

M16C/63,64,64A,64C,65,65C,6C,5LD,56D,5L,56,5M,57 グループ

R01AN2770JJ0100

Rev.1.00

PC ターミナルソフトウェアを用いた内蔵フラッシュメモリ書き換え

2015.07.23

## 要旨

本アプリケーションノートでは、PC ターミナルソフトを用いてマイコンの内蔵フラッシュメモリ（プログラム ROM1）を CPU 書き換えモード（EW0 モード）で書き換える方法を説明します。ただし、M16C/65 グループはプログラム ROM1 が 512K バイト以下の製品が対象です。

## 対象デバイス

- ・ M16C/63 グループ
- ・ M16C/64 グループ
- ・ M16C/64A グループ
- ・ M16C/64C グループ
- ・ M16C/65 グループ（プログラム ROM1 が 512K バイト以下の製品）
- ・ M16C/65C グループ
- ・ M16C/6C グループ
- ・ M16C/5LD グループ
- ・ M16C/56D グループ
- ・ M16C/5L グループ
- ・ M16C/56 グループ
- ・ M16C/5M グループ
- ・ M16C/57 グループ

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 .....	4
3. 関連アプリケーションノート .....	4
4. ハードウェア説明 .....	5
4.1 使用端子一覧 .....	5
5. ソフトウェア説明 .....	6
5.1 動作概要 .....	7
5.2 ファイル構成 .....	9
5.3 定数一覧 .....	9
5.4 変数一覧 .....	11
5.5 関数一覧 .....	12
5.6 関数仕様 .....	13
5.7 フローチャート.....	19
5.7.1 メイン処理 .....	19
5.7.2 CPU初期設定 .....	20
5.7.3 プログラマブル入出力ポート初期設定.....	21
5.7.4 周辺機能初期設定 .....	22
5.7.5 CPUクロック低速化 .....	23
5.7.6 CPUクロック高速化 .....	23
5.7.7 1バイト受信 .....	24
5.7.8 ブロック番号受信 .....	25
5.7.9 フラッシュメモリ消去 .....	26
5.7.10 フラッシュメモリ書き込みデータ受信.....	27
5.7.11 フラッシュメモリ書き込みデータ生成.....	28
5.7.12 フラッシュメモリ書き込み.....	29
5.7.13 ブロック消去関数 .....	30
5.7.14 ブロック書き込み関数.....	32
5.7.15 フルステータスチェック .....	33
5.7.16 制御プログラム転送.....	34
5.7.17 RAM転送処理.....	35
5.7.18 メッセージ送信 .....	36
6. サンプルコード.....	37
7. 参考ドキュメント .....	37

## 1. 仕様

本アプリケーションノートでは、M16C/65 を PC ターミナルソフトから制御し内蔵フラッシュメモリ（プログラム ROM1 領域）の消去および書き換えをブロック単位に行います。

内蔵フラッシュメモリは CPU 書き換えモード (EW0 モード) に設定され PC ターミナルソフトから消去および書き換えたいブロック番号を選択、書き込みデータを送信します。

M16C/65 は PC ターミナルソフトから指定のブロック番号、書き込みデータを受信し動作を実行します。なおブロック単位での消去を実行した場合、合わせてブランクチェックも実行します。

表 1.1 に使用する周辺機能と用途を、図 1.1 に動作概要を示します。

表 1.1 使用する周辺機能と用途

周辺機能	用途
シリアルインタフェース/クロック非同期形シリアルI/Oモード (チャンネル0)	PC ターミナルソフトと RSK ボード の通信
プログラマブル入出力ポート	LED の点灯/消灯

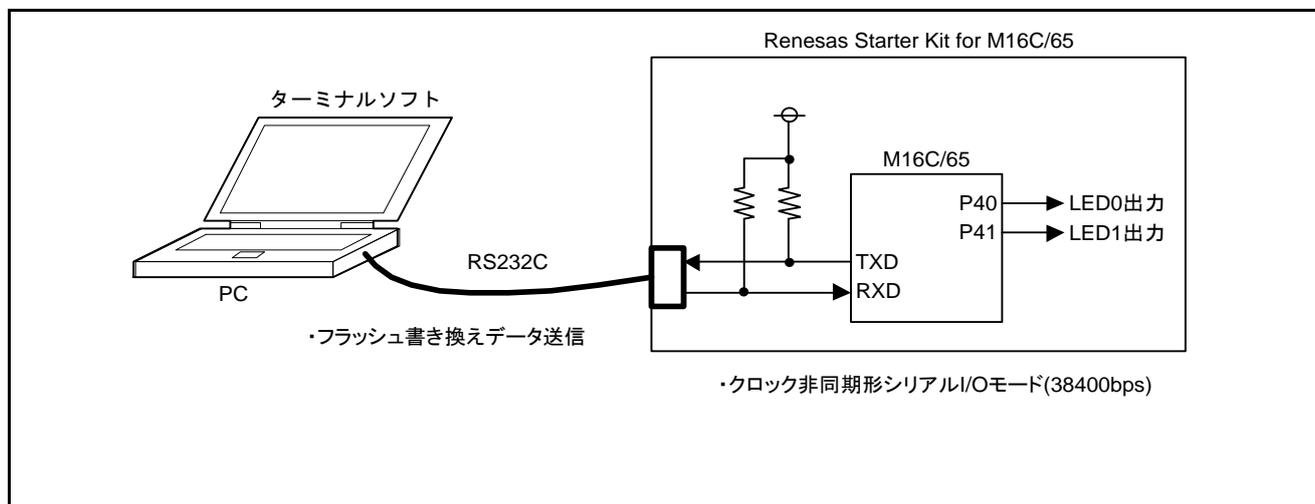


図 1.1 使用例

## 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	R5F3650TDFB (M16C/65 グループ)
動作周波数	メインクロック: 8MHz (水晶発振子) 周辺モジュールクロック: 8MHz
動作電圧	5V
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded Workshop V.4.09.01
C コンパイラ	ルネサスエレクトロニクス製 NC30 V.5.45 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。
SFR のバージョン	sfr65.h Version 2.00 (2010-09-30) [Hardware Manual Revision : 2.00]
エンディアン	リトルエンディアン
プロセッサモード	シングルチップモード
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 M16C/65(製品型名: R0K53650MC000BE)
使用ツール	ターミナルソフト

## 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。

M16C グループ CPU 書き換えモード (EW0 モード) の使用例 Rev.1.04 (R01AN0172JJ)

上記アプリケーションノートの初期設定関数を、本アプリケーションノートのサンプルコードで使用しています。Rev は本アプリケーションノート作成時点のもので、

最新版がある場合、最新版に差し替えて使用してください。最新版はルネサスエレクトロニクスホームページで確認および入手してください。

## 4. ハードウェア説明

### 4.1 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
P40/A16	出力	LED0 点灯（フラッシュメモリ消去正常終了）
P41/A17	出力	LED1 点灯（フラッシュメモリ書き込み正常終了）
P62/RXD0SCL0	入力	UART0 の入力端子（PC ターミナルソフトからの入力）
P63/TXD0SDA0	出力	UART0 の出力端子（PC ターミナルソフトへの出力）

## 5. ソフトウェア説明

本アプリケーションノートに関連する周辺機能を以下に示します。基本的な内容は、M16C/65 グループのユーザーズマニュアル／ハードウェア編（r01uh0135jj0210\_M16C/65.pdf）を参照ください。

### 1. クロック非同期型シリアル I/O モード

PC ターミナルソフトとの M16C/65 の通信は、クロック非同期型シリアル I/O モード（UART0）を使用しています。以下に UART0 の設定を示します。

〈 UART0 〉

使用モード : クロック非同期型シリアル I/O モード  
通信ビットレート : 38400bps  
CTS／RTS : 未使用  
ストップビット : 1ストップビット  
パリティ : なし  
データ長 : 8ビット

### 2. プログラマブル入出力ポート（I/O ポート）

Renesas Starter Kit for M16C/65 ボードには、LED が搭載されております。LED はプログラマブル入出力ポートに接続され、対応するポートが Low にセットされると点灯します。

フラッシュメモリへの消去が正常終了時に LED0、書き込み終了時に LED1 を点灯させます。

表 4.1 にプログラマブル入出力ポート端子と LED の対応表を示します。

表 4.1 プログラマブル入出力ポート端子と LED 対応表

フラッシュメモリへの動作	入出力ポート端子	LED(色)
消去が正常に終了	P40	LED0(Green)
書き込みが正常に終了	P41	LED1(Orange)

## 5.1 動作概要

内蔵フラッシュメモリ（プログラム ROM1 領域）書き換え動作の概要を以下に示します。

またサンプルプログラムと PC ターミナルソフトの制御に分けた主な動作説明を図 6.1 に示します。

- (1) PC ターミナルソフトを起動し RS232C 経由で RSK ボードと接続。
- (2) リセット解除後、CPU 初期設定および周辺機能初期設定。
- (3) PC ターミナルソフトから消去するブロック番号入力待ち状態。  
消去するブロックで選択可能な番号は 1~7 です。表 6.1 を参照ください。

表 6.1 消去可能なブロック番号

ブロック番号	内容
1	ブロック 1 : 64K バイト (先頭アドレス : 0xE0000h)
2	ブロック 2 : 64K バイト (先頭アドレス : 0xD0000h)
3	ブロック 3 : 64K バイト (先頭アドレス : 0xC0000h)
4	ブロック 4 : 64K バイト (先頭アドレス : 0xB0000h)
5	ブロック 5 : 64K バイト (先頭アドレス : 0xA0000h)
6	ブロック 6 : 64K バイト (先頭アドレス : 0x90000h)
7	ブロック 7 : 64K バイト (先頭アドレス : 0x80000h)

【注】上記以外のブロック番号を入力しても動作しません。

- (4) 消去するブロック番号が入力されると消去およびブランクチェックを実行します。  
フラッシュメモリのブロック消去が正常終了すると LED0 が点灯します。(注 1)
- (5) PC ターミナルソフトからの書き込みデータ入力待ちとなります。
- (6) PC ターミナルソフトから書き込みデータを入力します。入力データは 2 ワード (4 バイト) 単位ごとに入力可能です。ブロックの先頭アドレスから 4 バイト単位で書き込まれます。書き込まれるブロックは(3)で指定したブロックです。  
入力データの最後に"END"を必ず入力してください。
- (7) プログラムコマンドを発行しフラッシュメモリへの書き込みを実行します。  
正常に書き込みが終了すると LED1 が点灯します。(注 1)

