

H8S/2400 シリーズ

ユーザプログラムモードによるフラッシュメモリ書き換え (EW0 モード)

R01AN0770JJ0102
Rev.1.02
2011.09.07

要旨

本アプリケーションノートでは、ユーザプログラムモードを使用して内蔵フラッシュメモリの書き換え例を説明します。

H8S/2400 シリーズは、ユーザプログラムモードを使用することによりオンボードの状態の内蔵フラッシュメモリを書き換えることができます。

動作確認デバイス

H8S/2456, H8S/2456R, H8S/2454 グループ

はじめに

本アプリケーションノートは、動作確認デバイス H8S/2456, H8S/2456R, H8S/2454 グループを基に作成しています。

動作確認デバイスと同様の内部 I/O レジスタを持つ他の H8S/2400 シリーズは、本プログラムを使用することができます。ただし、本アプリケーションノートで使用している機能やアドレスマップが、一部変更している場合がありますので、最新のマニュアルを確認し、十分な評価を行ってください。

目次

1. 仕様	2
2. 適用条件	3
3. 使用機能説明	4
4. 動作説明	7
5. ソフトウェア説明	9
6. 参考ドキュメント	27

1. 仕様

本アプリケーションノートは、ユーザプログラムモードを使用して内蔵フラッシュメモリの任意の1ブロックを消去（イレース動作）し、消去したブロックの全領域にデータを書き込む（プログラム動作）例です。

以下に本アプリケーションノートの詳細仕様を示します。

- 消去制御プログラムは内蔵フラッシュメモリのブロック1（ユーザROM: 64KB）を消去します。
- 書き込み制御プログラムは256バイトの書き込みデータをブロック1の全領域に繰り返し書き込みます。
- 書き込みデータは内蔵RAMにソフトウェアで用意したH'00～H'FFのデータとします。
- イレースまたはプログラム動作後は内蔵フラッシュメモリの内容と期待値をベリファイします。
- SW1, SW2を使用しそれぞれイレース動作、プログラム動作の開始を制御します。
- LEDは「表1 LED仕様」にしたがって点灯/消灯します。

【注】 本アプリケーションノートでは、ブロック0に書き換え制御プログラムを配置しておりますので、ブロック0の書き換えは行わないでください。

図1に本アプリケーションノートの概要図を示します。

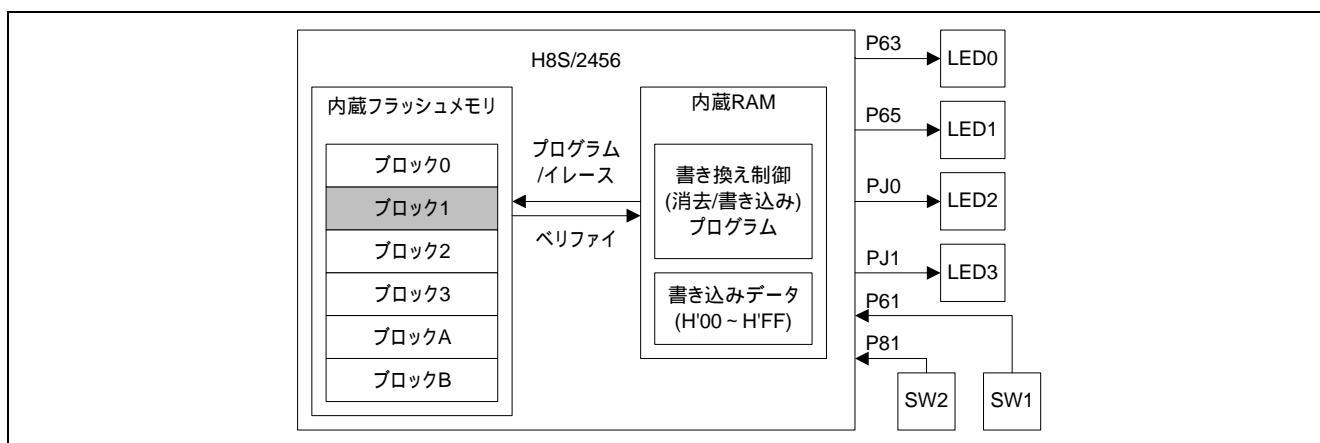


図1 概要図

表1に本アプリケーションノートのLED仕様を示します。

表1 LED仕様

点灯: 、消灯: 、点滅:

状態		LED0 (イレース)	LED1 (プログラム)	LED2 (エラー有無)	LED3 (チェックエラー)
起動時					
イレース動作	開始 (イレース動作中)				
	完了				
	エラー				
	イレースエラー				
	ブランクチェックエラー				
プログラム動作	開始 (プログラム動作中)				
	完了				
	エラー				
		プログラムエラー			
	ベリファイエラー				

2. 適用条件

表 2 に本アプリケーションノートの適用条件を示します。

表 2 適用条件

項目		内容	備考
動作周波数	入力クロック	16MHz	
	システムクロック (φ)	32MHz (16MHz の 2 通倍)	
動作電圧		3.3V	
動作モード		内蔵 ROM 有効拡張モード モード 4 (MD2 = 1, MD1 = 0, MD0 = 0)	
統合開発環境		High-performance Embedded Workshop	Version 4.09.00.007
評価ボード		ルネサス エレクトロニクス製 Renesas Starter Kit for H8S/2456R R0K524568S000BE	
C/C++コンパイラ		H8S,H8/300 C/C++ Compiler	V.6.02.02
最適化リンケージエディタ		Optimizing Linkage Editor	V.9.05.00
コンパイルオプション		-cpu=2600A:24 -nolist -chgincpath -nologo	
リンカオプション		-noprelink -rom=D=R,PFWRMAIN=RFWRMAIN(*) -nomessage -nooptimize -start=PRResetPRG,PlntPRG/0400,P,C,C\$DSEC,C\$B SEC,D/0800,PFWRMAIN/02000,B,R/0FEC000,RFW RMAIN/0FF0000,S/0FFBE00 -nologo -form=stype -exit	

【注】 * 本アプリケーションノートはリンカの ROM 化支援オプション (-rom) を使用して、PFWRMAIN セクションと同サイズの RFWRMAIN セクションを確保し、PFWRMAIN セクション内定義シンボルを RFWRMAIN セクション上のアドレスにリロケーションしています。なお、PFWRMAIN には書き込み制御プログラムを配置しています。

3. 使用機能説明

本アプリケーションノートでは、フラッシュメモリのユーザプログラムモードを使用します。

ユーザプログラムモードでは、CPU がソフトウェアコマンドを実行することにより、フラッシュメモリを書き換えることができます。したがって、ROM ライタなどを使用せずにマイクロコンピュータを基板に実装した状態で、ユーザ ROM を書き換えることができます。

プログラム、ブロックイレースのコマンドは、ユーザプログラムの各ブロック領域のみに対して実行してください。ユーザプログラムモードには、イレースライト 0 モード (EW0 モード) があります。

3.1 EW0 モード

フラッシュメモリコントロールレジスタ 1 (以降、FLMCR1) の FMCMDEN ビットを 1 (ユーザプログラムモードを有効) にするとユーザプログラムモードになり、ソフトウェアコマンドの受け付けが可能となります。

プログラム、イレース動作の制御はソフトウェアコマンドで行います。プログラム、イレースの終了時の状態などはフラッシュメモリステータスレジスタ (以降、FLMSTR レジスタ) またはステータスレジスタで確認できます。

3.2 ソフトウェアコマンド

ソフトウェアコマンドについて次に説明します。ソフトウェアコマンド、データの読み出し、書き込みは 16 ビット単位で、ユーザ ROM、またはデータフラッシュ内の偶数番地に行ってください。コマンドコード書き込み時、下位 8 ビット (D7~D0) は無視されます。

表 3 ソフトウェアコマンド一覧表

ソフトウェア コマンド	第 1 バスサイクル			第 2 バスサイクル			第 3 バスサイクル		
	モード	アドレス	データ (D15~D0)	モード	アドレス	データ (D15~D0)	モード	アドレス	データ (D15~D0)
リードアレイ	ライト	x	H'FFxx						
リードステータス レジスタ	ライト	x	H'70xx	リード	x	SRD			
クリアステータス レジスタ	ライト	x	H'50xx						
プログラム	ライト	WA0	H'41xx	ライト	WA0	WD0	ライト	WA1	WD1
ブロックイレース	ライト	x	H'20xx	ライト	BA	H'D0xx			
ブロックブランク チェック	ライト	x	H'25xx	ライト	BA	H'D0xx			

【記号説明】

SRD: ステータスレジスタデータ (D7~D0)

WA0: 下位ワード書き込み番地 (第 1 バスサイクルのアドレスは、第 2 バスサイクルのアドレスと同一偶数番地にしてください。)

