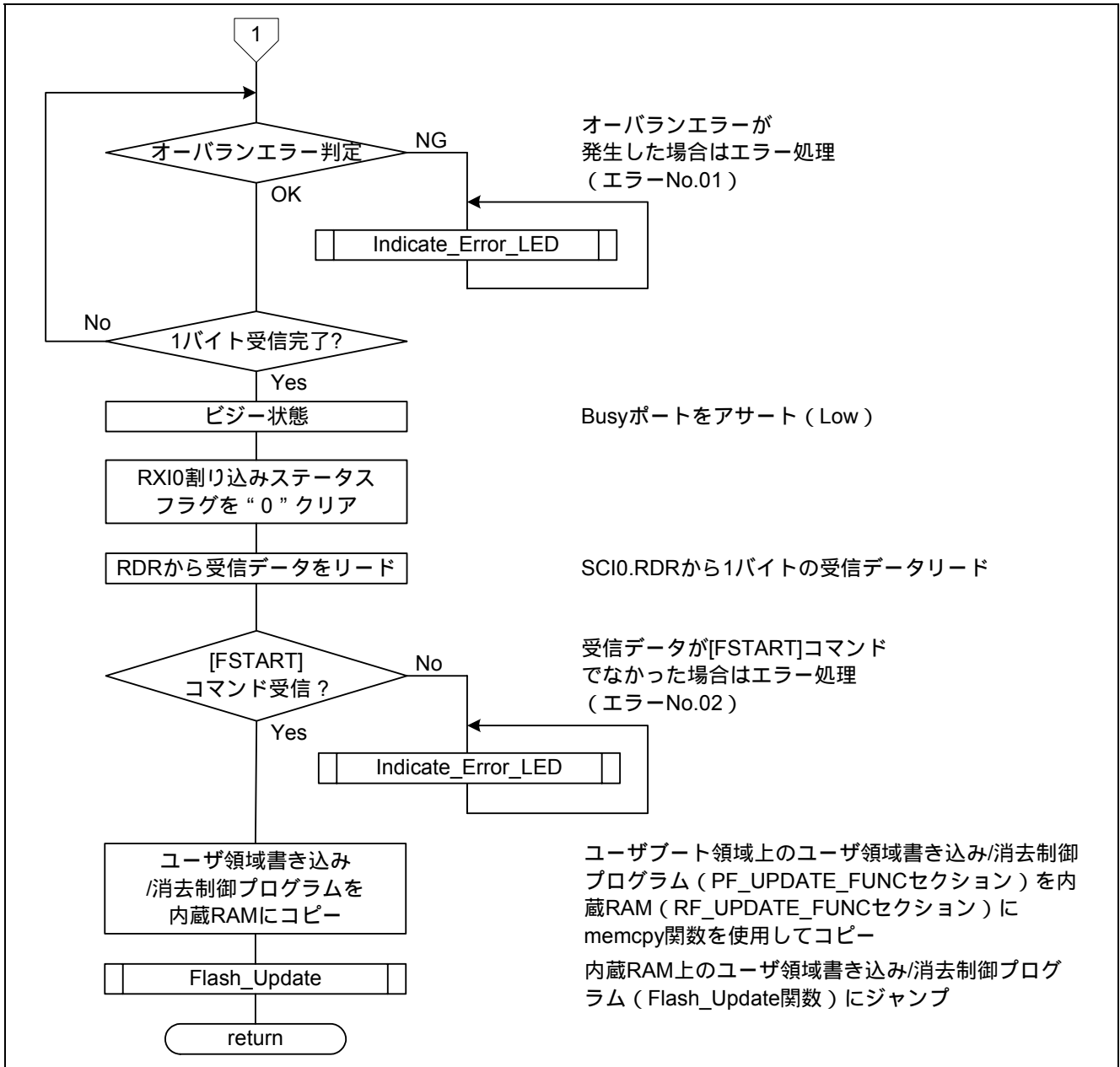


図5.11 メイン関数 (2)