注 1 本アプリケーションノートでは、エラー発生時にはメッセージの送信のみを行っています。  
必要に応じてエラー処理を追加してください。

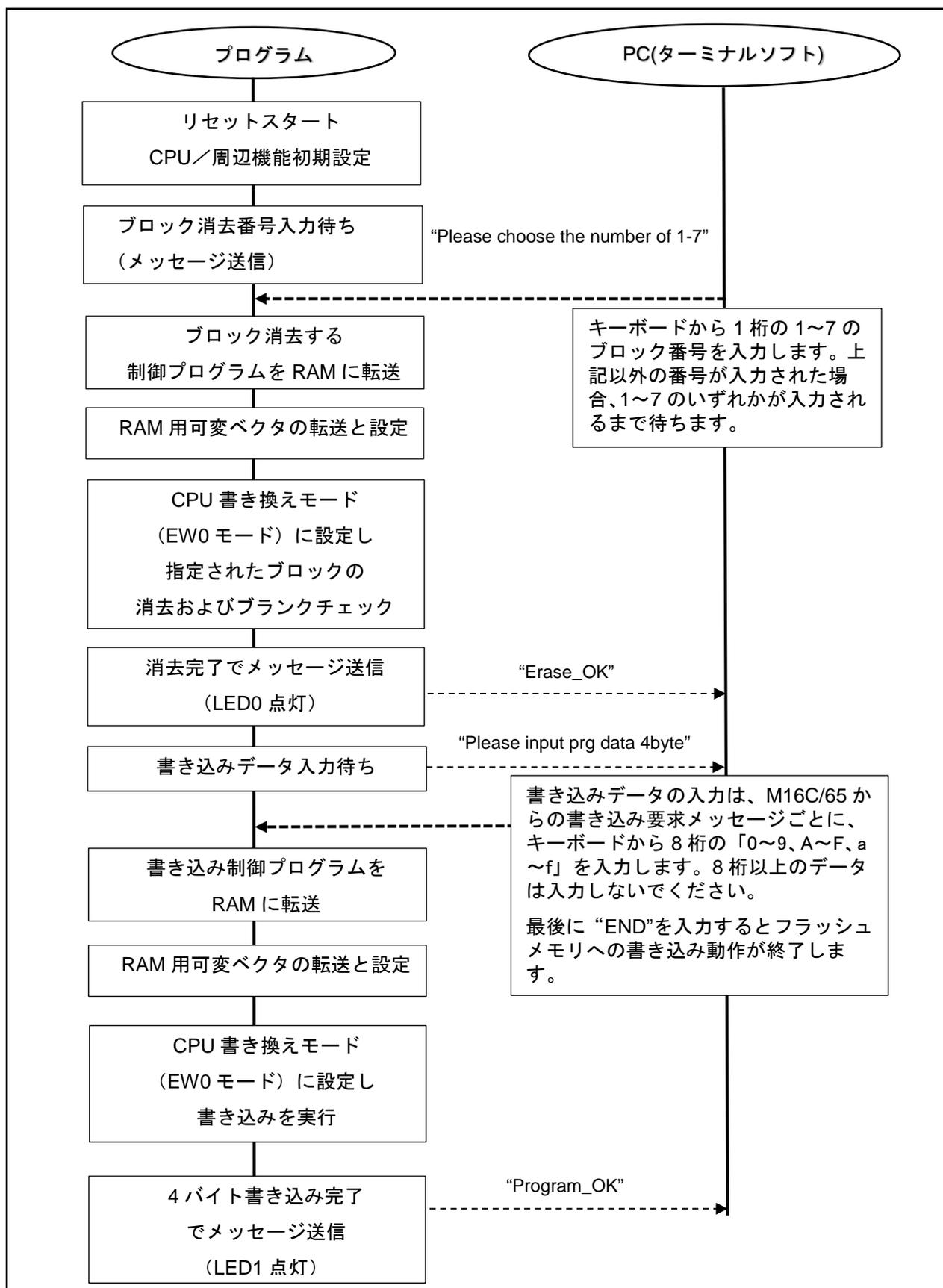


図 6.1 プログラムと PC ターミナルソフトの主な動作概要

## 5.2 ファイル構成

表 6.2 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は除きます。

表 6.2 サンプルコードで使用するファイル

ファイル名	概要
r01an2770_src.c	メイン処理

## 5.3 定数一覧

表 6.3 にサンプルコードで使用する定数を示します。

表 6.3 サンプルコードで使用する定数(r01an2770\_src.c)

定数名	設定値	内容
BLOCK_PGROM1_1_END	EFFFEh	ブロック 1 の最下位番地 (偶数番地)
BLOCK_PGROM1_2_END	DFFFEh	ブロック 2 の最下位番地 (偶数番地)
BLOCK_PGROM1_3_END	CFFFEh	ブロック 3 の最下位番地 (偶数番地)
BLOCK_PGROM1_4_END	BFFFEh	ブロック 4 の最下位番地 (偶数番地)
BLOCK_PGROM1_5_END	AFFFEh	ブロック 5 の最下位番地 (偶数番地)
BLOCK_PGROM1_6_END	9FFFEh	ブロック 6 の最下位番地 (偶数番地)
BLOCK_PGROM1_7_END	8FFFEh	ブロック 7 の最下位番地 (偶数番地)
BLOCK_PGROM1_1	E000h	ブロック 1 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_2	D000h	ブロック 2 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_3	C000h	ブロック 3 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_4	B000h	ブロック 4 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_5	A000h	ブロック 5 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_6	9000h	ブロック 6 のプログラムコマンド発行時の先頭番地
BLOCK_PGROM1_7	8000h	ブロック 7 のプログラムコマンド発行時の先頭番地
BK1	1	ブロック番号
BK2	2	ブロック番号
BK3	3	ブロック番号
BK4	4	ブロック番号
BK5	5	ブロック番号
BK6	6	ブロック番号
BK7	7	ブロック番号
COMPLETE	00h	正常動作の定数
PROGRAM_ERR	01h	プログラム動作エラー定数
ERASE_ERR	02h	消去動作エラー定数
CMD_SEQ_ERR	04h	コマンドシーケンスエラー定数
RECEIVE_ERROR	08h	フラッシュメモリへの書き込みデータ受信エラー
END_T	10h	"END"コマンドを受信した
END_F	20h	"END"コマンドを受信していない
ERS_CMD	0020h	消去コマンド発行時の値 (第 1 バスサイクル)
BLC_CMD	0025h	ブランクチェックコマンド発行時の値 (第 1 バスサイクル)
PRG_CMD	0041h	プログラムコマンド発行時の値 (第 1 バスサイクル)

READ_CMD	00FFh	リードアレイコマンド
CNF_CMD	00D0h	第 2 バスサイクルコマンド
CLR_STS_CMD	0050h	クリアステータスレジスタコマンドの値 (第 1 バスサイクル)
DATA_SIZE	8	シリアルデータ受信サイズ
LED_ON	0	LED 出力データ: 点灯
LED_OFF	1	LED 出力データ: 消灯
MAX_COUNT	3E80h	ブロック書き込み回数最大値(16000)
BIT_RATE	38400	38400bps
XIN_CLOCK_MHz	8000000	8MHz
XIN_BRG	(*1)	ビットレート
RP_RAM	(ram_p)	転送先のプログラム領域
RP_PROGRAM	(*2)	フラッシュメモリ書き込みプログラムの先頭アドレス

\*1 :  $(XIN\_CLOCK\_MHz / BIT\_RATE / 16) - 1$

\*2 :  $((\text{unsigned short far})ram\_p + ((\text{unsigned short far})program\_write\_ew0 - (\text{unsigned short far})block\_erase\_ew0))$

## 5.4 変数一覧

表 6.4 にグローバル変数を、表 6.5 に const 型変数を示します。

表 6.4 グローバル変数

型	変数名	内容	使用関数
unsigned char	block_number	書き込み/消去するブロック番号	main
unsigned char	ram_p[512]	書き込み/消去プログラム転送エリア	main
unsigned char	prg_data[8]	書き込みデータ受信バッファ	main
unsigned char	fwrite_data[8]	書き込みデータ格納バッファ	main
unsigned short	write_data_1st	書き込みデータ (2 バイト)	main
unsigned short	write_data_2nd	書き込みデータ (2 バイト)	main
unsigned short far*	prg_addr	書き込み先アドレス格納	main
unsigned short	program_cnt	書き込み回数カウント	main

表 6.5 const 型変数

型	変数名	内容	使用関数
const unsigned char	mess_bk_number	ブロック選択メッセージ	main
const unsigned char	mess_bk_number_set1	ブロック 1 選択時のメッセージ	main
const unsigned char	mess_bk_number_set2	ブロック 2 選択時のメッセージ	main
const unsigned char	mess_bk_number_set3	ブロック 3 選択時のメッセージ	main
const unsigned char	mess_bk_number_set4	ブロック 4 選択時のメッセージ	main
const unsigned char	mess_bk_number_set5	ブロック 5 選択時のメッセージ	main
const unsigned char	mess_bk_number_set6	ブロック 6 選択時のメッセージ	main
const unsigned char	mess_bk_number_set7	ブロック 7 選択時のメッセージ	main
const unsigned char	mess_uart_rx_err	UART 受信エラーのメッセージ	main
const unsigned char	mess_erase_ok	ブロック消去終了時のメッセージ	main
const unsigned char	mess_erase_err	ブロック消去エラー時のメッセージ	main
const unsigned char	mess_blank_check_err	ブランクチェックエラー時のメッセージ	main
const unsigned char	mess_prg_data	ブロック書き込み入力データ要求メッセージ	main
const unsigned char	mess_senddata_err	書き込みデータ送信エラーメッセージ	main
const unsigned char	mess_program_ok	ブロック書き込み終了時のメッセージ	main
const unsigned char	mess_program_err	ブロック書き込みエラー時のメッセージ	main
const unsigned char	mess_prg_end	ブロック書き込み動作終了メッセージ	main

## 5.5 関数一覧

表 6.6 に関数を示します。

表 6.6 関数

関数名	概要	記載ファイル
main	メイン処理	main.c
mcu_init	CPU 初期設定	main.c
port_init	プログラマブル入出力ポート初期設定	main.c
peripheral_init	周辺機能初期設定	main.c
cpu_slow	CPU クロック低速化	main.c
cpu_fast	CPU クロック高速化	main.c
rev_byte	UART でのバイトデータ受信	main.c
rev_bk_number	ブロック番号受信	main.c
flash_erase	フラッシュメモリ消去	main.c
rev_data_endchk	ブロック書き込みデータ受信および"END"チェック	main.c
flash_wdata_make	フラッシュメモリ書き込みデータ生成	main.c
flash_prg	フラッシュメモリ書き込み	main.c
block_erase_ew0	ブロック消去関数	main.c
program_write_ew0	ブロック書き込み関数	main.c
full_status_check	フルステータスチェック	main.c
send_to_ram	制御プログラム転送	main.c
asm_smovf	RAM 転送処理	main.c
send_message	メッセージ送信	main.c
dummy	ダミー関数	main.c

## 5.6 関数仕様

サンプルコードの関数仕様を示します。

---

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	初期設定後、PC(ターミナルソフト)からブロック番号を受信し内蔵フラッシュメモリの消去およびブランクチェック、PC(ターミナルソフト)から書き込みデータを受信しフラッシュメモリへ書き込みます。
引数	なし
リターン値	なし

---

mcu_init	
概要	CPU 初期設定
ヘッダ	なし
宣言	void mcu_init(void)
説明	プロセッサモード、システムクロックを設定します。
引数	なし
リターン値	なし

---

port_init	
概要	ポート初期設定
ヘッダ	なし
宣言	void port_init(void)
説明	ポート（P40/P41）の初期設定で LED0、LED1 を消灯。
引数	なし
リターン値	なし

---

peripheral_init	
概要	周辺機能初期設定
ヘッダ	なし
宣言	void peripheral_init(void)
説明	UART0 の送受信設定および起動を行います。
引数	なし
リターン値	なし

---

---

cpu_slow	
概要	CPU 低速化処理
ヘッダ	なし
宣言	void cpu_slow(void)
説明	メインクロックの分周を CM06=“0”、CM17、CM16 = “01”（2分周モード）、PM17=“1”（1ウェイト）の設定を行います。
引数	なし
リターン値	なし

---

cpu_fast	
概要	CPU 高速化処理
ヘッダ	なし
宣言	void cpu_fast(void)
説明	メインクロックの分周を CM06=“0”、CM17、CM16 = “00”（分周なし）、PM17=“0”（ウェイトなし）の設定を行います。
引数	なし
リターン値	なし

---

rev_byte	
概要	1バイト受信
ヘッダ	なし
宣言	unsigned char rev_byte(void)
説明	PC(ターミナルソフト)から UART0 経由で 1バイトのデータを受信します。 受信エラーが発生した場合、"UART Receive Error Please check"メッセージが送信されサンプルコードは停止します。
引数	なし
リターン値	result（受信データ）

---

rev_bk_number	
概要	フラッシュメモリ消去/書き込みブロック番号受信
ヘッダ	なし
宣言	unsigned char rev_bk_number(void)
説明	PC(ターミナルソフト)から入力される消去/書き込みブロック番号を ASCII から Binary 変換し判別します。1~7のいずれかの値が入力されるまで受信します。
引数	なし
リターン値	select_number(1~7)

---

flash_erase	
概要	フラッシュメモリブロック消去
ヘッダ	なし
宣言	void flash_erase(unsigned char snum)
説明	選択されたブロック番号 (1~7) の内蔵フラッシュメモリのブロック消去およびブランクチェックを実行します。ブロック消去エラーおよびブランクチェックエラーが発生した場合、" Erase_Error メッセージ"が送信されサンプルコードは停止します。
引数	char snum                      選択されたブロック番号
リターン値	なし
rev_data_endchk	
概要	フラッシュメモリ書き込みデータ受信
ヘッダ	なし
宣言	unsigned char rev_data_endchk(void)
説明	書き込みデータを1バイトごとに8回受信し配列 (prg_data[8]) に格納します。3バイト受信した時点でENDが入力されたかチェックしENDが入力された場合、受信動作は終了します。
引数	なし
リターン値	END_T : ENDが入力された END_F : ENDは入力されていない
flash_wdata_make	
概要	フラッシュメモリ書き込みデータ生成
ヘッダ	なし
宣言	unsigned char flash_wdata_make(unsigned char snum)
説明	rev_data_endchk関数で受信したデータを元にフラッシュメモリへの書き込みデータを作成します。また、受信したブロック番号から書き込みアドレスを生成します。 ・書き込みアドレス生成 PC(ターミナルソフト)から受信したブロック番号に従ってブロックの先頭アドレス(偶数番地)を設定します。 ・書き込みデータ生成 PC(ターミナルソフト)から受信された8桁(4バイト)のデータをASCII => Binary変換し配列に格納します。配列からデータを取り出し、フラッシュメモリ書き込み時の2バイトデータ(write_data_1stおよびwrite_data_2nd)を生成します。 書き込みデータにエラーが発生した場合、" Program_data_Error "メッセージが送信されサンプルコードは停止します。
引数	unsigned char snum              選択されたブロック番号
リターン値	COMPLETE : 正常書き込み終了 RECEIVE_ERROR : 書き込みデータが「0~9、A~F、a~f」以外の場合