WA1: 上位ワード書き込み番地

WD0: 書き込みデータ下位ワード (16 ビット)

WD1: 書き込みデータ上位ワード (16 ビット)

BA: ブロックの最上位番地 (「表 4 ブロックの最上位番地一覧表」参照)

x: プログラム ROM またはデータフラッシュ内の任意の偶数番地

xx: コマンドコード下位 8 ビット (無視されます)

表 4 ブロックの最上位番地一覧表

内蔵 ROM	ユーザ ROM				データフラッシュ	
ブロック	ブロック 0	ブロック 1	ブロック 2	ブロック 3	ブロック A	ブロック B
イレース指定番地	H'000000	H'010000	H'020000	H'030000	H'F00000	H'F01000

3.2.1 リードアレイ

フラッシュメモリを読むコマンドです。

第1バスサイクルで H'FFxx を書くと、リードアレイモードになります。次のバスサイクル以降で読む番地を入力すると、指定した番地の内容が 16 ビット単位で読めます。

リードアレイモードは、他のコマンドが書かれるまで保持されるので、複数の番地の内容を続けて読めません。

3.2.2 リードステータスレジスタ

ステータスレジスタを読むコマンドです。

第1バスサイクルで H'70xx を書くと、第2バスサイクルでステータスレジスタが読めます（「3.2.6 ステータスレジスタ」参照）。なお、読むときもユーザ ROM またはデータフラッシュ内の偶数番地を読んでください。

3.2.3 クリアステータスレジスタ

ステータスレジスタをクリアするコマンドです。

第1バスサイクルで H'50xx を書くと、FLMSTR レジスタの FMERSF, FMPSF は 0 になります。

3.2.4 プログラム

2ワード単位でフラッシュメモリにデータを書くコマンドです。

第1バスサイクルで H'41xx を書き、第2バスサイクルと第3バスサイクルで書き込み番地にデータを書くことで自動書き込み（データのプログラムとベリファイ）を開始します。第1バスサイクルにおけるアドレス値は、第2バスサイクルで指定する書き込み番地と同一の偶数番地にしてください。

自動書き込み終了は FLMSTR レジスタの FMRDY ビットで確認できます。FMRDY ビットは、自動書き込み期間中は 0（ビジー）、終了後は 1（レディ）になります。

自動書き込み終了後、FLMSTR レジスタの FMPSF ビットで自動書き込みの結果を知ることができます（「3.2.6 ステータスレジスタ」参照）。

すでにプログラムされた番地には追加書き込みはできません。

EW0 モードでは、自動書き込み開始とともにリードステータスレジスタモードとなり、ステータスレジスタが読めます。ステータスレジスタの SR7 ビットは自動書き込み開始とともに 0 となり、終了とともに 1 に戻ります。この場合のリードステータスレジスタモードは、次にリードアレイコマンドを書くまで継続されます。また、自動書き込み終了後、ステータスレジスタを読み出すことにより、自動書き込みの結果を知ることができます。

3.2.5 ブロックイレース

第1バスサイクルで H'20xx、第2バスサイクルで H'D0xx をブロックの最上位番地（「表4 ブロックの最上位番地一覧表」参照）に書くと指定されたブロックに対し、自動消去（イレースとイレースベリファイ）を開始します。

自動消去の終了は、FLMSTR レジスタの FMRDY ビットで確認できます。

FMRDY ビットは、自動消去期間中は0（ビジー）、終了後は1（レディ）になります。

自動消去終了後、FLMSTR レジスタの FMERSF ビットで、自動消去の結果を知ることができます（「3.2.6 ステータスレジスタ」参照）。

EW0 モードでは、自動消去開始とともにリードステータスレジスタモードとなり、ステータスレジスタが読めます。ステータスレジスタの SR7 ビットは自動消去の開始とともに0となり、終了とともに1に戻ります。この場合のリードステータスレジスタモードは、次にリードアレイコマンドを書くまで継続されます。

なお、イレースエラーが発生した場合は、イレースエラーが発生しなくなるまで、クリアステータスレジスタコマンド ブロックイレースコマンドを少なくとも3回実行してください。

3.2.6 ステータスレジスタ

ステータスレジスタは、フラッシュメモリの動作状態やイレース、プログラムの正常、エラー終了などの状態を示すレジスタです。ステータスレジスタの状態は、FLMSTR レジスタの FMRDY, FMPRSF, FMERSF ビットで読めます。

表5 にステータスレジスタを示します。

なお、EW0 モードでは次のときステータスレジスタを読めます。

- リードステータスレジスタコマンドを書いた後、ユーザ ROM、またはデータフラッシュ内の任意の偶数番地を読んだとき
- プログラムコマンド、ブロックイレースコマンド、またはブロックブランクチェックコマンド実行後、リードアレイコマンドを実行するまでの期間に、ユーザ ROM、またはデータフラッシュ内の任意の偶数番地を読んだとき

表5 ステータスレジスタ

ステータス レジスタのビット	FMLSTR レジスタのビット	ステータス名	内容		リセット後の値
			0	1	
SR0 (D0)	—	予約ビット	—	—	—
SR1 (D1)	—	予約ビット	—	—	—
SR2 (D2)	—	予約ビット	—	—	—
SR3 (D3)	—	予約ビット	—	—	—
SR4 (D4)	FMPRSF	プログラムステータス	正常終了	エラー終了	0
SR5 (D5)	FMERSF	イレースステータス	正常終了	エラー終了	0
SR6 (D6)	—	予約ビット	—	—	—
SR7 (D7)	FMRDY	シーケンサステータス	ビジー	レディ	1

【記号説明】

D0～D7: リードステータスレジスタコマンドを実行したときに読み出されるデータバスを示す。

【注】 FMERSF ビット (SR5 ビット)、FMPRSF ビット (SR4 ビット) は、クリアステータスレジスタコマンドを実行すると0になります。

FMERSF ビット (SR5 ビット) または FMPRSF ビット (SR4 ビット) が1の場合、プログラム、ブロックイレース、ブロックブランクチェックコマンドは受け付けられません。

4. 動作説明

本アプリケーションノートは、以下の手順でスイッチを押下することによりイレース動作/プログラム動作します。

- (1) 起動後に書き換え制御プログラムを内蔵RAMへコピーし、書き込みデータ (H'00 ~ H'FF の 256 バイト) を内蔵RAM上に作成します。
- (2) SW1 押下によりブロック 1 をイレース動作し、イレース動作終了後、ブロック 1 の全領域が H'FF に消去されているかブランクチェックします。
- (3) SW2 押下により 256 バイトのデータをブロック 1 の全領域へ繰り返しプログラム動作し、プログラム動作終了後、ブロック 1 の全領域に H'00 ~ H'FF が繰り返し書き込めているかベリファイします。
- (4) (2)に戻ります。