5.15.2 書き込み/消去関数

図 5.12に書き込み/消去処理のフローチャートを示します。

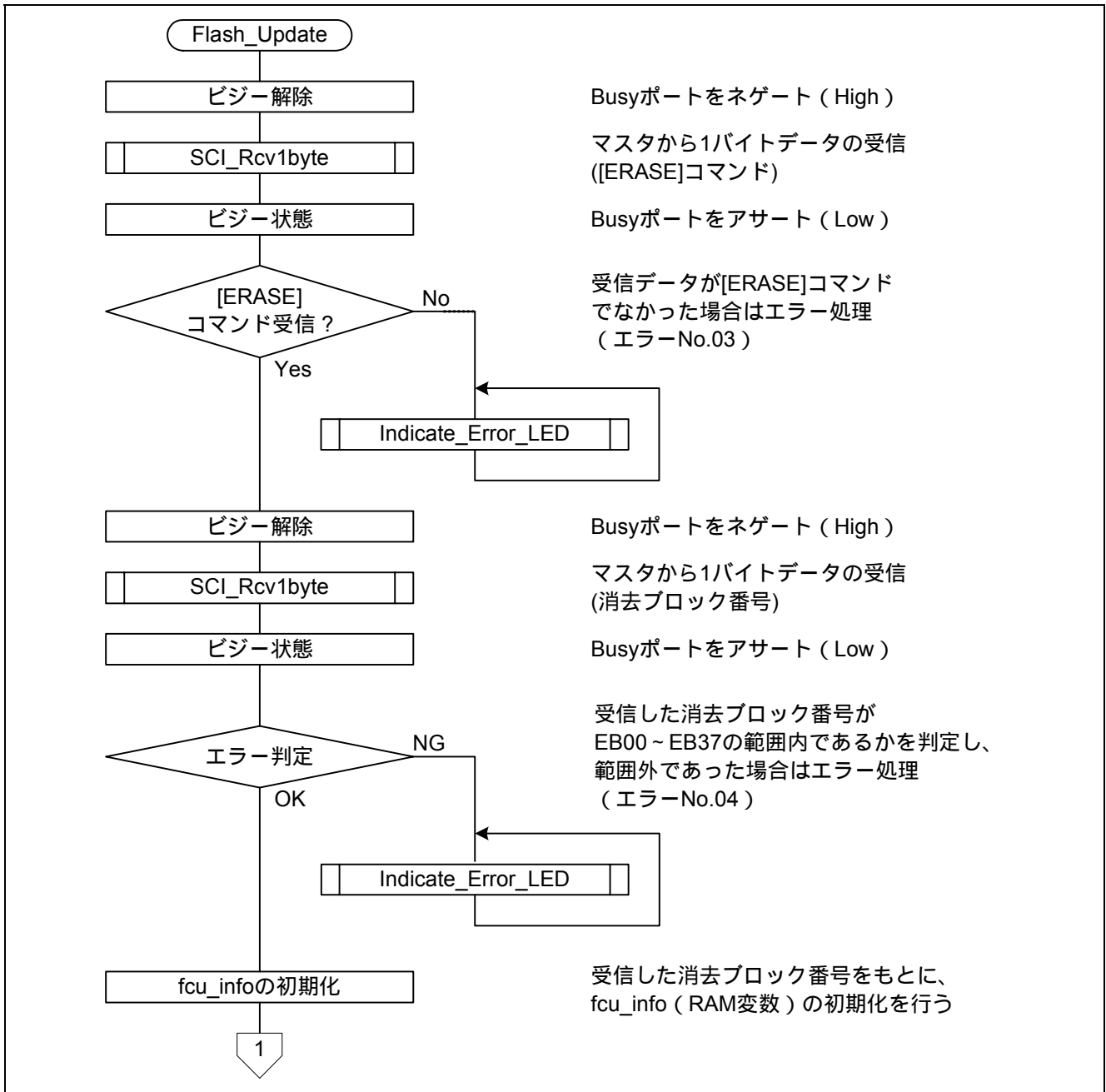


図5.12 書き込み/消去処理（1）