---

**flash\_prg**

---

概要	フラッシュメモリのブロックへ書き込み
ヘッダ	なし
宣言	void flash_prg (void)
説明	選択されたブロック番号（1～7）へ PC(ターミナルソフト)から入力されたデータを書き込みます。書き込みエラーが発生した場合、" Program_Error "メッセージが送信されサンプルコードは停止します。
引数	なし
リターン値	なし

---

**block\_erase\_ew0**

---

概要	CPU 書き換えモード（EW0 モード）でのブロック消去
ヘッダ	なし
宣言	unsigned char block_erase_ew0(unsigned short far* ers_addr);
説明	CPU 書き換えモード（EW0 モード）にて消去、ブランクチェック、フルステータスチェックを実行します。
引数	unsigned short far* ers_addr : 消去するブロックの最終アドレス（偶数番地）
リターン値	COMPLETE : 正常終了 CMD_SEQ_ERR : コマンドシーケンスエラー ERASE_ERR : 消去エラー

---

**program\_write\_ew0**

---

概要	CPU 書き換えモード（EW0 モード）での書き込み（プログラム）
ヘッダ	なし
宣言	unsigned char bk_program_write_ew0(unsigned short far* w_addr, unsigned short wd_1st, unsigned short wd_2nd)
説明	PC(ターミナルソフト)から送信された最大 64KB のデータを EW0 モードで書き込み、フルステータスチェックを実行します。
引数	unsigned short far* w_addr : ブロックの書き込み番地（末尾は 0h,4h,8h,Ch） unsigned short far* wd_1st : 書き込みデータ下位ワード（16 ビット） unsigned short far* wd_2nd : 書き込みデータ上位ワード（16 ビット）
リターン値	COMPLETE : 正常書き込み終了 CMD_SEQ_ERR : コマンドシーケンスエラー PROGRAM_ERR : 書き込みエラー

---

**full\_status\_check**

---

概要	フルステータスチェック
ヘッダ	なし
宣言	unsigned char full_status_check(void)
説明	フルステータス（コマンドエラー、書き込みエラー、消去エラー）チェックを行い結果を返します。
引数	なし
リターン値	COMPLETE : 正常終了 CMD_SEQ_ERR : コマンドシーケンスエラー PROGRAM_ERR : 書き込みエラー ERASE_ERR : 消去エラー

---

**send\_to\_ram**

---

概要	制御プログラム転送
ヘッダ	なし
宣言	void send_to_ram(void)
説明	イレーズ関数、プログラム関数、フルステータスチェック関数を RAM に転送します。
引数	なし
リターン値	なし

---

**asm\_smovf**

---

概要	RAM 転送処理
ヘッダ	なし
宣言	void asm_smovf(void _far *_source, void _near *_dest, unsigned int _size)
説明	指定された領域を RAM に転送します。
引数	void_far*_source            転送元アドレス（プログラム） void_near*_dest            転送先アドレス（RAM 領域） unsigned int_size           転送サイズ
リターン値	なし

---

**send\_message**

---

概要	メッセージ送信
ヘッダ	なし
宣言	void send_message (const unsigned char *mess)
説明	PC(ターミナルソフト)へメッセージを送信します。
引数	const unsigned char mess*    送信メッセージ配列の先頭アドレス
リターン値	なし

---

**dummy**

---

概 要	ダミー関数
ヘッダ	なし
宣 言	void dummy(void)
説 明	ダミー関数を使用して RAM に転送するプログラム容量を算出します。
引 数	なし
リターン値	なし