なお、(2)または(3)でエラーが発生した場合、LED を点灯させ無限ループします。

図 2 に動作説明図を示します。

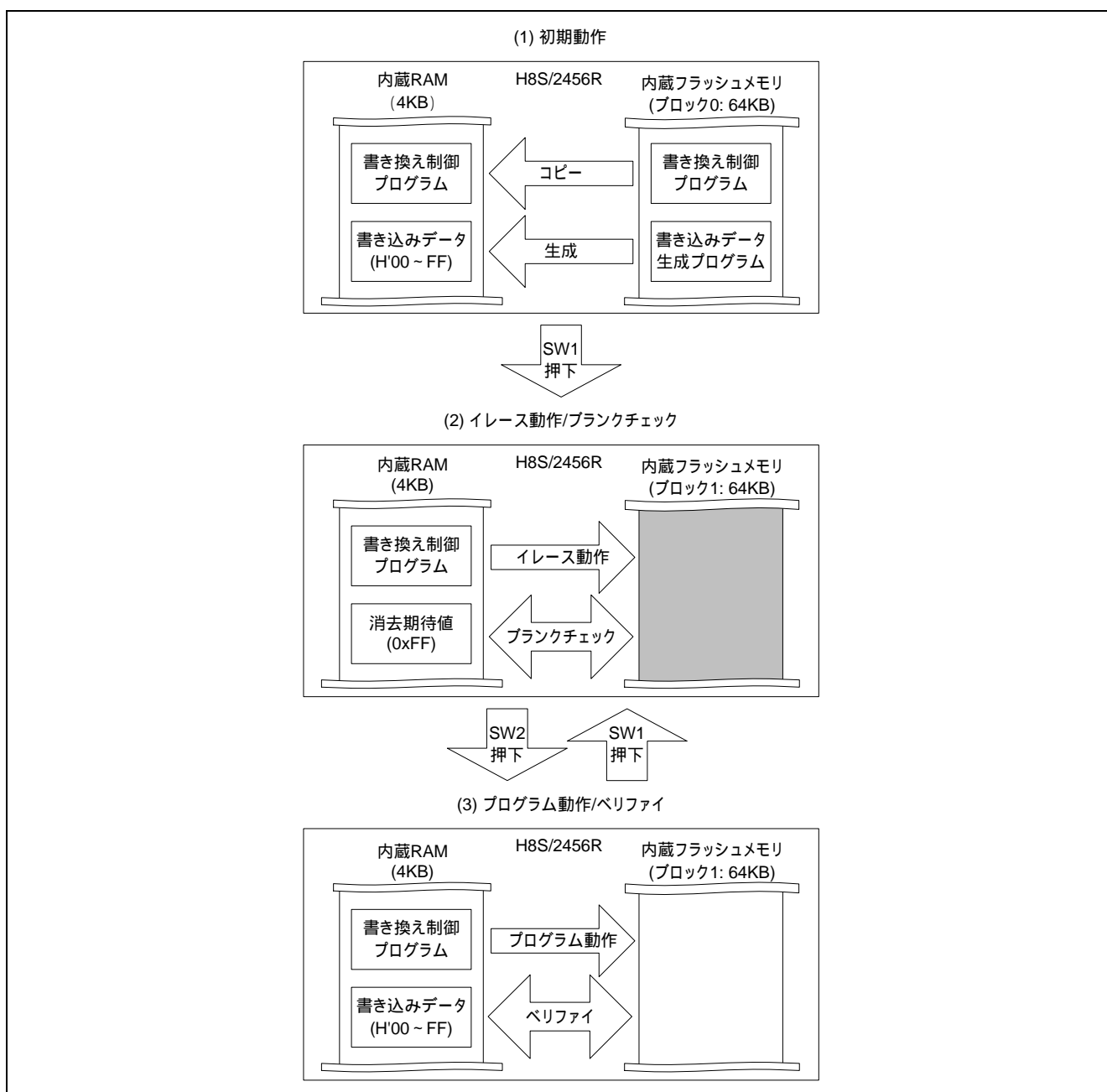


図 2 動作説明図

本アプリケーションノートは、最適化リンカージェネレータの ROM 化支援オプション、および標準ライブラリ関数として提供しているセクションの初期化ルーチン (_INITSCT 関数) を使用して、内蔵 RAM のリロケーションと、内蔵フラッシュメモリに準備した書き換え制御プログラムを内蔵 RAM 上にコピーしています。

ROM 化支援オプションは初期化データ領域の ROM 用、RAM 用領域を確保し ROM セクション内定義シンボルを RAM セクション内アドレスになるようにリロケーションするリンカオプションです。なお、本オプションはリロケーションするのみでコピーしません。したがって、別途書き換え制御プログラムを内蔵 ROM 上から内蔵 RAM 上へコピーする必要があります。

_INITSCT 関数は未初期化データセクションをゼロで初期化し、初期化データセクションの内蔵 ROM 上にある初期化データを内蔵 RAM 上にコピーする関数です。

本アプリケーションノートでは、_INITSCT 関数を使用して書き換え制御プログラムを内蔵 ROM 上から (ROM 化支援オプションによりリロケーションした) 内蔵 RAM 上にコピーしています。

なお、_INITSCT 関数で内蔵 ROM 上から内蔵 RAM 上にコピーするためには、セクション初期化用テーブル (DTBL, BTBL) にコピー元のアドレス、コピー元のデータサイズ、コピー先のアドレスを追加する必要があります。

セクション初期化用テーブル (DTBL, BTBL) は High-performance Embedded Workshop (HEW) にて初期ルーチンファイルを含む実行プログラムを作成するためのプロジェクトを作成すると dbsect.c ファイルに自動で生成されます。

以下に本アプリケーションノートでのセクション初期化用テーブルの追加例を示します。

```
#pragma section $DSEC
static const struct {
    _UBYTE *rom_s;      /* Start address of the initialized data section in ROM */
    _UBYTE *rom_e;      /* End address of the initialized data section in ROM */
    _UBYTE *ram_s;      /* Start address of the initialized data section in RAM */
}DTBL[] = {
    {__sectop("D"), __secend("D"), __sectop("R")},
    {__sectop("PFWMAIN"), __secend("PFWMAIN"), __sectop("RFWMAIN")}, (*)
// {__sectop("$ABS8D"), __secend("$ABS8D"), __sectop("$ABS8R")},
// {__sectop("$ABS16D"), __secend("$ABS16D"), __sectop("$ABS16R")}
};
#pragma section $BSEC
static const struct {
    _UBYTE *b_s;        /* Start address of non-initialized data section */
    _UBYTE *b_e;        /* End address of non-initialized data section */
}BTBL[] = {
    {__sectop("B"), __secend("B")},
// {__sectop("$ABS8B"), __secend("$ABS8B")},
// {__sectop("$ABS16B"), __secend("$ABS16B")}
};
```

(*) 本アプリケーションノートでの追加箇所

ROM 化支援オプションおよび _INITSCT 関数に関する詳細は、6. 参考ドキュメント『開発環境マニュアル』を参照してください。

5. ソフトウェア説明

5.1 記号定数

表 6 記号定数一覧

定数	設定値	内容	使用関数
BLOCK0	0x01	ユーザROMのブロック 0	(未使用)
BLOCK1	0x02	ユーザROMのブロック 1	main
BLOCK2	0x04	ユーザROMのブロック 2	(未使用)
BLOCK3	0x08	ユーザROMのブロック 3	(未使用)
BLOCKA	0x10	データフラッシュのブロック A	(未使用)
BLOCKB	0x20	データフラッシュのブロック B	(未使用)
BLOCK_MAX	6	ブロック数	flash_erase
PWR_ON_RST	0x00	初期設定後の状態	main set_led
ERASING	0x01	イレース動作中の状態	main set_led
ERASE_END	0x02	イレース完了後の状態	main set_led
ERASE_ERROR	0x04	イレースエラー状態	main set_led
ERASE_COMPARE_ERROR	0x05	イレース後の コンペアチェックエラー状態	main set_led
PROGRAMING	0x06	プログラム動作中の状態	main set_led
PROGRAM_END	0x07	プログラム完了後の状態	main set_led
PROGRAM_ERROR	0x09	プログラムエラー状態	main set_led
PROGRAM_COMPARE_ERROR	0x0A	プログラム後の コンペアチェックエラー状態	main set_led
LED0	P6.DR.BIT.B3	RSKのLED0	set_led
LED1	P6.DR.BIT.B5	RSKのLED1	set_led
LED2	PJ.DR.BIT.B0	RSKのLED2	set_led
LED3	PJ.DR.BIT.B1	RSKのLED3	set_led
LED_ON	0	LED点灯	set_led
LED_OFF	1	LED消灯	set_led
MCUFLASH_OK	0	イレース/プログラム正常動作	main flash_erase flash_program block_erase program_256_bytes full_chk
MCUFLASH_NG	-1	イレース/プログラム異常動作	main flash_erase flash_program block_erase program_256_bytes full_chk

5.2 ROM化変数

表 7 ROM化変数

型名	変数名	設定値	内容	使用関数
const unsigned long	erase_address[6]	0x00000000ul, 0x00010000ul, 0x00020000ul, 0x00030000ul, 0x00F00000ul, 0x00F01000ul	内蔵フラッシュメモリの イレース設定アドレス	flash_erase

5.3 RAM変数

表 8 RAM 変数

型名	変数名	内容	使用関数
unsigned char	write_data[256]	内蔵フラッシュメモリ書き換え用データ	main mcu_init
volatile _evenacc ess unsigned short *	wp	コマンド発行時のアドレス	block_erase program_256_bytes full_chk
unsigned char	imask	ユーザプログラムモード遷移前の割り込みマスク 保存	enter_EW0mode exit_EW0mode