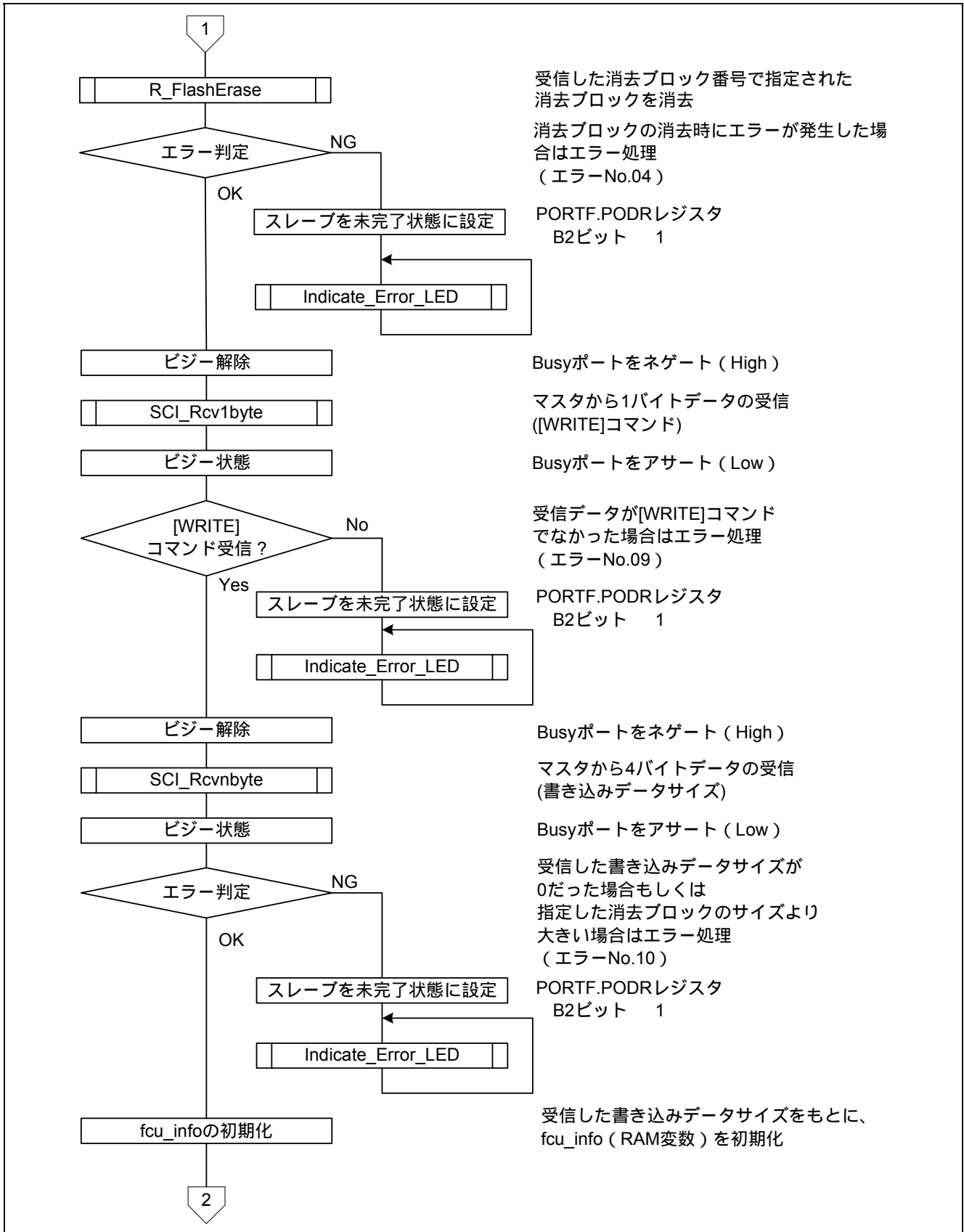


図5.13 書き込み/消去処理（2）

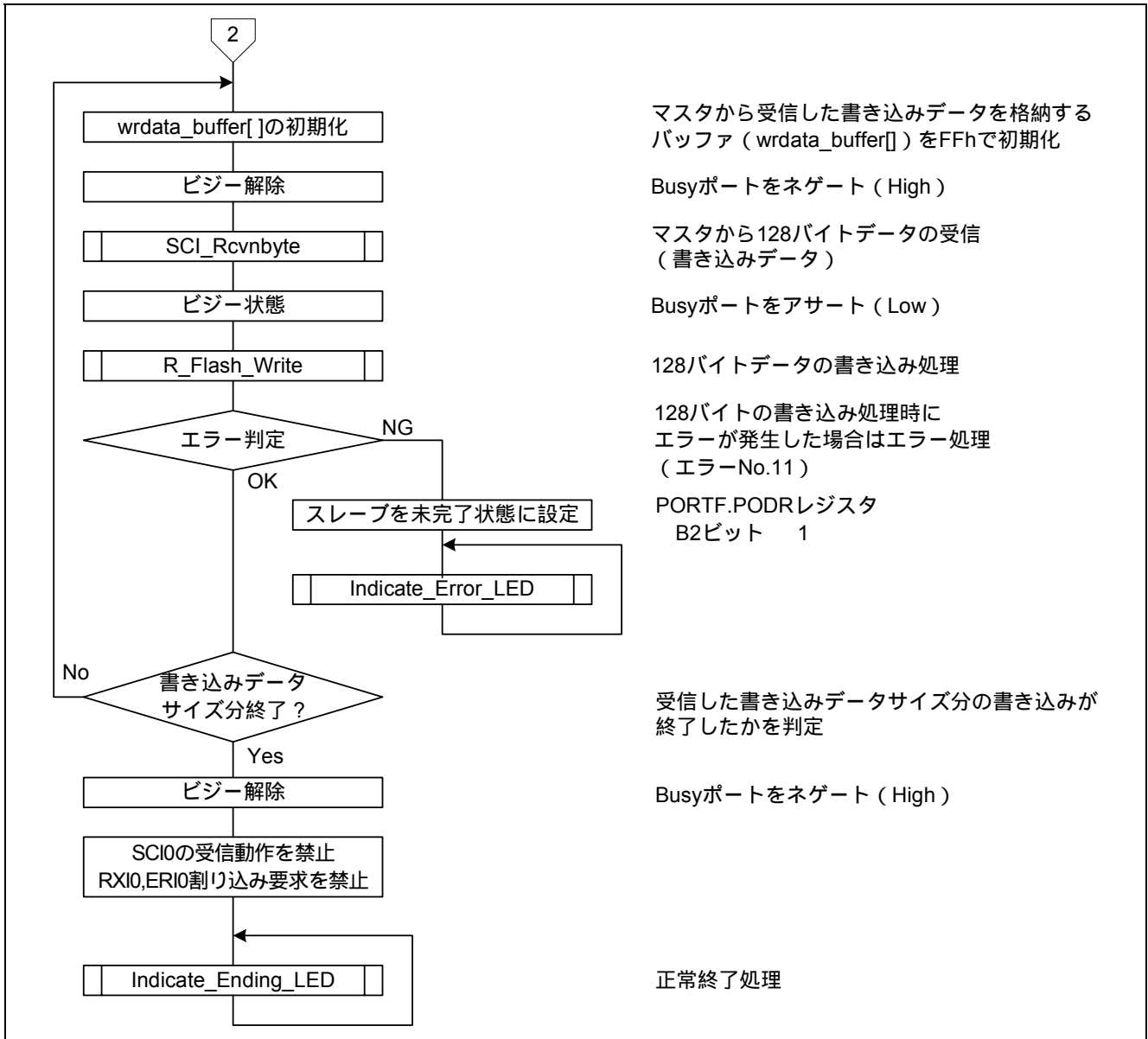


図5.14 書き込み/消去処理 (3)