## 5.7 フローチャート

## 5.7.1 メイン処理

図 5.1 にメイン処理のフローチャートを示します。

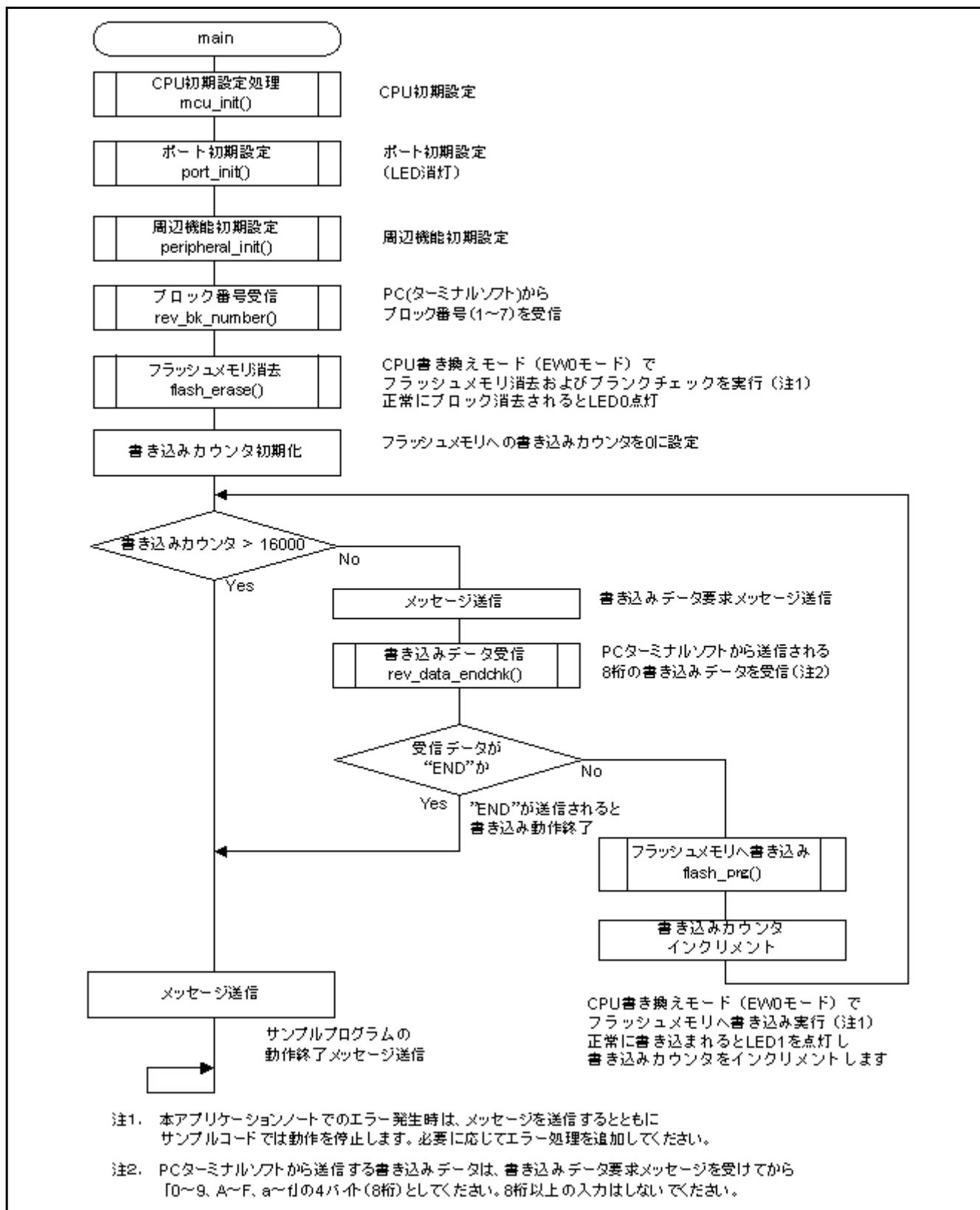


図 5.1 メイン処理

## 5.7.2 CPU 初期設定

図 5.2 に CPU 初期設定のフローチャートを示します。

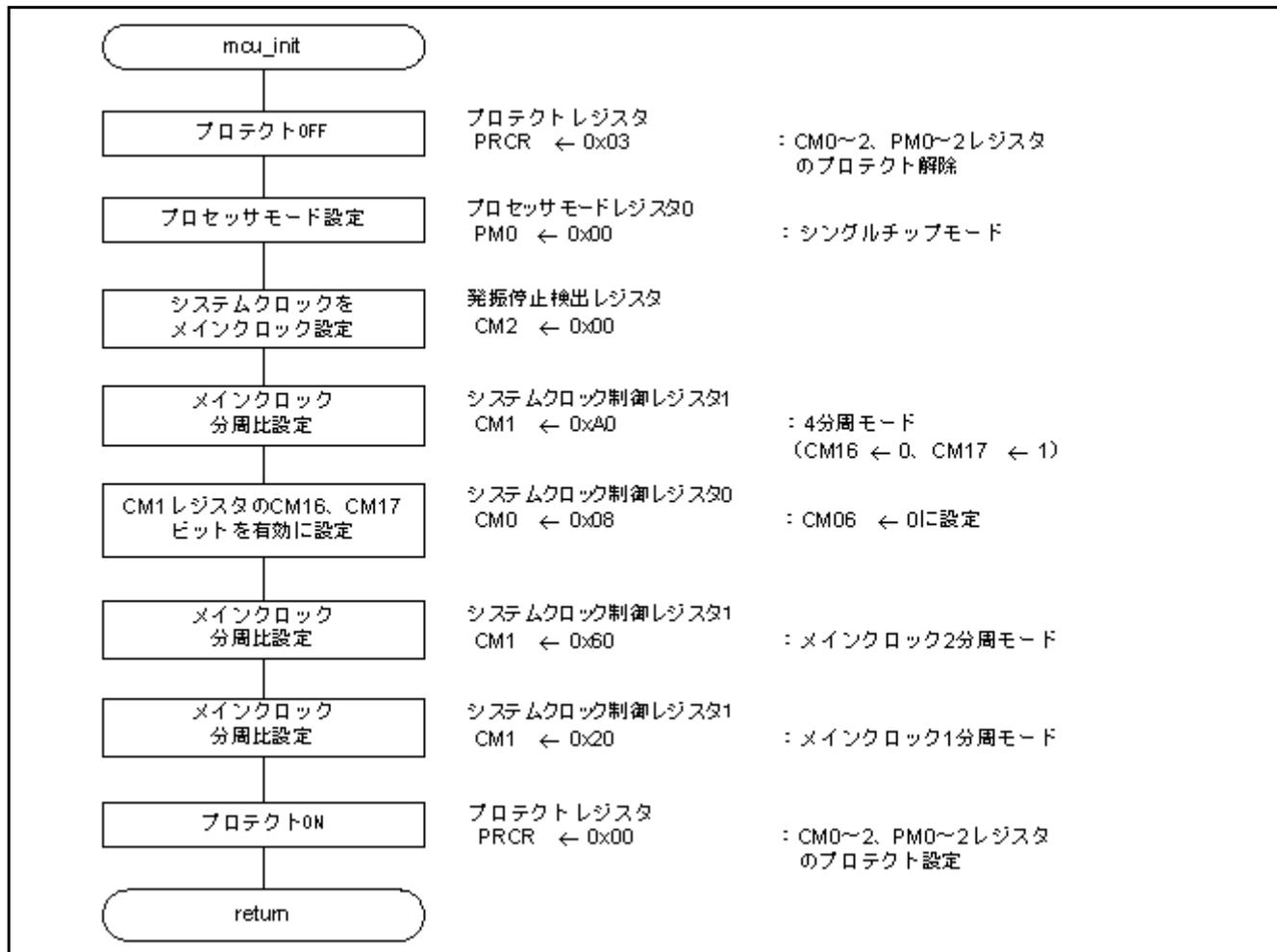


図 5.2 CPU 初期設定

## 5.7.3 プログラマブル入出力ポート初期設定

図 5.3 にプログラマブル入出力ポート初期設定のフローチャートを示します。

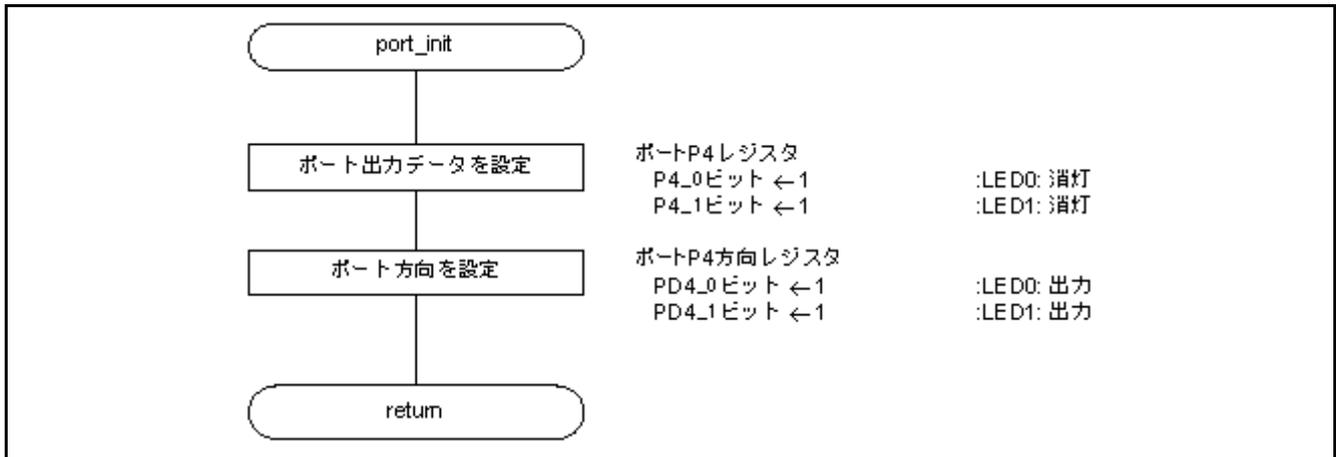


図 5.3 プログラマブル入出力ポート初期設定

## 5.7.4 周辺機能初期設定

図 5.4 に周辺機能初期設定のフローチャートを示します。

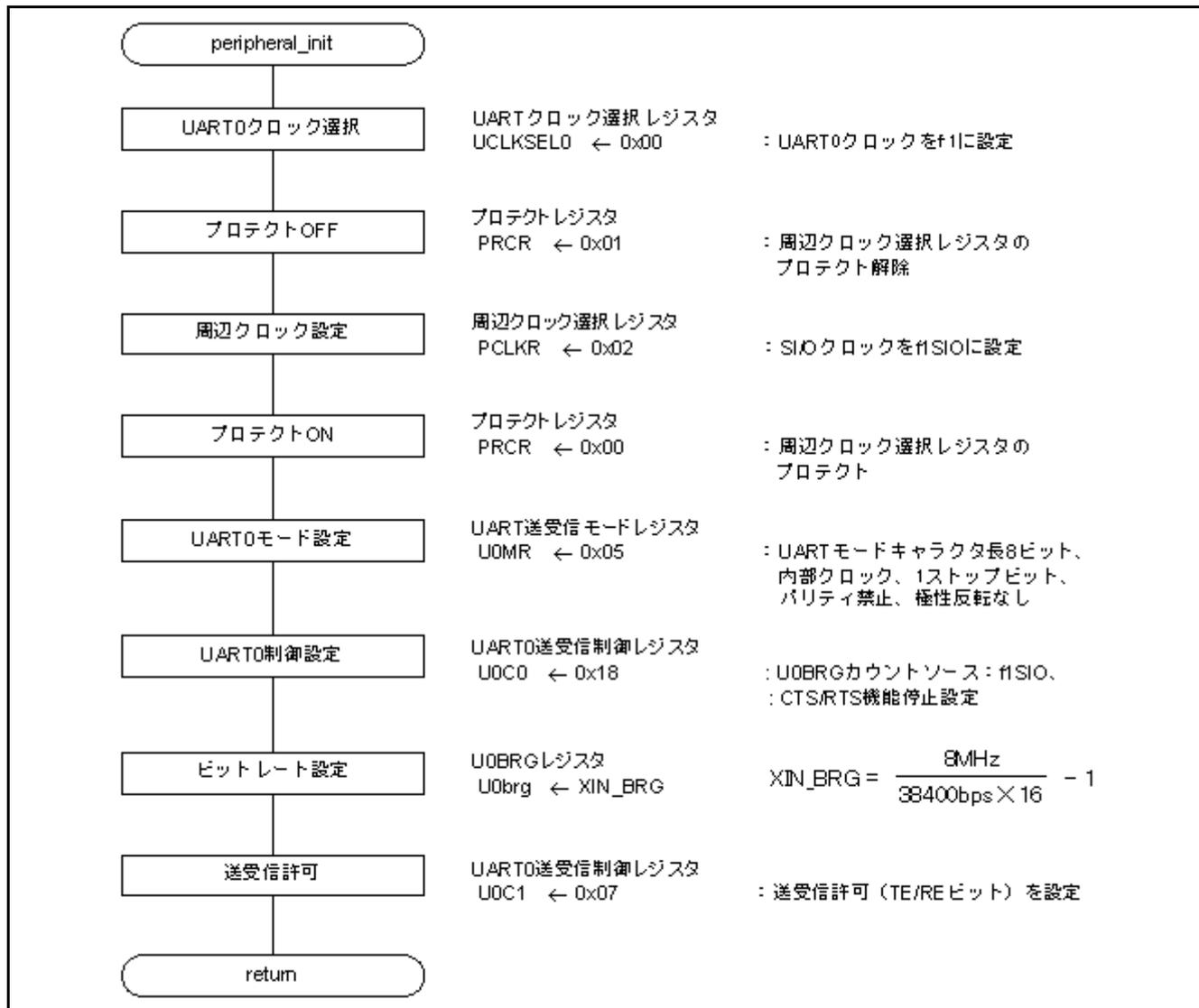


図 5.4 周辺機能初期設定

