

---

## R8C/L3AM グループ

パワーオフ1モードを使用したパワーコントロール

R01AN0366JJ0101

Rev.1.01

2012.1.6

---

### 要旨

本アプリケーションノートでは、R8C/L3AM グループのパワーオフ1モードを使用して、パワーコントロールを行うための設定方法と、パワーオフモード時のデータ管理方法について説明します。

### 対象デバイス

R8C/L3AM 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. ハードウェア説明 .....                 | 4  |
| 3.1 使用端子一覧 .....                  | 4  |
| 4. ソフトウェア説明 .....                 | 5  |
| 4.1 動作概要 .....                    | 5  |
| 4.2 データフラッシュ領域の書き込みアドレス検索方法 ..... | 6  |
| 4.2.1 空レコード検索(データ“FFh”サーチ).....   | 6  |
| 4.3 必要メモリサイズ .....                | 10 |
| 4.4 定数一覧 .....                    | 10 |
| 4.5 構造体/共用体一覧 .....               | 11 |
| 4.6 変数一覧 .....                    | 11 |
| 4.7 関数一覧 .....                    | 11 |
| 4.8 関数仕様 .....                    | 12 |
| 4.9 フローチャート.....                  | 15 |
| 4.9.1 メイン処理.....                  | 15 |
| 4.9.2 リセット要因判別処理.....             | 16 |
| 4.9.3 システムクロック設定処理 .....          | 17 |
| 4.9.4 ユーザプログラム処理.....             | 18 |
| 4.9.5 タイマRE初期設定処理.....            | 19 |
| 4.9.6 パワーコントロール処理 .....           | 20 |
| 4.9.7 レコード書き込みアドレス初期設定処理 .....    | 21 |
| 4.9.8 データ書き込み制御処理 .....           | 24 |
| 4.9.9 書き込みデータ設定処理 .....           | 24 |
| 4.9.10 ブロックイレーズ処理.....            | 25 |
| 4.9.11 プログラムコマンド処理 .....          | 26 |
| 4.9.12 フルステータスチェック処理.....         | 28 |
| 5. サンプルコード.....                   | 29 |
| 6. 参考ドキュメント .....                 | 29 |

1. 仕様

標準動作モードからプログラムでパワーオフ1モードへ移行します。WKUP0端子に“L”パルスを入力するか、1分経過することにより、パワーオフ1モードを解除してリセット動作します。パワーオフ1モード解除によるリセット回数をカウントし、データフラッシュにカウントデータを書き込みます。

表 1.1に使用する周辺機能と用途を、図 1.1に動作例を示します。

表 1.1 使用する周辺機能と用途

| 周辺機能     | 用途               |
|----------|------------------|
| タイマRE    | パワーオフ1モードからの復帰要因 |
| フラッシュメモリ | リセット回数の書き込み      |