5.15.3 正常処理関数

図 5.15に正常処理関数のフローチャートを示します。

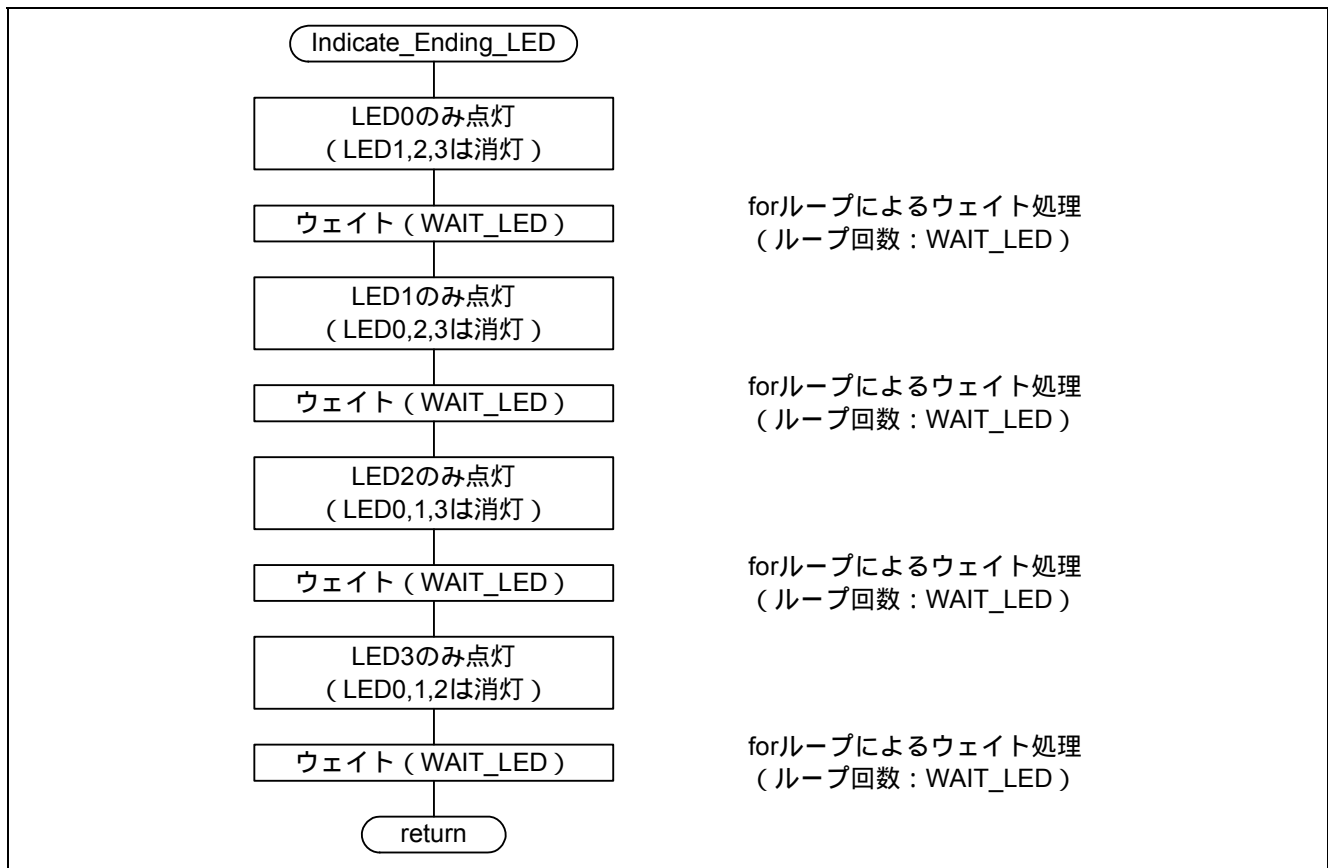


図5.15 正常処理関数

5.15.4 エラー終了処理関数

図 5.16にエラー終了処理関数のフローチャートを示します。

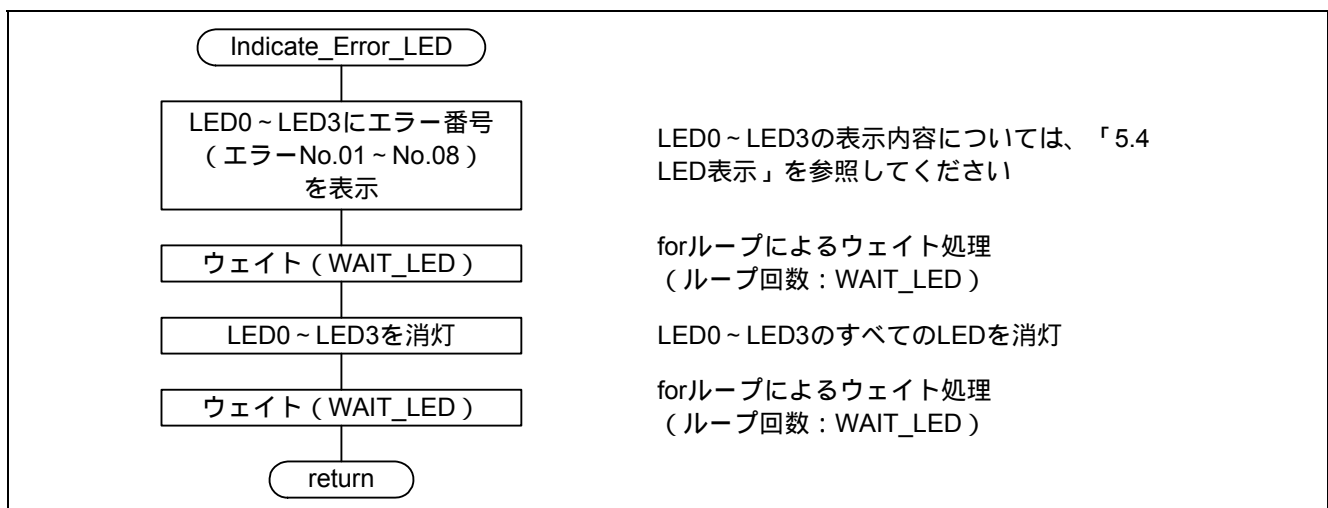


図5.16 エラー終了処理関数

5.15.5 1バイトデータ受信関数

図 5.17に 1バイトデータ受信関数のフローチャートを示します。