## 5.7.5 CPU クロック低速化

図 5.5 に CPU クロック低速化のフローチャートを示します。

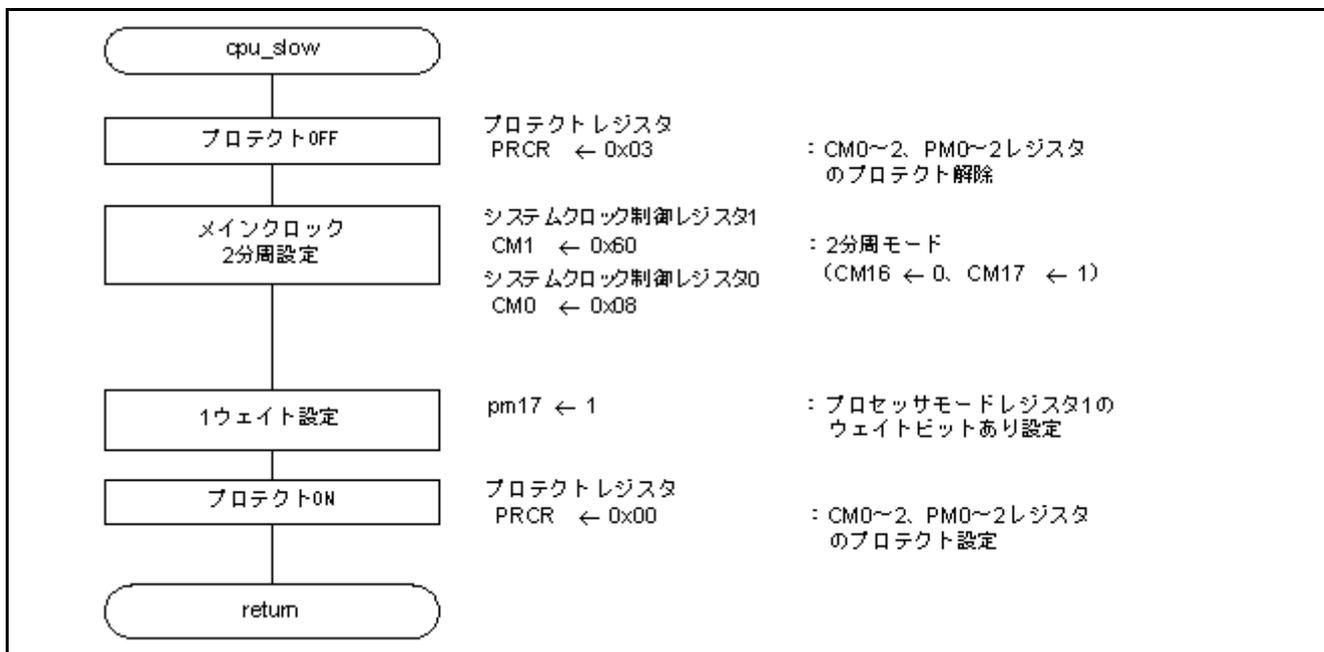


図 5.5 CPU クロック低速化

## 5.7.6 CPU クロック高速化

図 5.6 に CPU クロック高速化のフローチャートを示します。

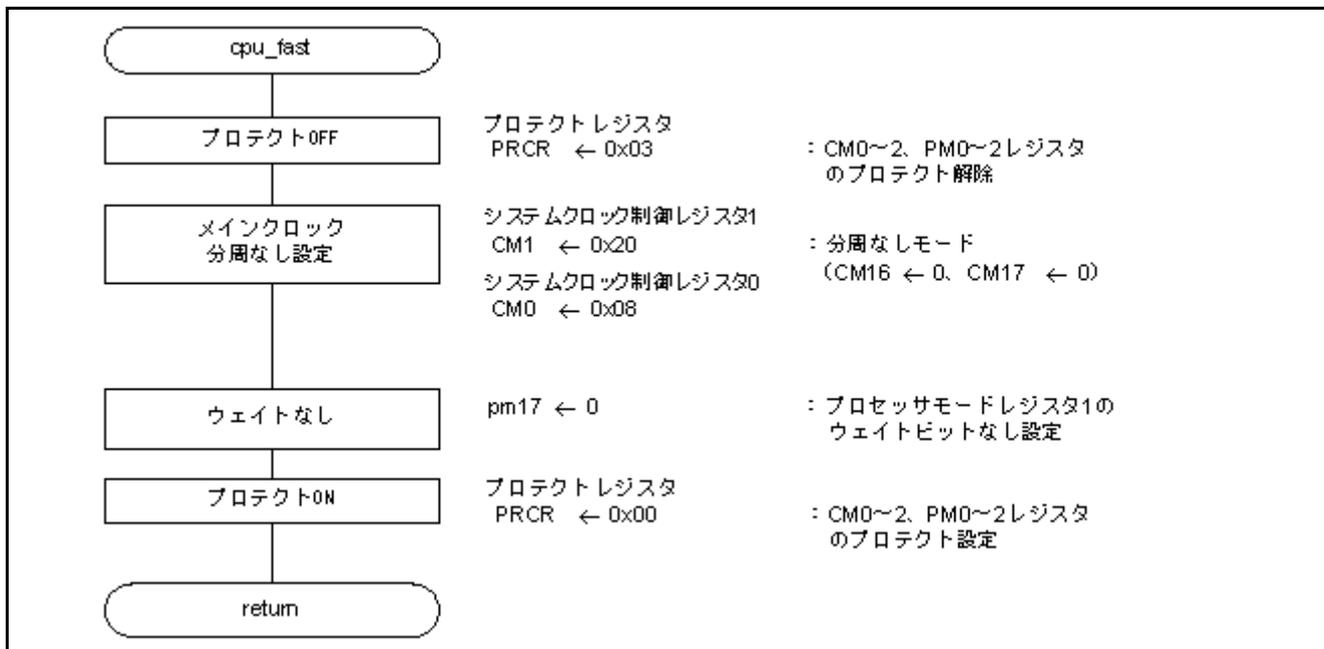


図 5.6 CPU クロック高速化

## 5.7.7 1 バイト受信

図 5.7 に 1 バイト受信のフローチャートを示します。

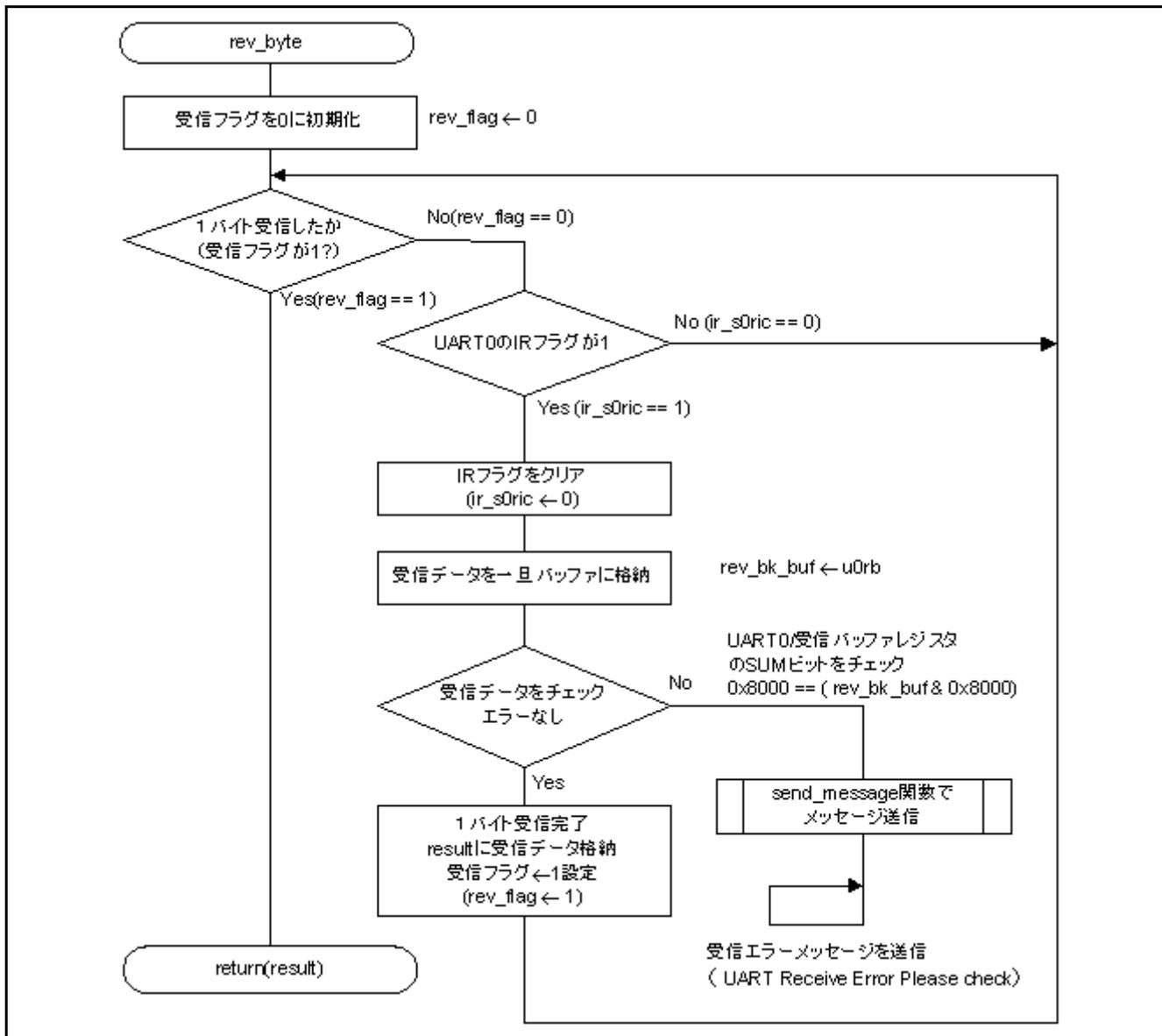


図 5.7 1 バイト受信

## 5.7.8 ブロック番号受信

図 5.8 にブロック番号受信のフローチャートを示します。

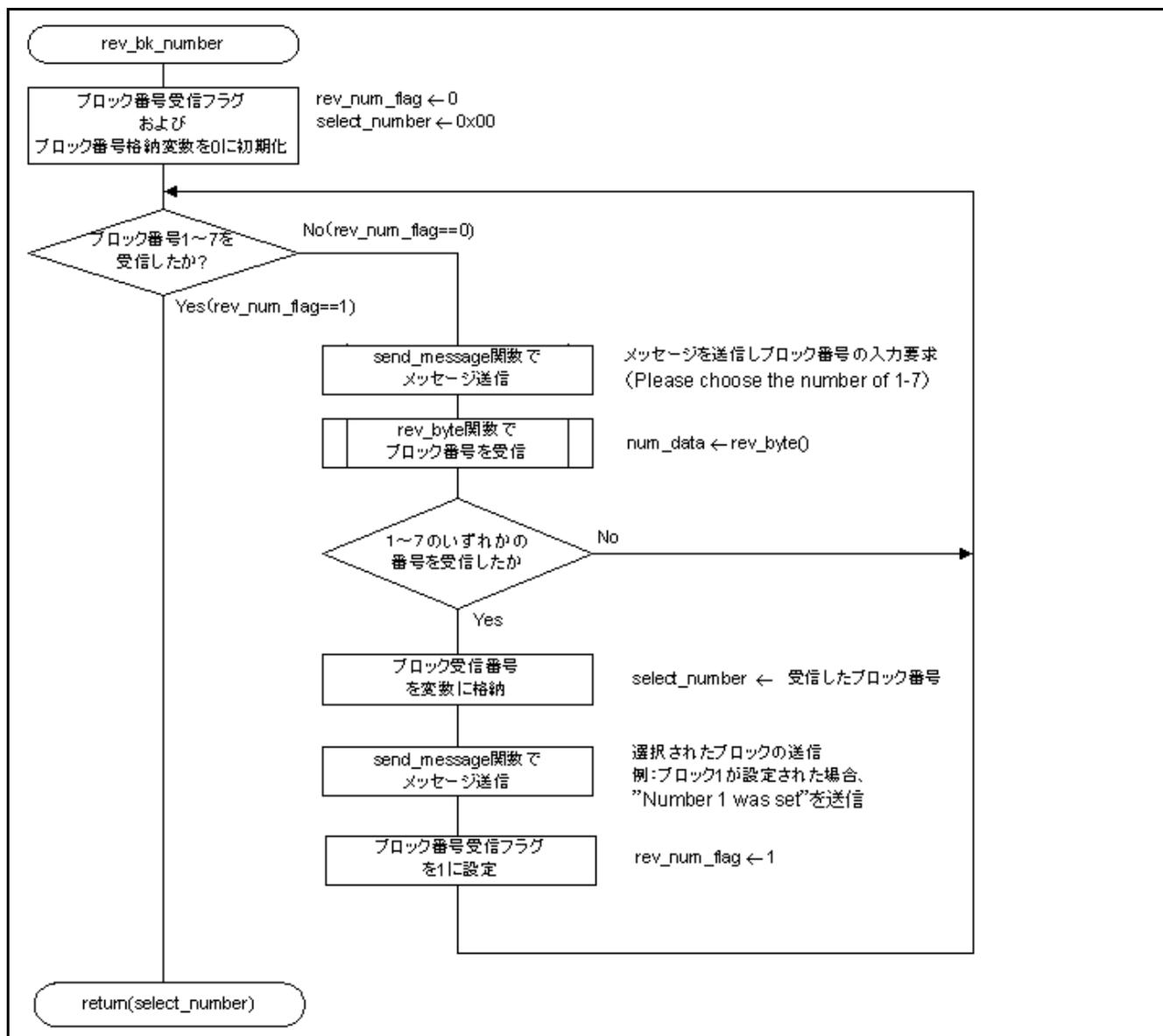


図 5.8 ブロック番号受信

## 5.7.9 フラッシュメモリ消去

図 5.9 にフラッシュメモリ消去のフローチャートを示します。