5.4 関数一覧

表 9 関数一覧

関数名	説明
PowerOn_Reset	<ul style="list-style-type: none"> 初期設定関数 スタックポインタ (SP) の初期化、割り込みマスクビットの設定、未初期化/ 初期化データの設定、main 関数の呼び出し。
main	<ul style="list-style-type: none"> メイン関数 mcu_init 関数の呼び出し、ブロックのイレース、プログラム動作。
mcu_init	<ul style="list-style-type: none"> MCU 初期化関数 書き込みデータ作成、各レジスタの初期化。
led_set	<ul style="list-style-type: none"> LED 点灯関数 LED 点灯/消灯/点滅動作。
flash_erase	<ul style="list-style-type: none"> 内蔵フラッシュメモリのイレース関数 指定された内蔵フラッシュメモリのブロックを消去。
flash_program_256bytes	<ul style="list-style-type: none"> 内蔵フラッシュメモリのプログラム関数 指定データを内蔵フラッシュメモリの指定アドレスに書き込む。
enter_EW0mode	<ul style="list-style-type: none"> ユーザプログラムモード有効関数 割り込み要求マスクレベルの退避とユーザプログラムモード有効設定。
exit_EW0mode	<ul style="list-style-type: none"> ユーザプログラムモード無効関数 割り込み要求マスクレベルの復帰とユーザプログラムモード無効設定。
block_erase	<ul style="list-style-type: none"> ブロックイレース関数 指定ブロックのイレース動作。
program_256_bytes	<ul style="list-style-type: none"> 256 バイトプログラム関数 指定アドレスへのプログラム動作。
full_chk	<ul style="list-style-type: none"> フルステータスチェック関数 ステータスチェックによる正常/異常チェック。

5.5 関数説明

5.5.1 PowerON_Reset

(1) 機能概要

PowerON_Reset 関数では、スタックポインタ (SP) を初期化し、組み込み関数や標準ライブラリ関数を用いて、割り込みマスクビットの設定や未初期化/初期化データを設定します。そして、main 関数を呼び出します。

(2) 引数

なし

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

なし

(5) フローチャート

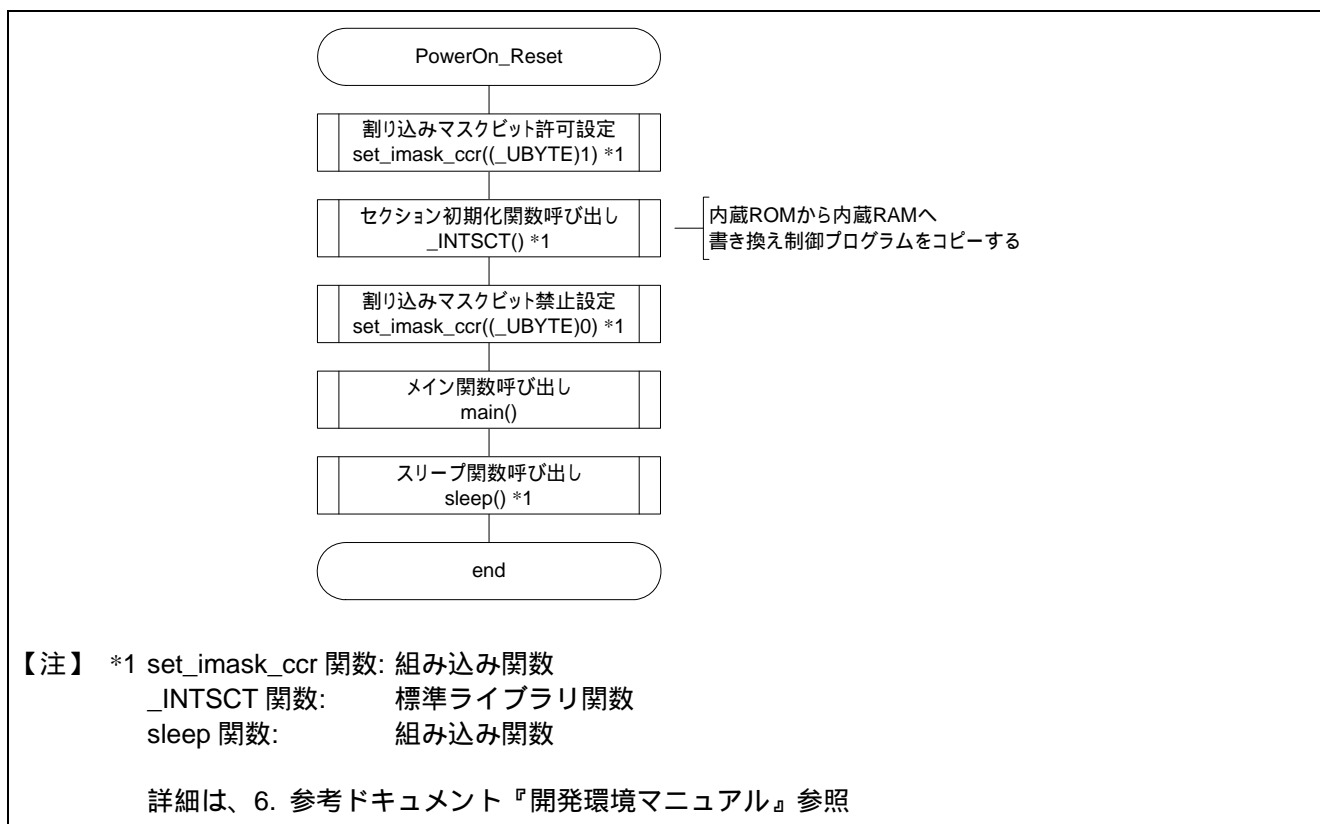


図 3 フローチャート (PowerON_Reset)

5.5.2 main関数

(1) 機能概要

main 関数では、mcu_init 関数を呼び出し、イレース動作、プログラム動作を行います。

(2) 引数

なし

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

• ポート 6 レジスタ (PORT6) ビット数: 8 ビット アドレス: H'FFFFFF55

ビット	ビット名	設定値	R/W	説明
1	P61	—	R/W	このレジスタをリードすると、P6DDR がセットされているビットは、P6DR の値がリードされます。P6DDR がクリアされているビットは端子の状態がリードされます。

• ポート 8 レジスタ (PORT8) ビット数: 8 ビット アドレス: H'FFFFFF57

ビット	ビット名	設定値	R/W	説明
1	P81	—	R/W	このレジスタをリードすると、P8DDR がセットされているビットは、P8DR の値がリードされます。P8DDR がクリアされているビットは端子の状態がリードされます。

(5) フローチャート

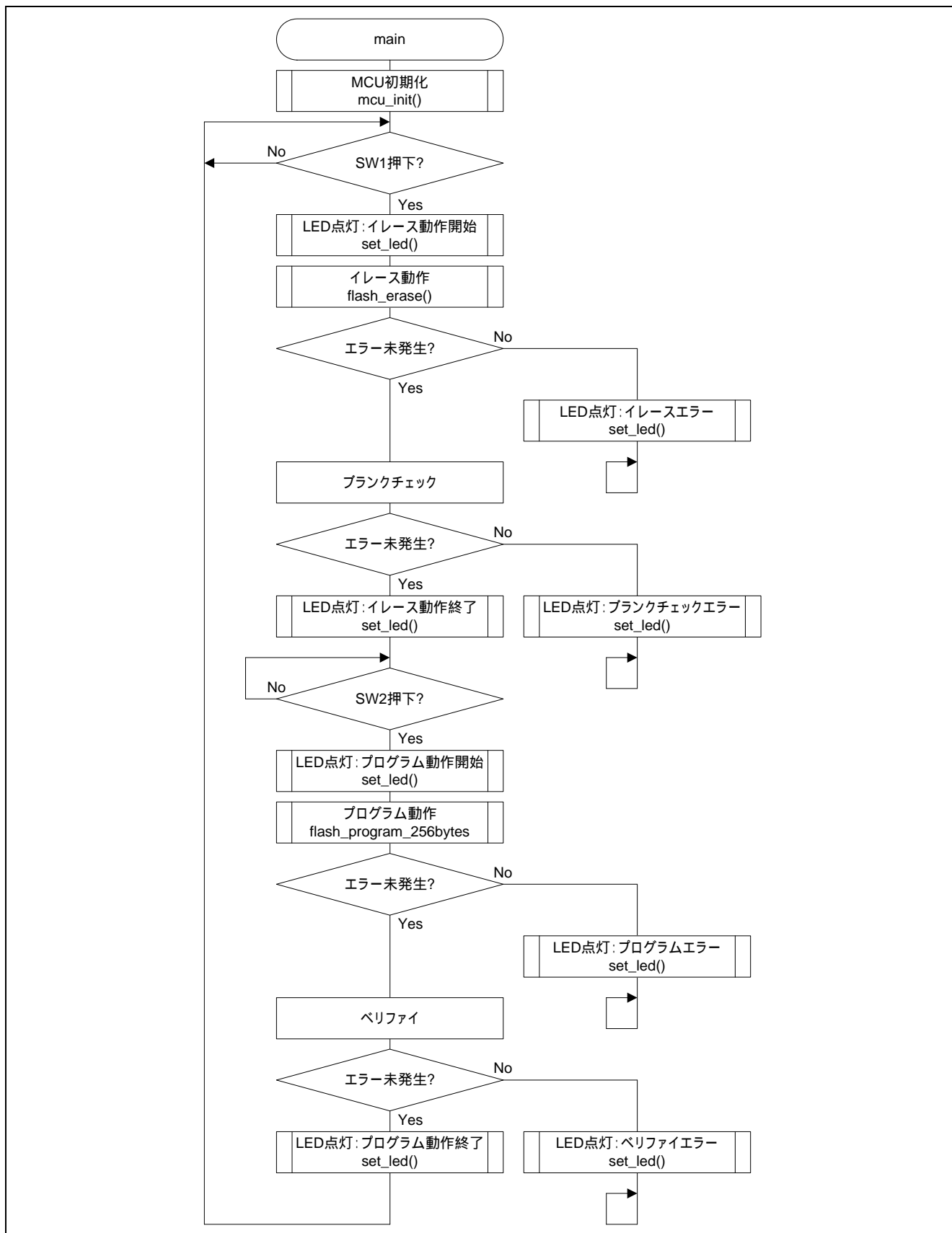


図 4 フローチャート (main)