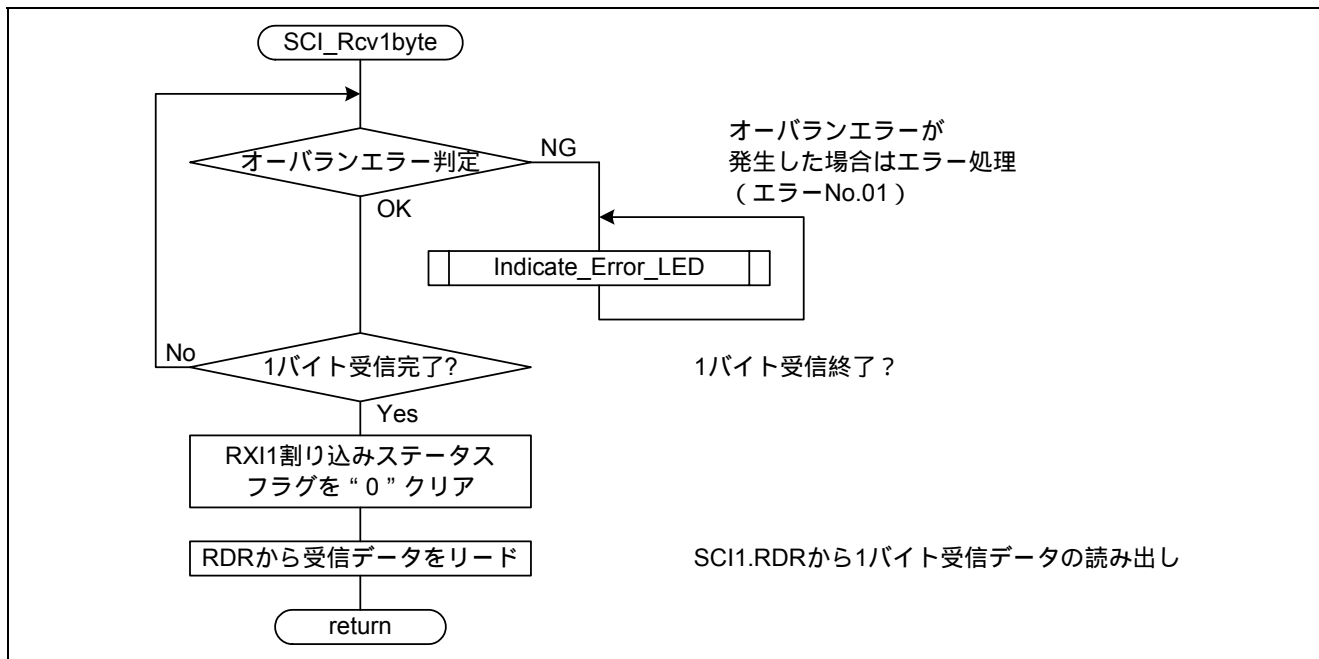


図5.17 1バイトデータ受信関数

5.15.6 n バイトデータ受信関数

図 5.18にnバイトデータ受信関数のフローチャートを示します。

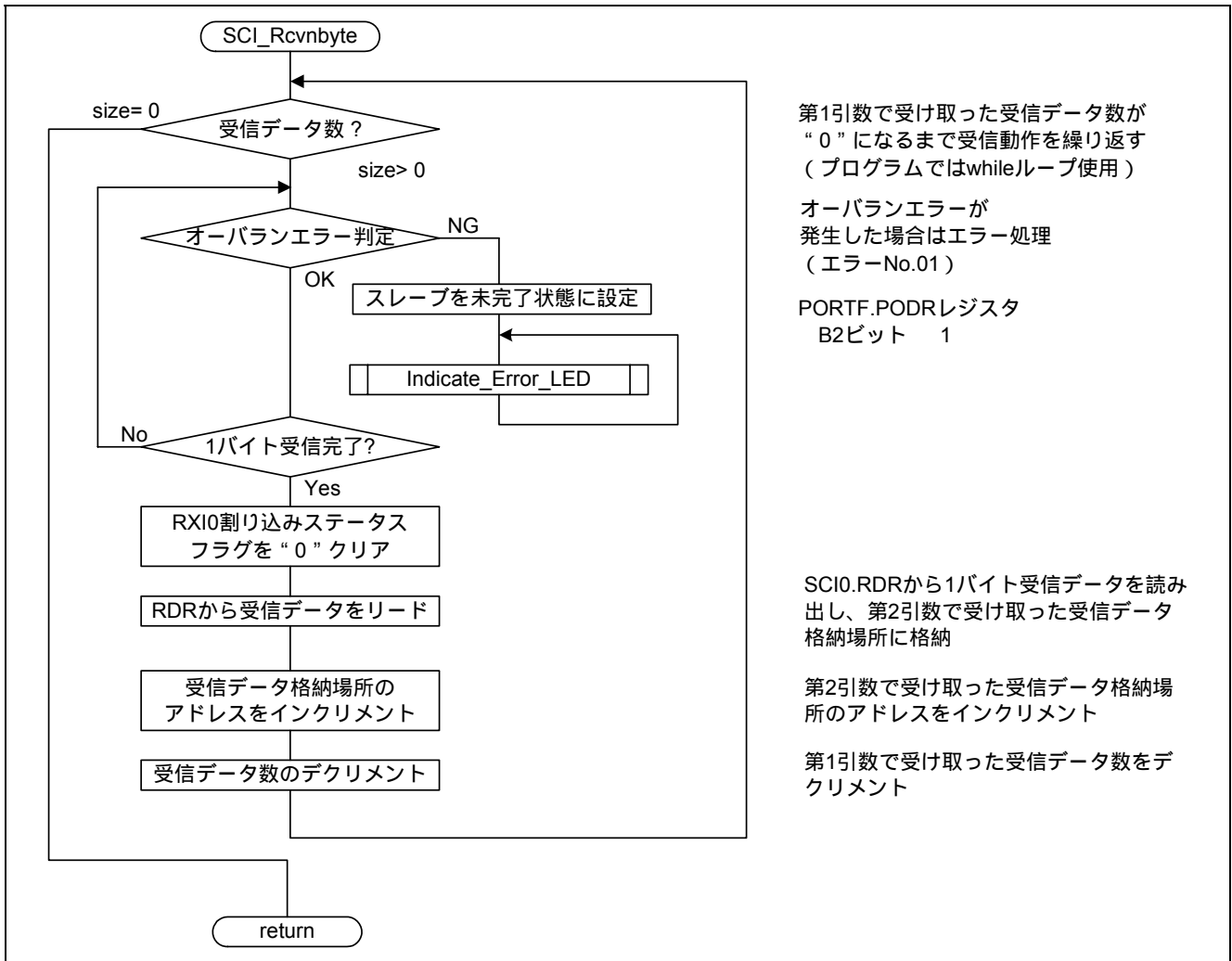


図5.18 nバイトデータ受信関数

5.15.7 SCI 初期設定関数

図 5.19にSCI初期設定関数のフローチャートを示します。

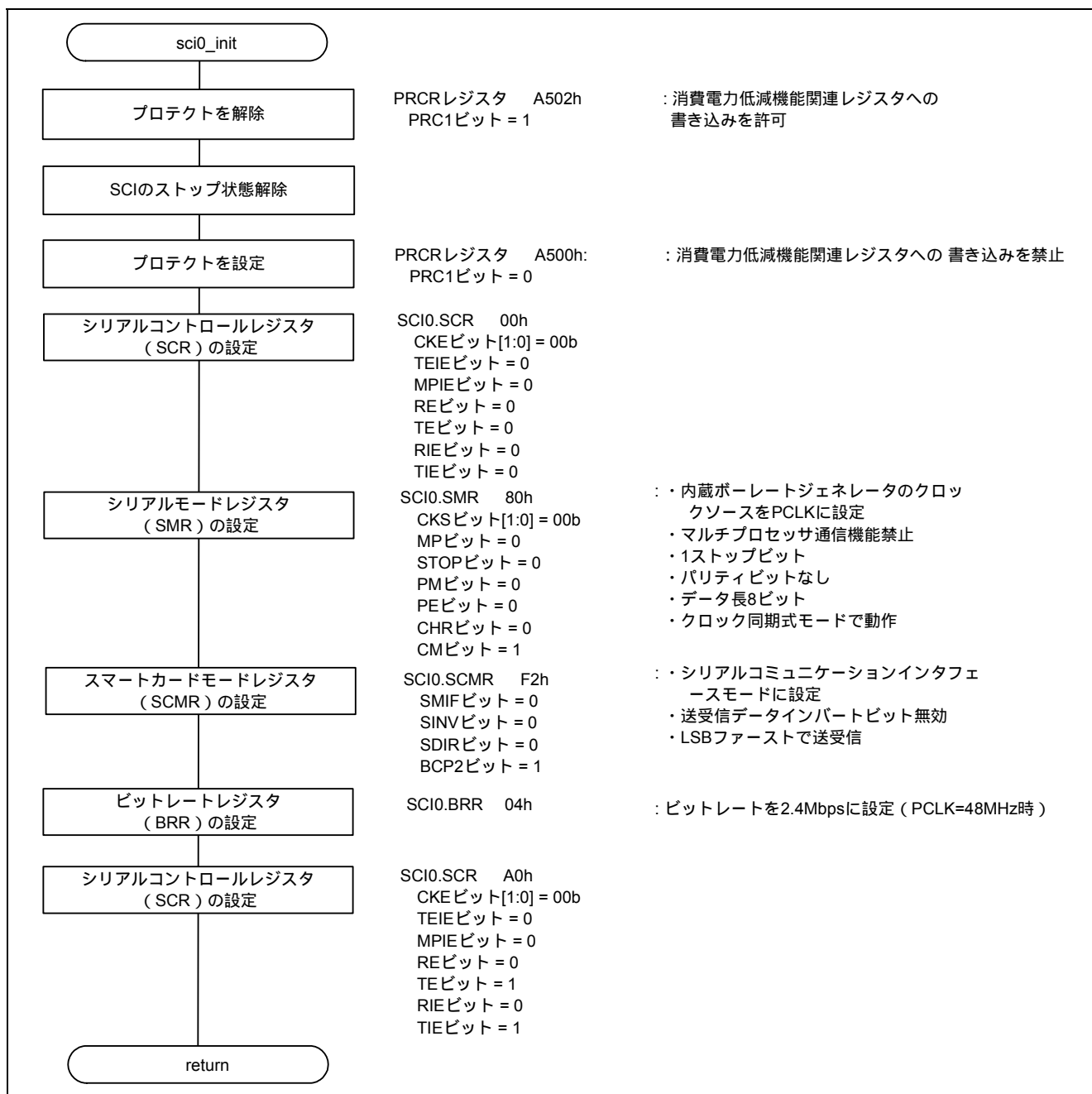