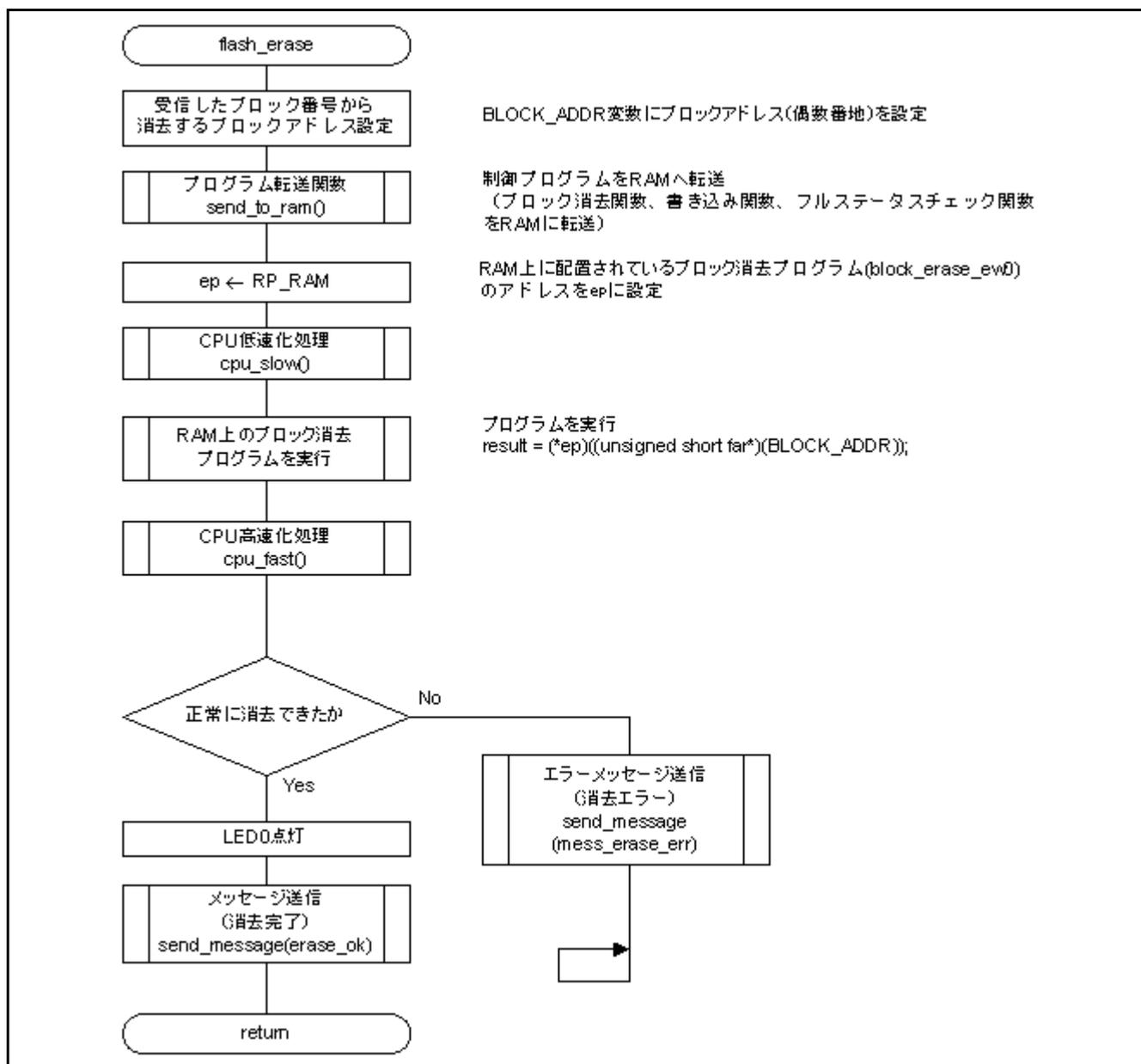


図 5.9 フラッシュメモリ消去

## 5.7.10 フラッシュメモリ書き込みデータ受信

図 5.10 にフラッシュメモリ書き込みデータ受信のフローチャートを示します。

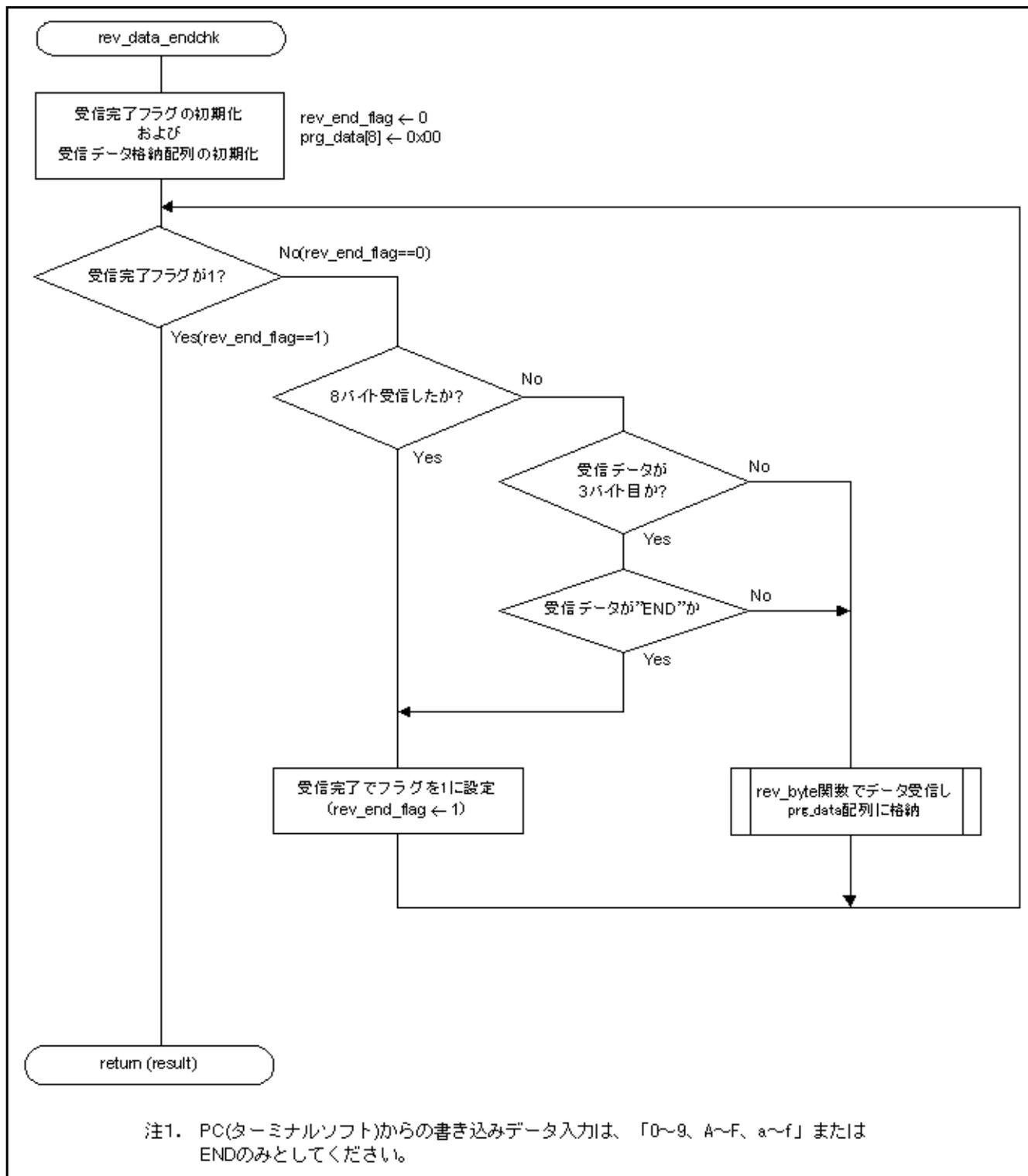


図 5.10 フラッシュメモリ書き込みデータ受信

## 5.7.11 フラッシュメモリ書き込みデータ生成

図 5.11 にフラッシュメモリ書き込みデータ生成のフローチャートを示します。

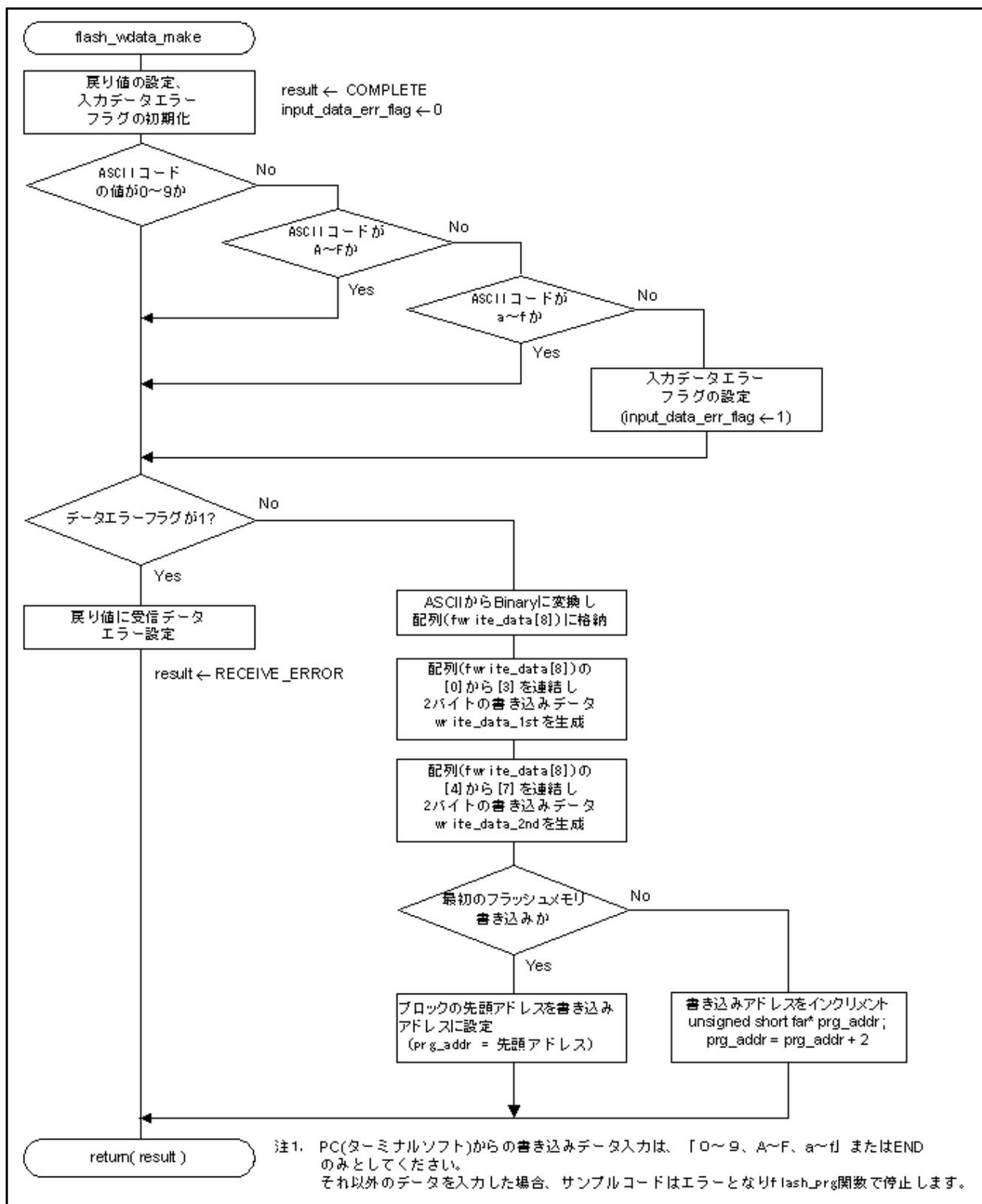


図 5.11 フラッシュメモリ書き込みデータ生成

## 5.7.12 フラッシュメモリ書き込み

図 5.12 にフラッシュメモリ書き込みのフローチャートを示します。

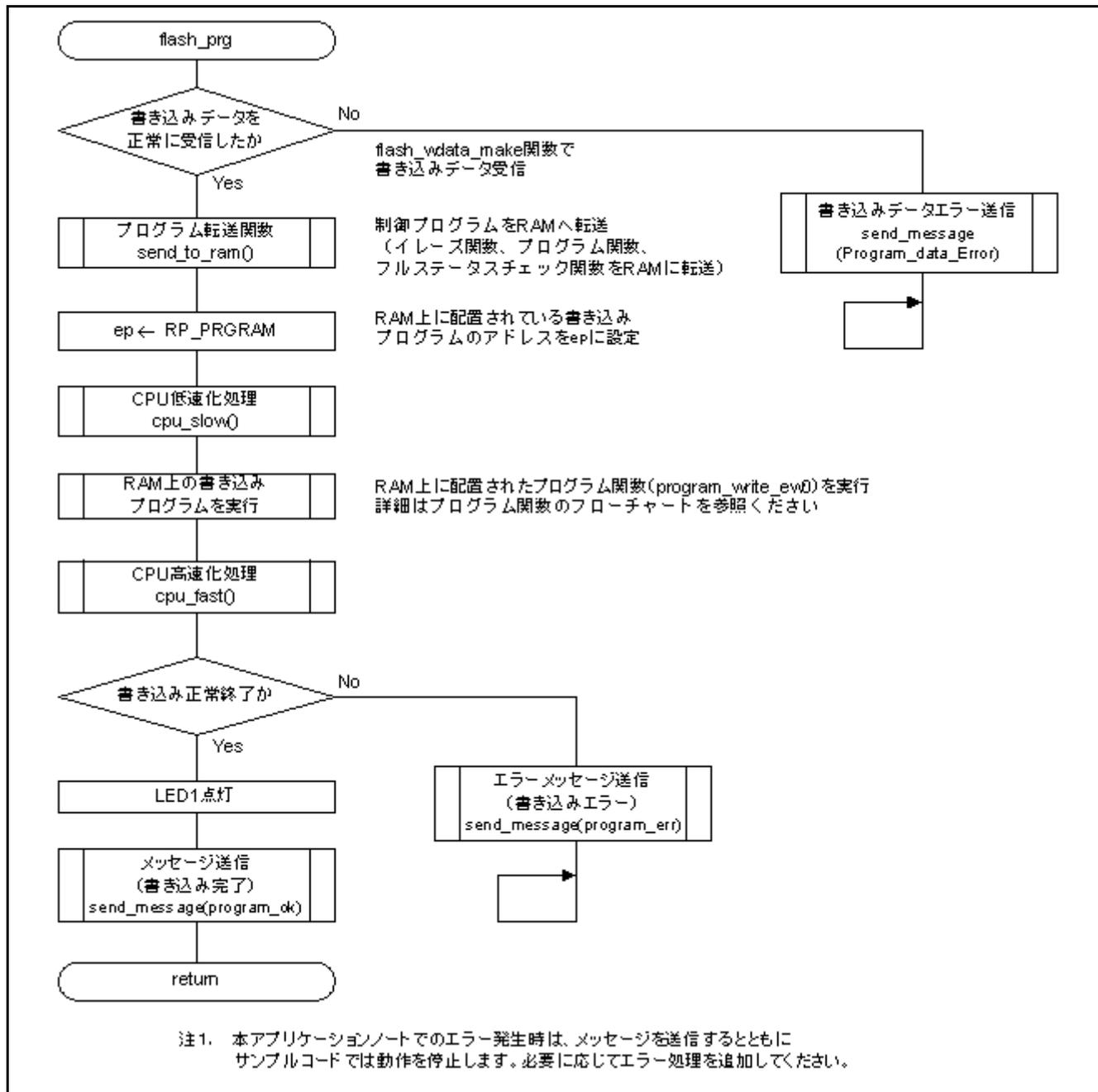
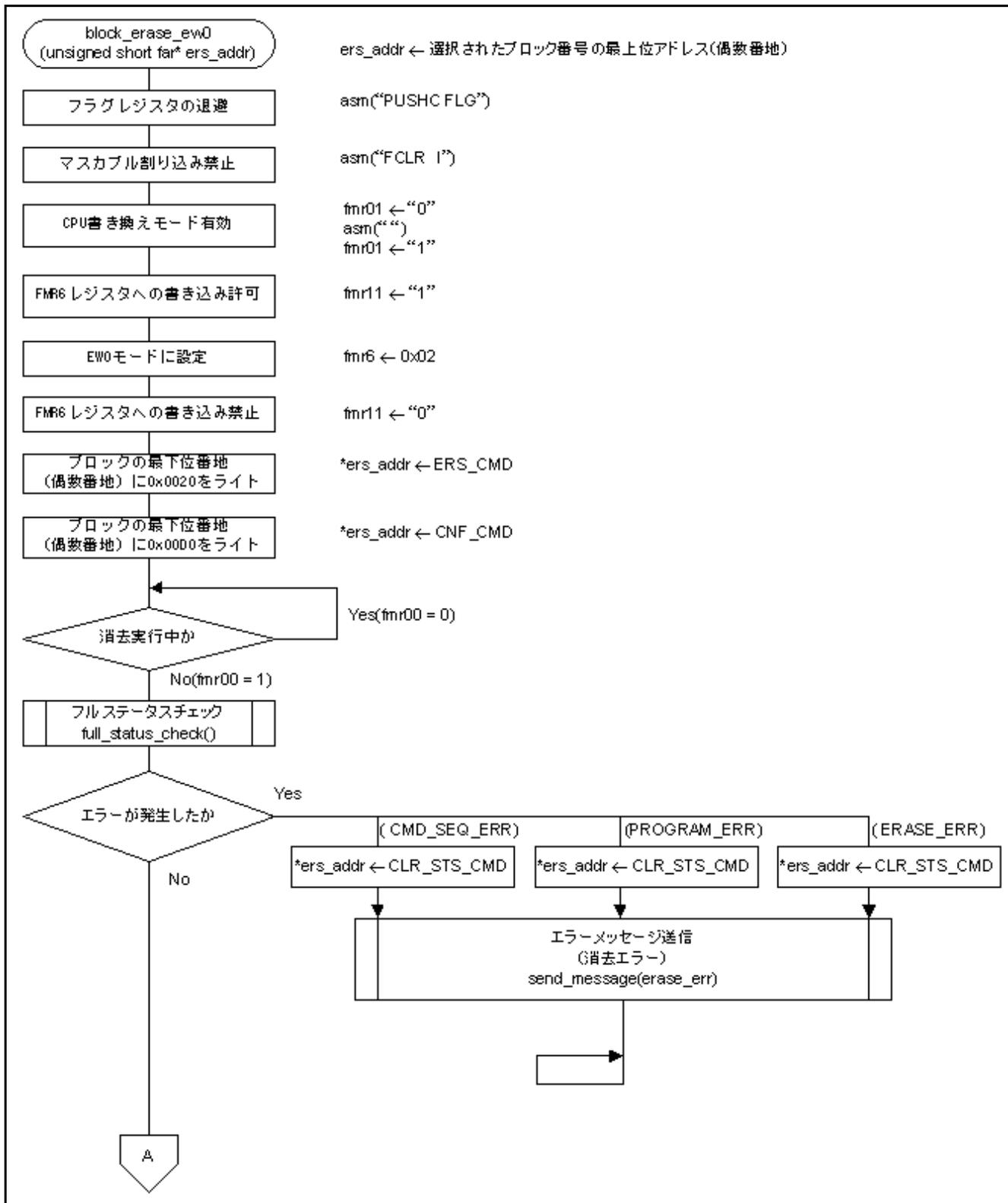