5.5.3 mcu_init関数

(1) 機能概要

mcu_init 関数では、クロック設定、レジスタの初期化、書き込みデータの生成を行います。

(2) 引数

なし

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

● モードコントロールレジスタ (MDCR) ビット数: 8 ビット アドレス: H'FFFFFF3E

ビット	ビット名	設定値	R/W	説明
2	MDS2	—*	R	モードセレクト 2~0 モード端子 (MD2~MD0) の入力レベルを反映した値 (現在の動作モード) を示しています。これらのビットは MD2~MD0 端子にそれぞれ対応します。これらのビットはリード専用でライトは無効です。MDCR をリードすると、MD2~MD0 端子の入力レベルがこれらのビットにラッチされます。このラッチはリセットで解除されます。
1	MDS1	—*	R	
0	MDS0	—*	R	

【注】 * MD2~MD0 端子の設定により決定されます。

● ポート 6 データディレクションレジスタ (P6DDR) ビット数: 8 ビット アドレス: H'FFFFFFE25

ビット	ビット名	設定値	R/W	説明
5	P65DDR	1	W	汎用入出力ポートの機能が選択されているとき、このビットを 1 にセットすると対応する端子は出力ポートとなり、0 にクリアすると入力ポートになります。
4	P64DDR	0	W	
3	P63DDR	1	W	
2	P62DDR	0	W	
1	P61DDR	0	W	
0	P60DDR	0	W	

● ポート J データディレクションレジスタ (PJDDR) ビット数: 8 ビット アドレス: H'FFFFFF75

ビット	ビット名	設定値	R/W	説明
1	PJ1DDR	1	W	汎用入出力ポートの機能が選択されているとき、このビットを 1 にセットすると対応する端子は出力ポートとなり、0 にクリアすると入力ポートになります。
0	PJ0DDR	1	W	

H8S/2400 シリーズ ユーザプログラムモードによるフラッシュメモリ書き換え(EW0 モード)

- モジュールストップコントロールレジスタ H (MSTPCR) ビット数: 8 ビット アドレス: H'FFFFFF42

ビット	ビット名	設定値	R/W	説明
14	MSTP14	1	R/W	EXDMA コントローラ (EXDMAC) *
13	MSTP13	1	R/W	DMA コントローラ (DMAC)
12	MSTP12	1	R/W	データトランスファコントローラ (DTC)

【注】 * H8S/2454 ではサポートしていません。

- システムクロックコントロールレジスタ (SCKCR) ビット数: 8 ビット アドレス: H'FFFFFF3B

ビット	ビット名	設定値	R/W	説明
3	STCS	1	R/W	周波数通倍率切り替えモード選択 PLLCR レジスタ変更時の動作を選択します。 1: 変更した通倍率は、STC1, STC0 ビット書き換え後に有効

- PLL コントロールレジスタ (PLLCR) ビット数: 8 ビット アドレス: H'FFFFFF45

ビット	ビット名	設定値	R/W	説明
1	STC1	0	R/W	システムクロック PLL 回路周波数通倍率、システムクロック分周器設定 発振器に対するシステムクロックの通倍率、分周比を設定します。 01:×2
0	STC0	1	R/W	

- 割り込みコントロールレジスタ (INTCR) ビット数: 8 ビット アドレス: H'FFFFFF31

ビット	ビット名	設定値	R/W	説明
5	INTM1	1	R/W	割り込み制御選択モード 1, 0 割り込みコントローラの割り込み制御モードを選択します。 10: 割り込み制御モード 2 I2~I0 ビットと IPR で割り込みを制御します。
4	INTM0	0	R/W	

- システムコントロールレジスタ (SYSCR) ビット数: 8 ビット アドレス: H'FFFFFF3D

ビット	ビット名	設定値	R/W	説明
3	FLSHE	1	R/W	フラッシュメモリコントロールレジスタイネーブル フラッシュメモリの制御レジスタ (FLMCR1, FLMDBPR, FLMSTR) の CPU アクセスを制御します。このビットを 1 にセットすると、フラッシュメモリの制御レジスタをリード/ライトすることができます。0 にクリアするとフラッシュメモリの制御レジスタは非選択となります。このとき、フラッシュメモリの制御レジスタの内容は保持されています。フラッシュメモリ版以外は 0 をライトしてください。 1: アドレス H'FFFEB0 ~ H'FFFEB3 のエリアはフラッシュメモリのレジスタを選択

(5) フローチャート



図5 フローチャート (mcu_init)

5.5.4 led_set関数

(1) 機能概要

led_set 関数では、指定された引数に従って LED 点灯/消灯します。

(2) 引数

引数	型名	引数名	引数値	説明
第 1 引数	unsigned char	led_mode	PWR_ON_RST ERASING ERASE_END ERASE_ERROR ERASE_COMPARE_ERROR PROGRAMING PROGRAM_END PROGRAM_ERROR PROGRAM_COMPARE_ERROR	指定された引数に従って、LED 仕様の通りに LED を点灯/消灯します。 LED 仕様は「表 1 LED 仕様」を参照してください。 引数値の説明は「表 6 記号定数一覧」を参照してください。

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

なし

(5) フローチャート

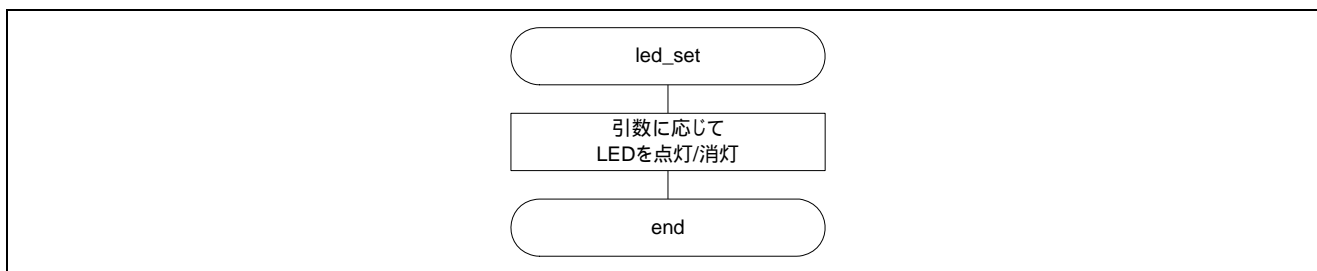


図 6 フローチャート (led_set)

5.5.5 flash_erase関数

(1) 機能概要

flash_erase 関数では、指定された内蔵フラッシュメモリのブロックを消去します。

(2) 引数

引数	型名	引数名	引数値	説明
第 1 引数	unsigned char	block	BLOCK0 BLOCK1 BLOCK2 BLOCK3 BLOCKA BLOCKB	引数に指定されたブロックを消去します。なお、複数ブロック指定も可能です。

(3) 戻り値

型名	戻り値名	戻り値	説明
char	status	MCUFLASH_OK MCUFLASH_NG	MCUFLASH_OK: 消去正常終了 MCUFLASH_NG: 消去エラー発生