図5.19 SCI 初期設定関数

5.15.8 ICU 初期設定関数

図 5.20にICU初期設定関数のフローチャートを示します。

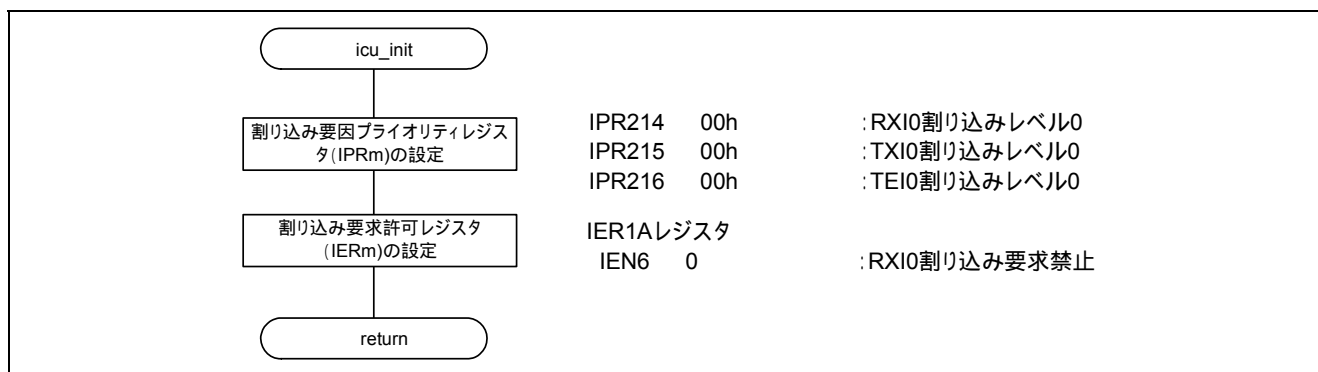


図5.20 ICU 初期設定関数

5.15.9 MPC 初期設定関数

図 5.21にMPC初期設定関数のフローチャートを示します。

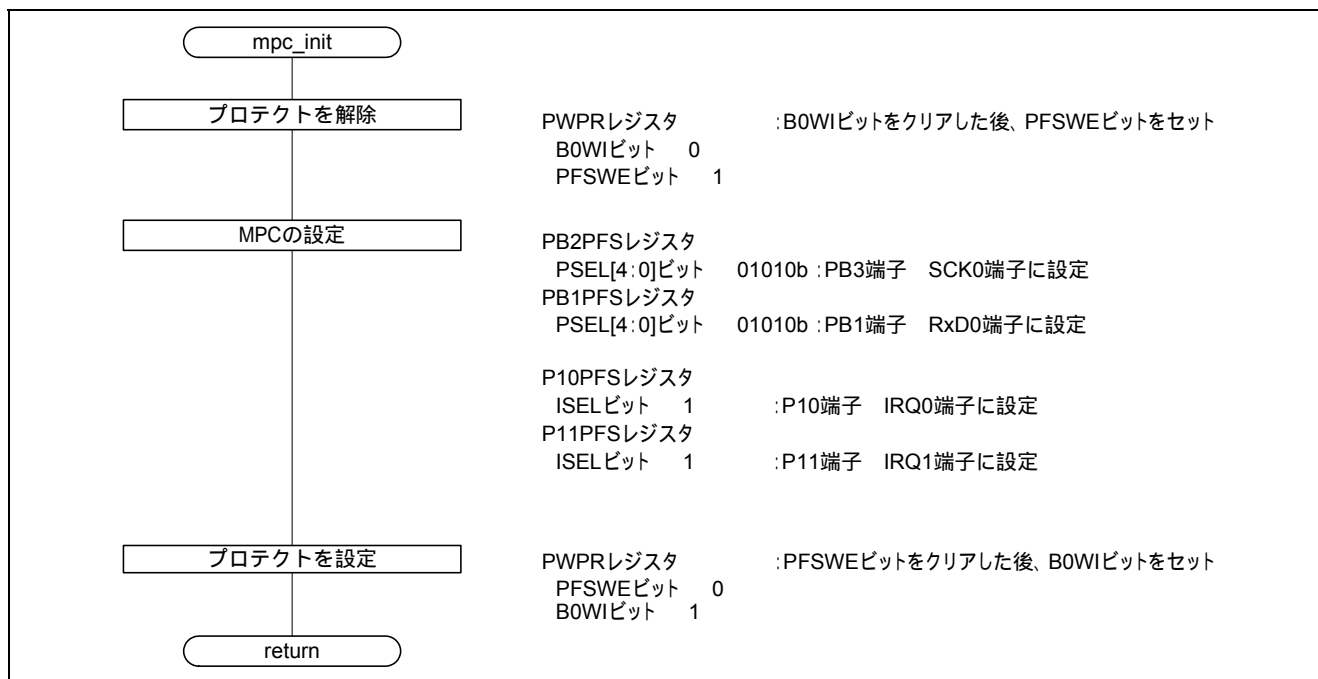


図5.21 MPC 初期設定関数

5.15.10 PMR 初期設定関数