図 5.12 フラッシュメモリ書き込み

5.7.13 ブロック消去関数

図 5.13 にブロック消去のフローチャートを示します。



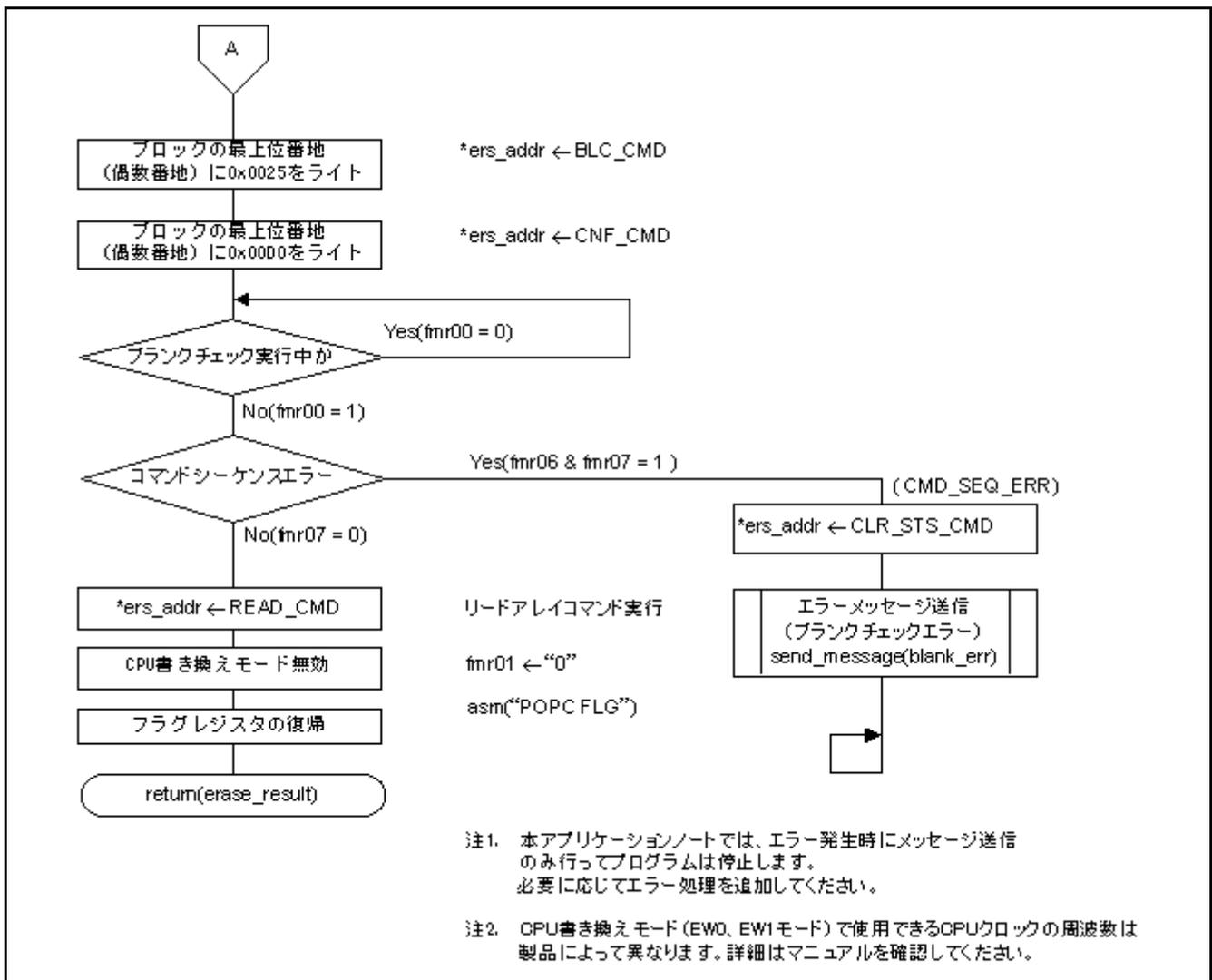


図 5.13 ブロック消去関数

## 5.7.14 ブロック書き込み関数

図 5.14 にブロック書き込み関数のフローチャートを示します。

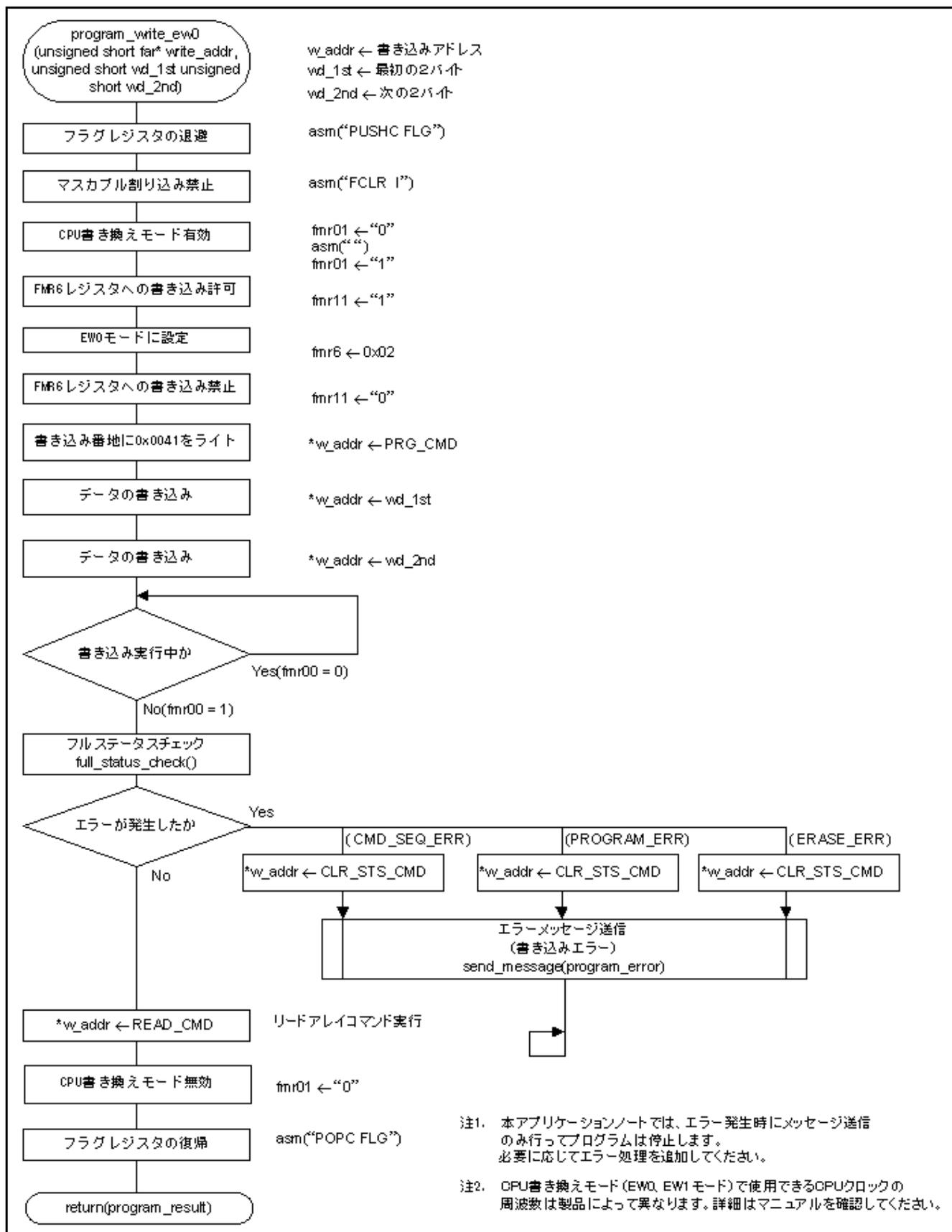


図 5.14 書き込み関数

## 5.7.15 フルステータスチェック

図 5.15 にフルステータスチェックのフローチャートを示します。

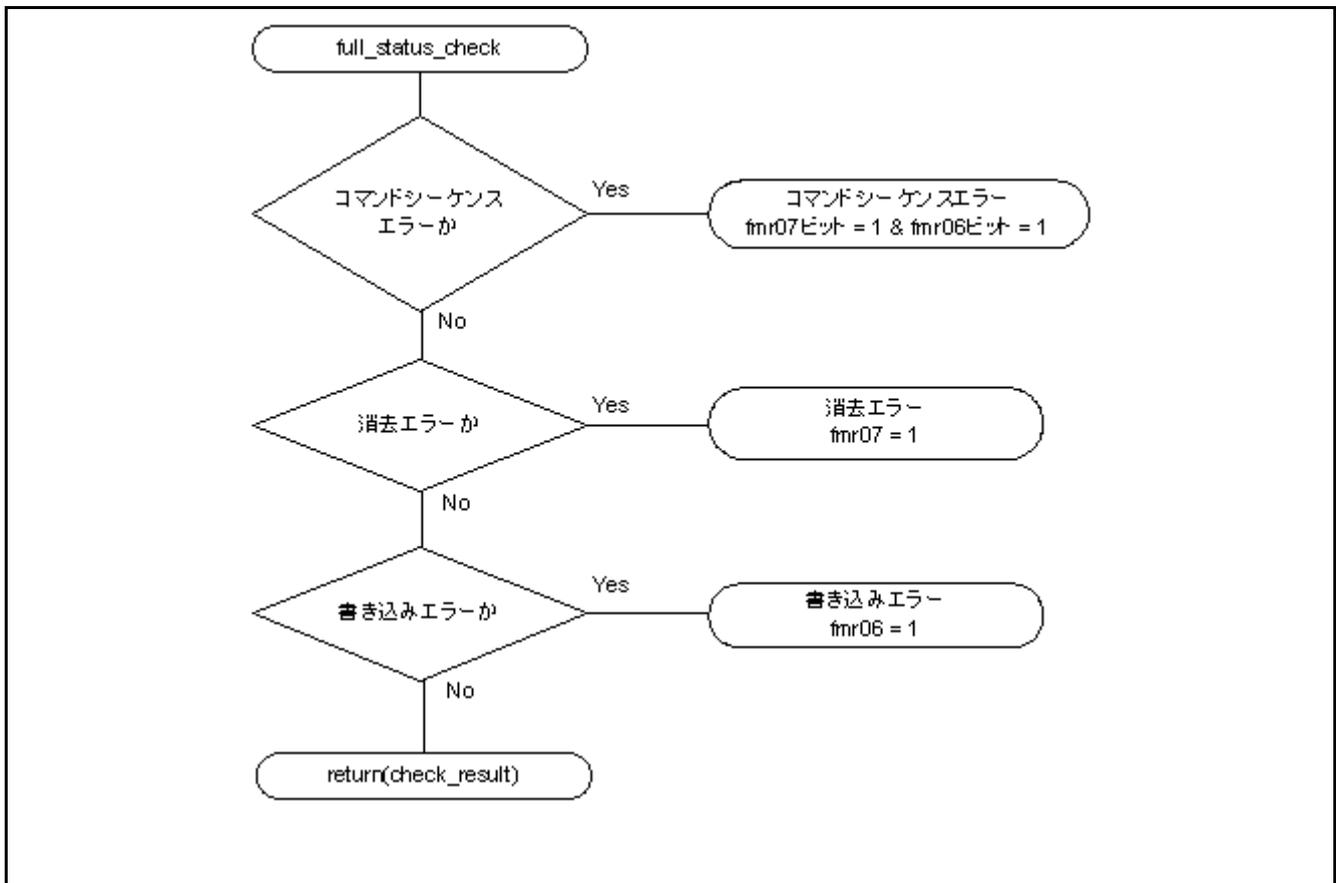


図 5.15 フルステータスチェック

## 5.7.16 制御プログラム転送

図 5.16 に制御プログラム転送のフローチャートを示します。

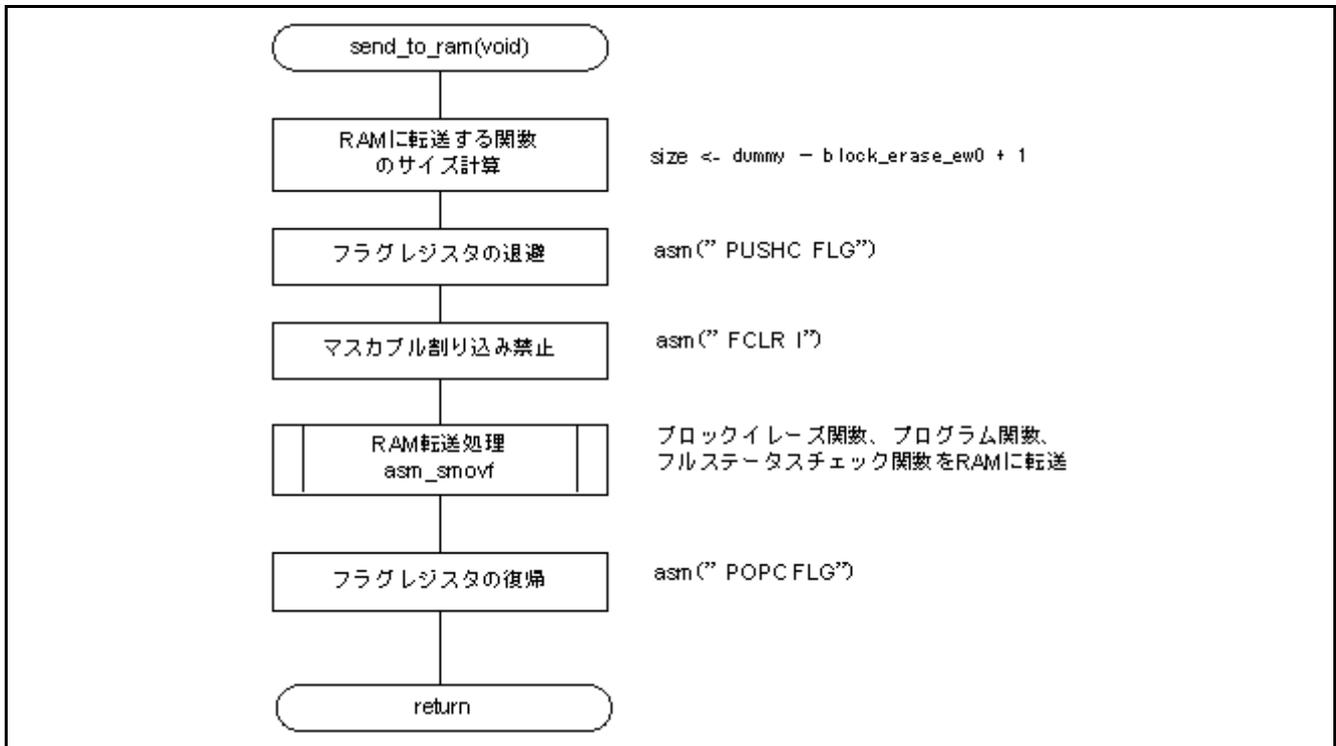


図 5.16 制御プログラム転送

## 5.7.17 RAM 転送処理

図 5.19 に RAM 転送処理のフローチャートを示します。

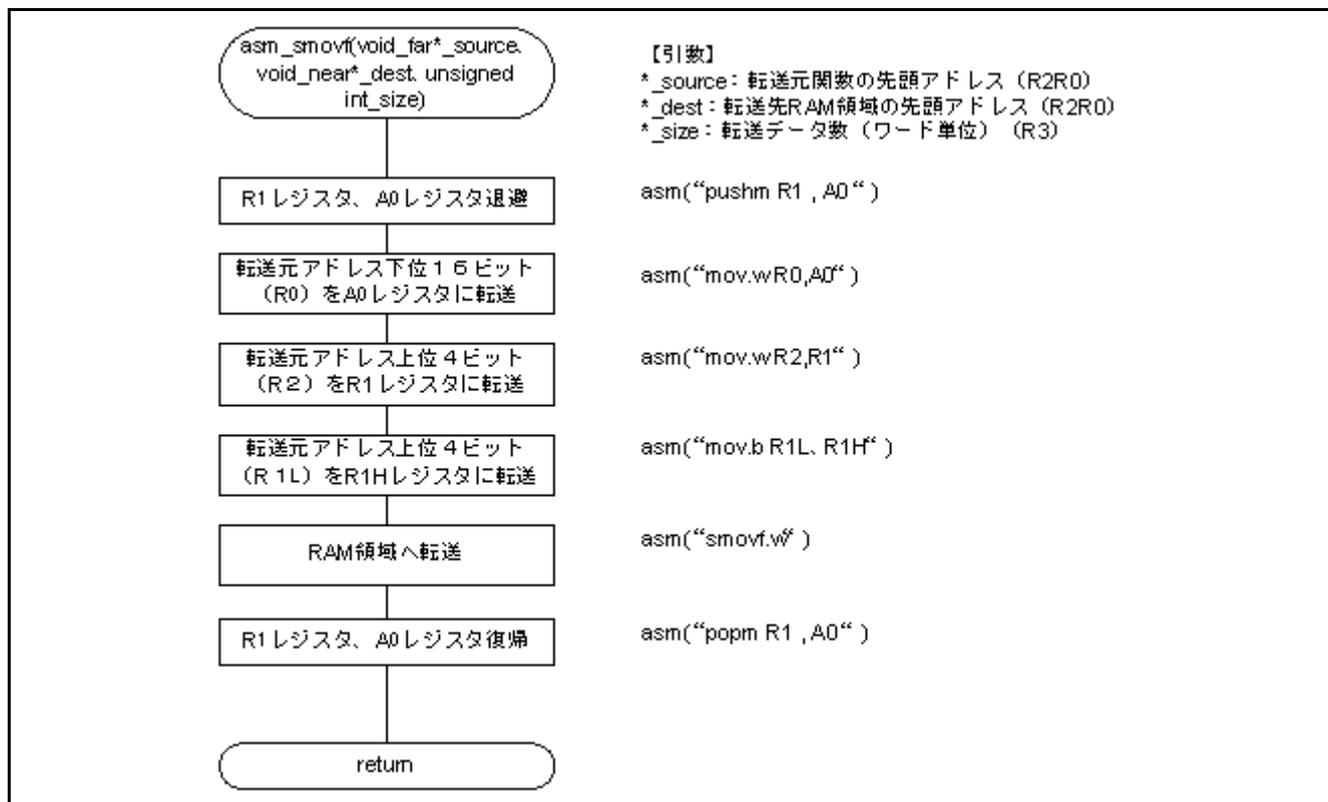


図 5.19 RAM 転送処理

## 5.7.18 メッセージ送信

図 5.20 にメッセージ送信のフローチャートを示します。

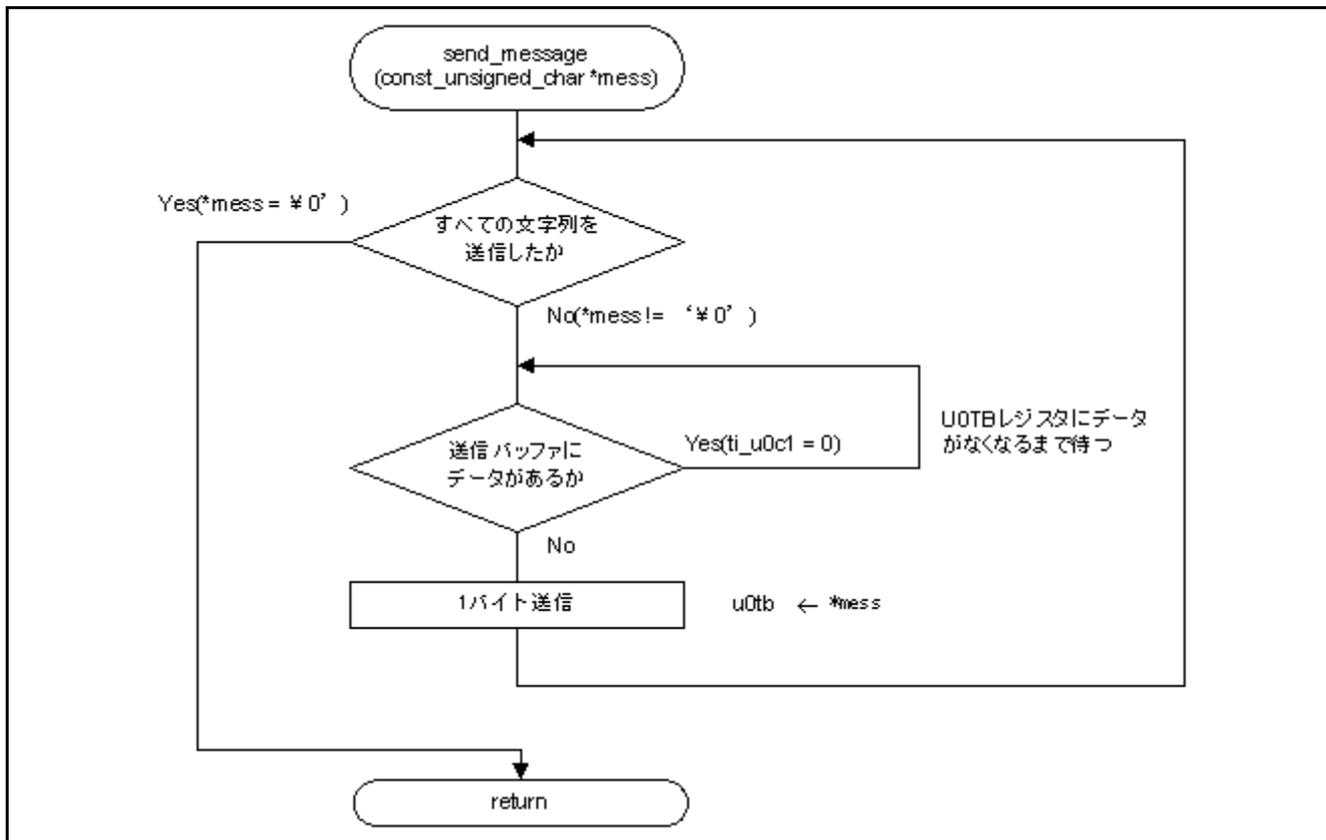


図 5.20 メッセージ送信

## 6. 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。

## 7. 参考ドキュメント

### ユーザーズマニュアル：ハードウェア

- M16C/63 グループ ユーザーズマニュアルハードウェア編 Rev.2.20
  - M16C/64 グループ ユーザーズマニュアルハードウェア編 Rev.1.05
  - M16C/64A グループ ユーザーズマニュアルハードウェア編 Rev.2.10
  - M16C/64C グループ ユーザーズマニュアルハードウェア編 Rev.1.10
  - M16C/65 グループ ユーザーズマニュアルハードウェア編 Rev.2.10
  - M16C/65C グループ ユーザーズマニュアルハードウェア編 Rev.1.10
  - M16C/6C グループ ユーザーズマニュアルハードウェア編 Rev.2.10
  - M16C/5LD グループ ユーザーズマニュアルハードウェア編 Rev.1.20
  - M16C/56D グループ ユーザーズマニュアルハードウェア編 Rev.1.20
  - M16C/5L グループ ユーザーズマニュアルハードウェア編 Rev.1.10
  - M16C/56 グループ ユーザーズマニュアルハードウェア編 Rev.1.10
  - M16C/5M グループ ユーザーズマニュアルハードウェア編 Rev.1.10
  - M16C/57 グループ ユーザーズマニュアルハードウェア編 Rev.1.10
- (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

- M16C シリーズ、R8C ファミリー用 C コンパイラパッケージ V5.45
- (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/>

改訂記録	M16C グループ PC ターミナルソフトウェアを用いた 内蔵フラッシュメモリ書き換え
------	--

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2015.07.23	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。

## 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（予約領域）のアクセス禁止

【注意】リザーブアドレス（予約領域）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（予約領域）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、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違えば、内部ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が異なる製品に変更する場合は、個々の製品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い合わせください。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 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

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/>