(4) 使用内部 I/O レジスタ説明

なし

(5) フローチャート

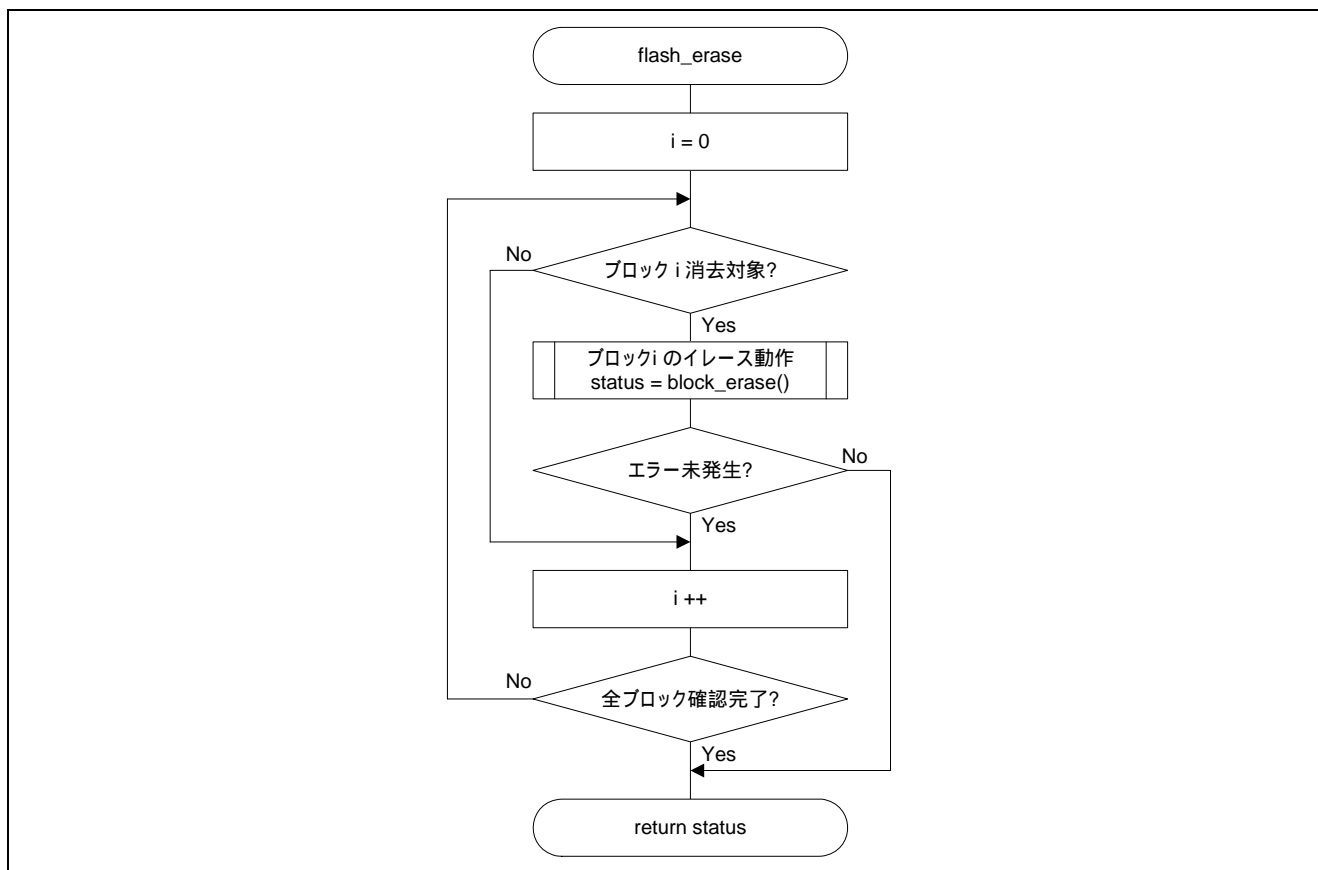


図 7 フローチャート (flash_erase)

5.5.6 flash_program_256bytes関数

(1) 機能概要

flash_program_256bytes 関数では、指定されたデータを内蔵フラッシュメモリの指定されたアドレスに書き込みます。

(2) 引数

引数	型名	引数名	引数値	説明
第 1 引数	unsigned long	addr	書き込み先のアドレス	書き込み先のアドレスを指定
第 2 引数	unsigned short *	data	書き込みデータのポインタ	書き込みデータの保存先ポインタを指定

(3) 戻り値

型名	戻り値名	戻り値	説明
char	status	MCUFLASH_OK MCUFLASH_NG	MCUFLASH_OK: プログラム正常終了 MCUFLASH_NG: プログラムエラー発生

(4) 使用内部 I/O レジスタ説明

なし

(5) フローチャート

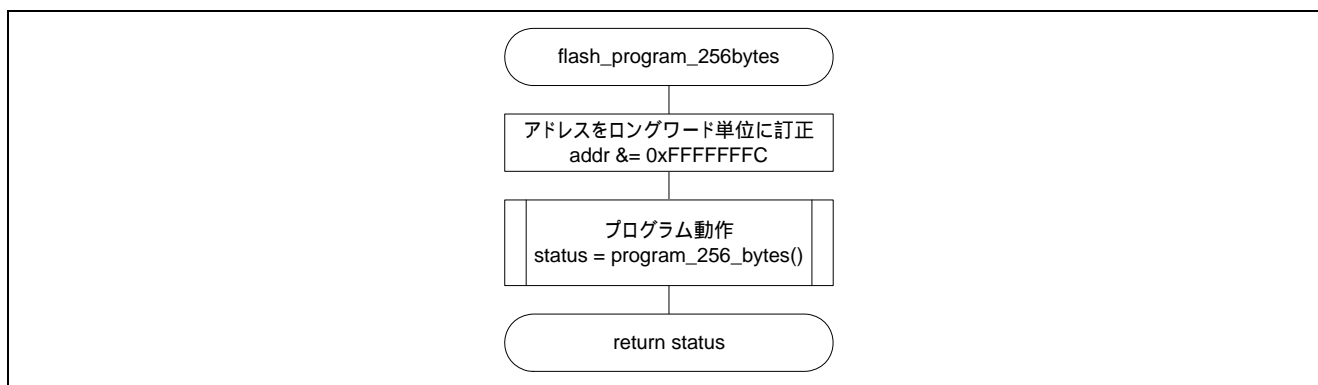


図 8 フローチャート (flash_program_256bytes)

5.5.7 enter_EW0mode関数

(1) 機能概要

enter_EW0mode 関数では、割り込み要求マスクレベル (EXR) の退避とユーザプログラムモードを有効に設定します。

(2) 引数

なし

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

- フラッシュメモリコントロールレジスタ 1 (FLMCR1) ビット数: 8 ビット アドレス: H'FFFFFFE0

ビット	ビット名	設定値	R/W	説明
6	CBIDB	0	R/W	ユーザプログラムモード選択 このビットを 0 (ユーザプログラムモード) にすると、コマンドの受け付けが可能になります。 0: ユーザプログラムモード選択有効
0	FMCMDEN	1	R/W	フラッシュメモリソフトウェアコマンド許可ビット このビットを 1 (ユーザプログラムモード) にすると、コマンドの受け付けが可能になります。 1: フラッシュメモリソフトウェアコマンド許可 1 を書く場合は 0 を書いた後、続けて 1 を書いてください。

(5) フローチャート

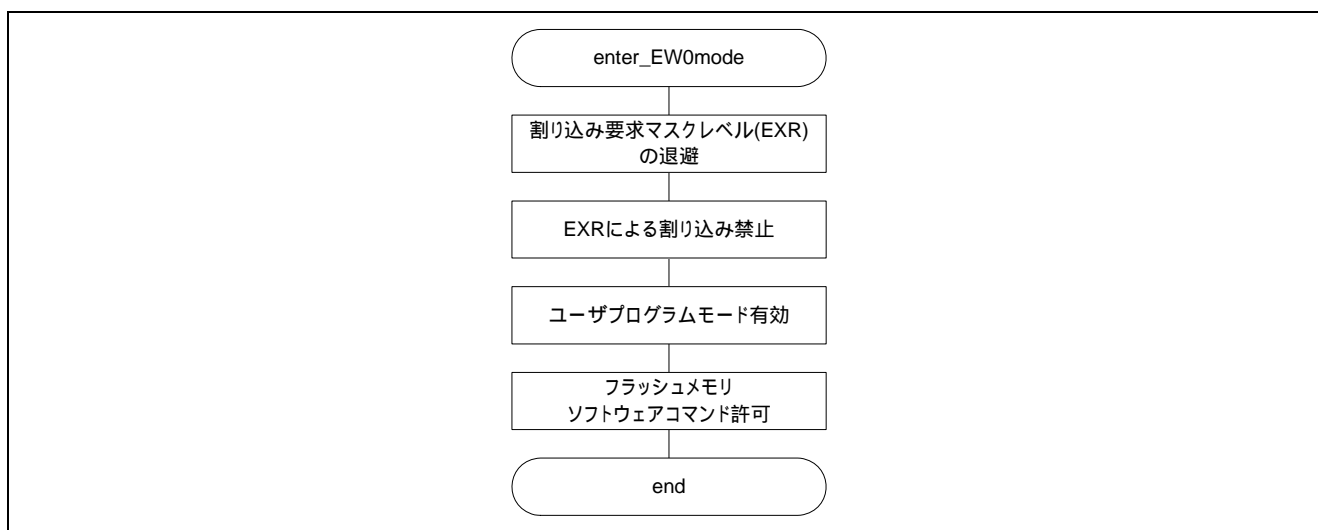


図 9 フローチャート (enter_EW0mode)