図 5.22にPMR初期設定関数のフローチャートを示します。

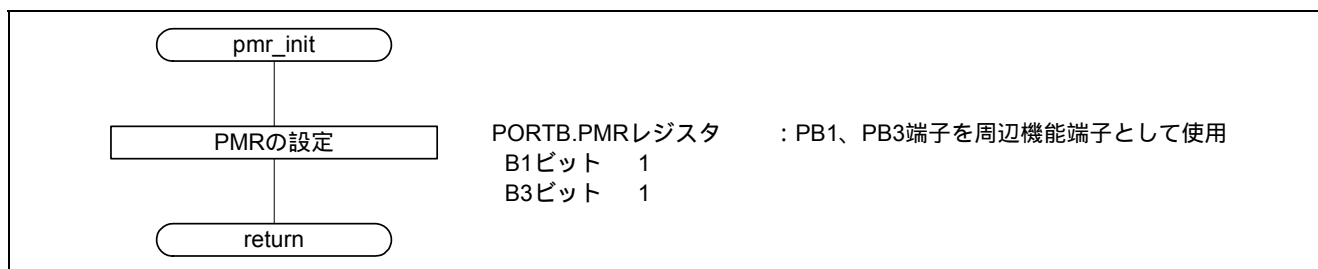


図5.22 PMR 初期設定関数

6. 注意事項

6.1 ユーザブートモード時の固定ベクタについて

MD、P00 端子によりユーザブートモードに設定しリセット解除を行うと、ユーザブートモードに遷移します。このときのリセットベクタは、ユーザブート領域の FF7F FFFCh 番地になります。その他のベクタテーブルは、通常のベクタテーブルを参照します。