5.5.8 exit_EW0mode関数

(1) 機能概要

exit_EW0mode 関数では、割り込み要求マスクレベル (EXR) の復帰とユーザプログラムモードを無効に設定します。

(2) 引数

なし

(3) 戻り値

なし

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

- フラッシュメモリコントロールレジスタ 1 (FLMCR1) ビット数: 8 ビット アドレス: H'FFFFFFE0

ビット	ビット名	設定値	R/W	説明
6	CBIDB	1	R/W	ユーザプログラムモード選択 このビットを 0 (ユーザプログラムモード) にすると、コマンドの受け付けが可能になります。 1: ユーザプログラムモード選択無効
0	FMCMDEN	0	R/W	フラッシュメモリソフトウェアコマンド許可ビット このビットを 1 (ユーザプログラムモード) にすると、コマンドの受け付けが可能になります。 0: フラッシュメモリソフトウェアコマンド禁止 1 を書く場合は 0 を書いた後、続けて 1 を書いてください。

(5) フローチャート

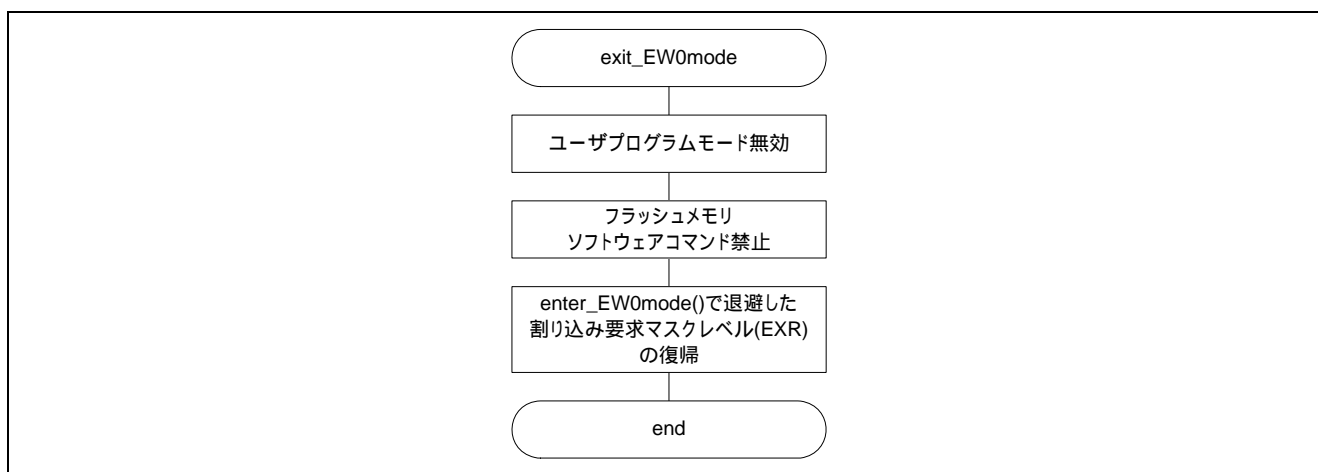


図 10 フローチャート (exit_EW0mode)

5.5.9 block_erase関数

(1) 機能概要

block_erase 関数では、指定ブロック（「表4 ブロックの最上位番地一覧表」参照）のイレース動作を行います。

(2) 引数

引数	型名	引数名	引数値	説明
第1引数	unsigned long	addr	消去先のアドレス	消去先の最上位番地を指定

(3) 戻り値

型名	戻り値名	戻り値	説明
char	status	MCUFLASH_OK MCUFLASH_NG	MCUFLASH_OK: イレース正常終了 MCUFLASH_NG: イレースエラー発生

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

- フラッシュメモリステータスレジスタ (FLMSTR) ビット数: 8 ビット アドレス: H'FFFFFFEB3

ビット	ビット名	設定値	R/W	説明
0	FMRDY	—	R	フラッシュメモリレディ/ビジーステータス 0: ビジー (割り込み、消去実行中) 1: レディ

(5) フローチャート

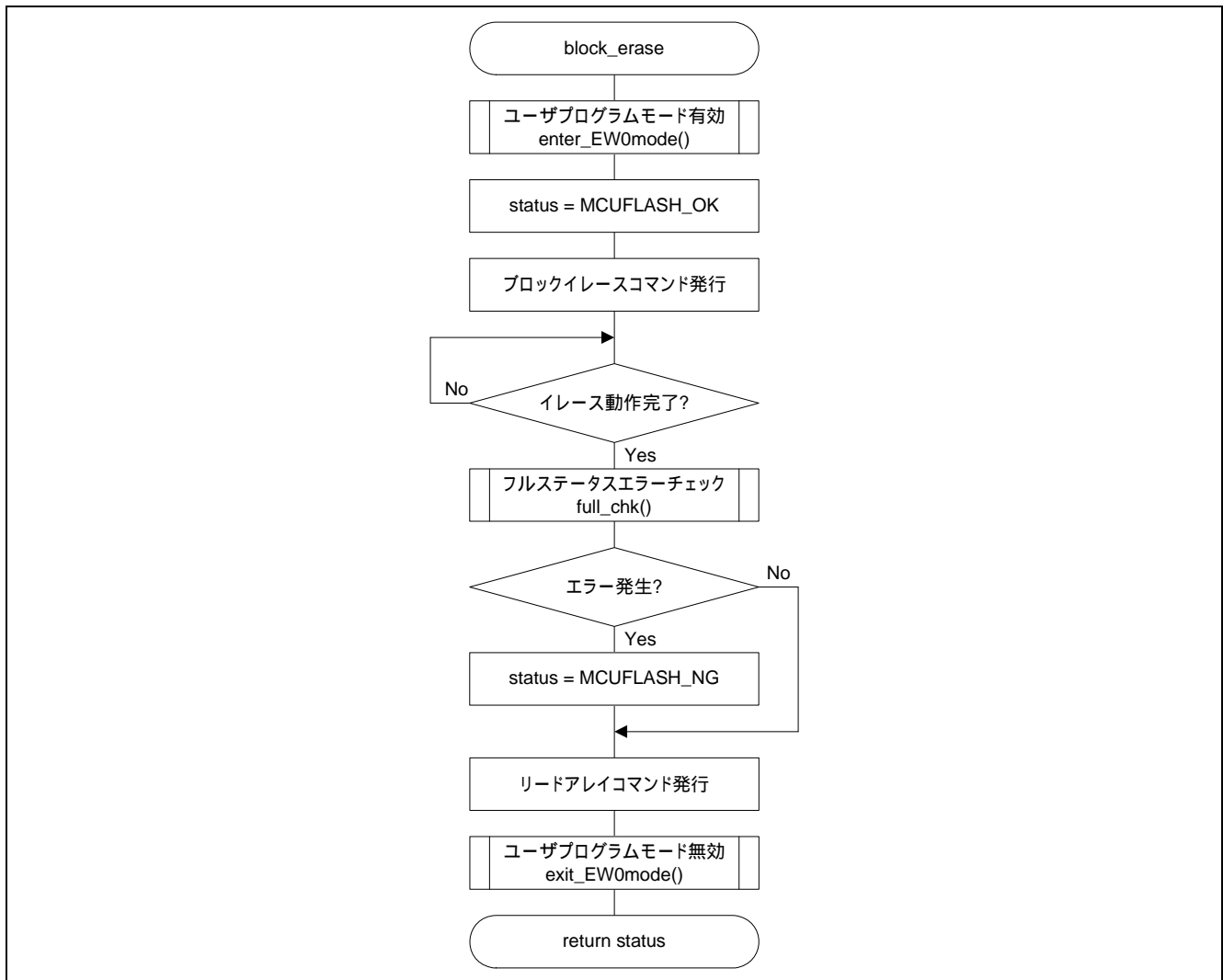


図 11 フローチャート (block_erase)

5.5.10 program_256_bytes関数

(1) 機能概要

program_256_bytes 関数では、指定アドレスへのプログラム動作を行います。

(2) 引数

引数	型名	引数名	引数値	説明
第 1 引数	unsigned long	addr	書き込み先のアドレス	書き込み先のアドレスを指定
第 2 引数	unsigned short *	data	書き込みデータのポインタ	書き込みデータの保存先ポインタを指定

(3) 戻り値

型名	戻り値名	戻り値	説明
char	status	MCUFLASH_OK MCUFLASH_NG	MCUFLASH_OK: イレース正常終了 MCUFLASH_NG: イレースエラー発生

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

- フラッシュメモリステータスレジスタ (FLMSTR) ビット数: 8 ビット アドレス: H'FFFFFFEB3

ビット	ビット名	設定値	R/W	説明
0	FMRDY	—	R	フラッシュメモリレディ/ビジーステータス 0: ビジー (割り込み、消去実行中) 1: レディ

(5) フローチャート

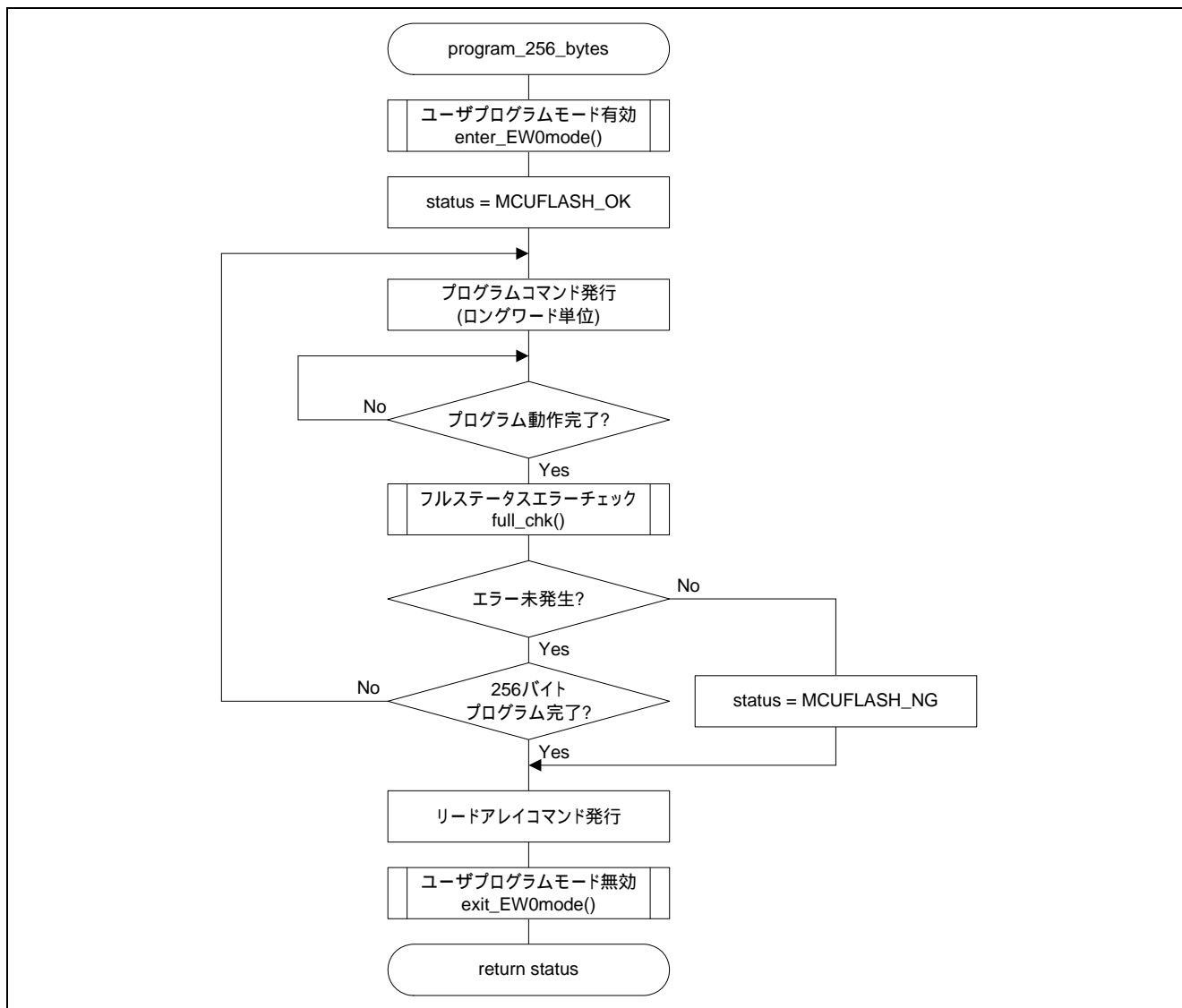


図 12 フローチャート (program_256_bytes)

5.5.11 full_chk関数

(1) 機能概要

full_chk 関数では、ステータスチェックによる正常/異常チェックを行います。

(2) 引数

なし

(3) 戻り値

型名	戻り値名	戻り値	説明
char	status	MCUFLASH_OK MCUFLASH_NG	MCUFLASH_OK: イレース正常終了 MCUFLASH_NG: イレースエラー発生

(4) 使用内部 I/O レジスタ説明

本関数で使用する内部 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において使用している値であり、初期値とは異なります。

- フラッシュメモリステータスレジスタ (FLMSTR) ビット数: 8ビット アドレス: H'FFFFFFEB3

ビット	ビット名	設定値	R/W	説明
5	FMERSF	—	R	イレース・ブランクチェックステータスフラグ 0: 正常終了 1: エラー終了
3	FMPRSF	—	R	プログラムステータスフラグ 0: 正常終了 1: エラー終了

(5) フローチャート

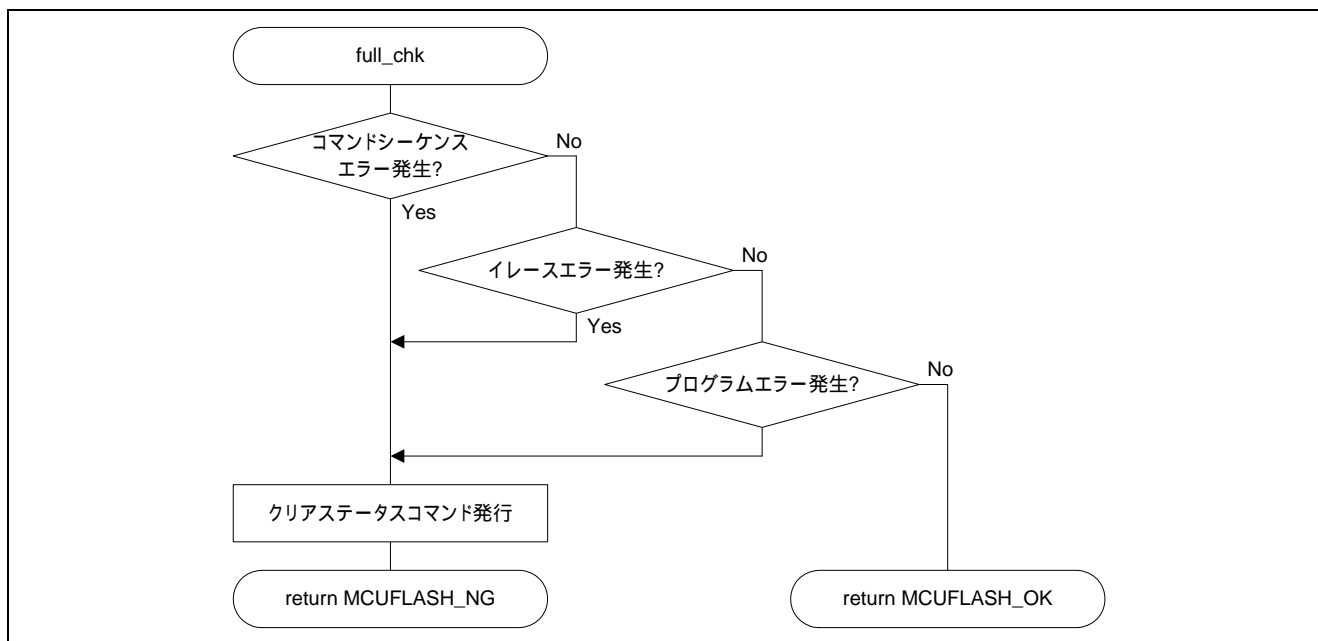


図 13 フローチャート (full_chk)

6. 参考ドキュメント

- ハードウェアマニュアル
H8S/2456, H8S/2456R, H8S/2454 グループ ハードウェアマニュアル (RJJ09B0493)
(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
H8S H8/300 シリーズ C/C++コンパイラパッケージ ユーザーズマニュアル (RJJ10J2552)
(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/inquiry>

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	2010.06.07	—	初版発行
1.01	2010.06.15	3	評価ボードを修正
1.02	2011.09.07	2	注意文の追加
		3	適用条件の修正
		10	volatile_evenaccess の宣言追加
		16	マニュアル改訂によるフローチャート、表の修正
		—	ソースファイルを改訂 (flash.c, flash_main.c)

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、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

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て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。



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・電話番号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 エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■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/inquiry>