固定ベクタテーブルは、テーブルの配置アドレスが固定されたベクタテーブルです。FFFF FFD0h ~

FFFF FFFh 番地に、特権命令例外、未定義命令例外、浮動小数点例外、ノンマスクابل割り込み、リセットの各ベクタを配置しています。

図 6.1 に本アプリケーションノートにおける固定ベクタテーブルの設定方法を示します。

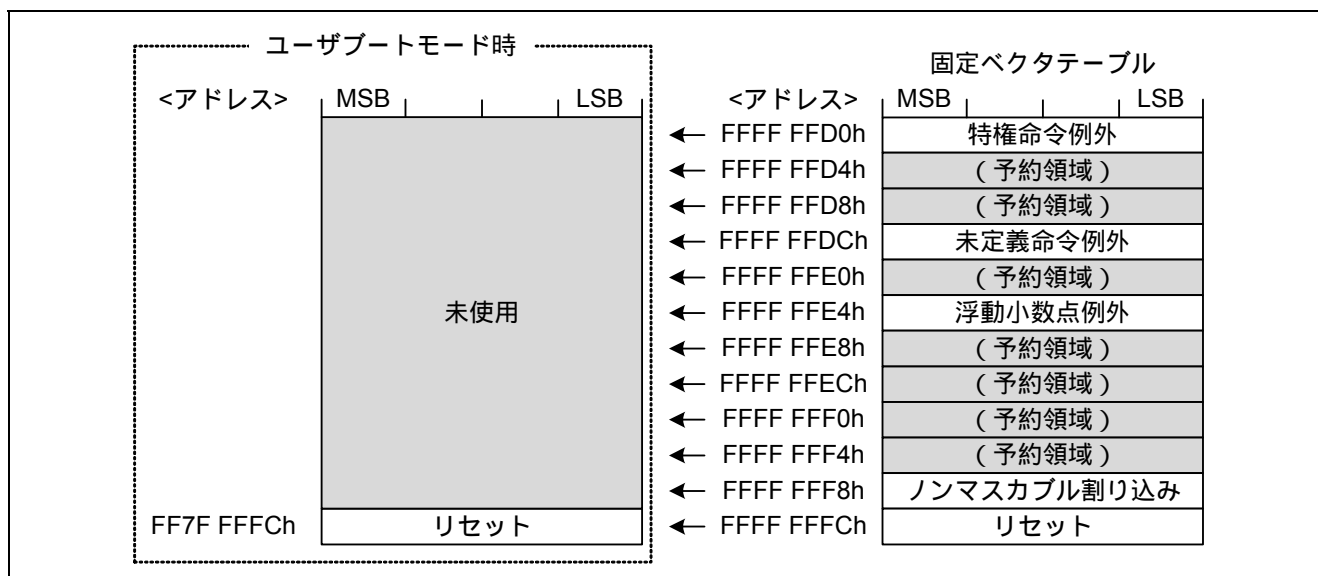


図6.1 本アプリケーションノートにおける固定ベクタテーブル設定方法

本アプリケーションノートのスレーブは、ユーザブートモードで動作します。スレーブの固定ベクタの設定は、High-performance Embedded Workshop が自動生成するファイル (vecttbl.c、vect.h、intprg.c) の固定ベクタテーブルの特権命令例外 (Excep_SuperVisorInst 関数)、未定義命令例外 (Excep_UndefinedInst 関数)、浮動小数点例外 (Excep_FloatingPoint 関数)、ノンマスクابل割り込み (NonMaskableInterrupt)、および予約領域の Dummy 関数をコメントアウトして、リセットだけの固定ベクタ (4 バイト) として、ユーザブート領域の FF7F FFFC 番地に配置しています。

なお、本アプリケーションノートのスレーブプログラムは、ユーザブート領域上のリセットベクタの設定のみを行っており、ユーザ領域の固定ベクタの設定は行っておりません。

ユーザにて本アプリケーションノートのスレーブプログラムを使用される場合には、ユーザ領域の固定ベクタ (特権命令例外、未定義命令例外、浮動小数点例外、ノンマスクابل割り込み、予約領域)、および各例外ハンドラを設定する必要がありますので、ご注意ください。

6.2 消去ブロック BLOCK_0 の書き換え時の注意事項

消去ブロック BLOCK_0 (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。

消去ブロック番号を BLOCK_00 に設定して、BLOCK_0 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、BLOCK_0 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、8. 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 動作モードの設定

本アプリケーションノートでは、モード端子を MD=Low、P00=High に設定し動作モードをユーザブートモードに、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 を有効にそれぞれ設定しています。

本アプリケーションノートにおける動作モードの設定を表 6.1 に示します。

表6.1 動作モードの設定

モード設定端子		SYSCR0 レジスタ	動作モード	内蔵 ROM
MD	P00	ROME		
Low	High	1	ユーザブートモード	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。

6.4 エンディアンの設定

本アプリケーションノートのサンプルコードは、リトルエンディアン/ビッグエンディアンの両方に対応しています。なお、エンディアンはマスタ側とスレーブ側で同じエンディアン設定にしてください。

6.4.1 リトルエンディアン使用時

リトルエンディアンで動作する場合は、コンパイラオプションのエンディアンの設定で “ Little-endianデータ ” を指定してください。5.9 オプション設定メモリの MDEB はリトルエンディアンの値になります。

6.4.2 ビッグエンディアン使用時

ビッグエンディアンで動作する場合は、コンパイラオプションのエンディアンの設定で “ Big-endianデータ ” を指定してください。5.9 オプション設定メモリの MDEB はビッグエンディアンの値になります。

7. 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。

8. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX63T グループ ユーザーズマニュアル ハードウェア編 Rev.2.00

(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。)

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 ファミリー 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(V.1.02 添付資料含む)

(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

RX63T グループ

ユーザブートモードによる内蔵フラッシュメモリ書き換え (マスタ) (R01AN1421JJ) 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。)

アプリケーションノート

RX600&RX200 シリーズ RX 用のシンプルフラッシュ API Rev.2.40

(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/>

お問い合わせ先

<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	2014.02.20	—	初版発行

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、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

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合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。
総合お問合せ窓口：<http://japan.renesas.com/contact/>