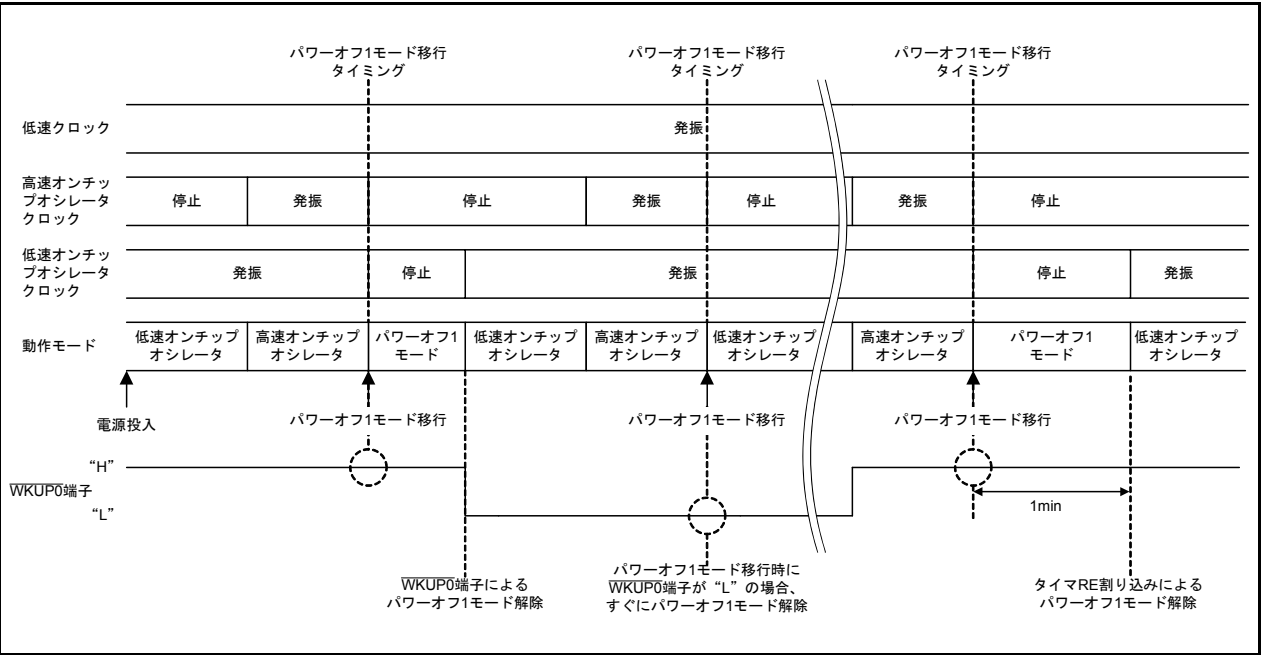


図 1.1 動作例

## 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 | R8C/L3AMグループ                                                                                                                                                        |
| 動作周波数  | ・ 高速オンチップオシレータクロック : 40MHz(標準)<br>・ XCINクロック : 32.768kHz<br>・ システムクロック : 20MHz<br>・ CPUクロック : 20MHz                                                                 |
| 動作電圧   | 5.0V(2.7V ~ 5.5Vで動作可能)                                                                                                                                              |
| 統合開発環境 | ルネサス エレクトロニクス製<br>High-performance Embedded Workshop Version 4.07                                                                                                   |
| Cコンパイラ | ルネサス エレクトロニクス製<br>M16C Series, R8C Family C Compiler V.5.45 Release 01<br>コンパイルオプション<br>-D__UART0__ -c -finfo -dir "\$(CONFIGDIR)" -R8C<br>(統合開発環境のデフォルト設定を使用しています) |

## 3. ハードウェア説明

### 3.1 使用端子一覧

表 3.1に使用端子と機能を示します。

表 3.1 使用端子と機能

| 端子名   | 入出力 | 内容           |
|-------|-----|--------------|
| WKUP0 | 入力  | パワーオフモード解除入力 |

## 4. ソフトウェア説明

### 4.1 動作概要

標準動作モードからプログラムでパワーオフ1モードへ移行します。WKUP0 端子に“L”パルスを入力するか、タイマRE割り込みによりパワーオフ1モードを解除します。タイマRE割り込み要求は、分周期割り込みタイミングで発生します。パワーオフ1モード解除によるリセット回数をカウントし、データフラッシュにカウントデータを書き込みます。

<設定>

- ・タイマREのカウントソースはfC4を使用します。
- ・24時間モードを選択します。
- ・タイマRE割り込み(分周期割り込み)を使用します。
- ・パワーオフモードは、パワーオフ1モードを使用します。

#### (1) リセット要因判別

判別するリセット要因は、以下の通りです。

<リセット要因>

- ・コールドスタートハードウェアリセット
- ・電圧監視0リセット
- ・ソフトウェアリセット
- ・ウォッチドッグタイマリセット
- ・ウォームスタートハードウェアリセット
- ・パワーオフ1モード解除

#### (2) 空レコード検索/書き込み済みカウントデータ取得

<リセット要因：コールドスタートハードウェアリセット、電圧監視0リセット、パワーオフ1モード解除>

データフラッシュ領域から空レコードを検索し、書き込みアドレスに空レコードのアドレス、使用ブロック選択に空レコードのブロック番号を設定します。また、検索結果より書き込み済みカウントデータの値を取得します。(全てのブロックがブランク(FFh)の場合は、カウントデータは“0xFFFFFFFF”となります。)

<リセット要因：ソフトウェアリセット、ウォッチドッグタイマリセット、ウォームスタートハードウェアリセット>

空レコード検索/書き込み済みカウントデータ取得は行いません。

#### (3) データフラッシュへの書き込み

<リセット要因：パワーオフ1モード解除>

カウントデータを更新し、書き込みアドレスにカウントデータを書き込みます。

<リセット要因：パワーオフ1モード解除以外>

データフラッシュへの書き込みは行いません。

#### (4) パワーオフ1モード移行

パワーオフ1モードに移行します。(パワーオフ1モード移行時にWKUP0端子が“L”の場合は、一旦、パワーオフ1モード移行後、すぐに解除します。)

#### (5) パワーオフ1モード解除

WKUP0端子への“L”パルス入力、またはタイマREの分周期割り込み発生によりパワーオフ1モードを解除します。

#### (6) 以降、(1)~(5)を繰り返します。

## 4.2 データフラッシュ領域の書き込みアドレス検索方法

本アプリケーションノートでは1レコードを4バイトとし、ブロックを256分割します。使用するブロック(ブロックA、ブロックB)では512レコードになります。図4.1にデータフラッシュとレコードの関係図を示します。

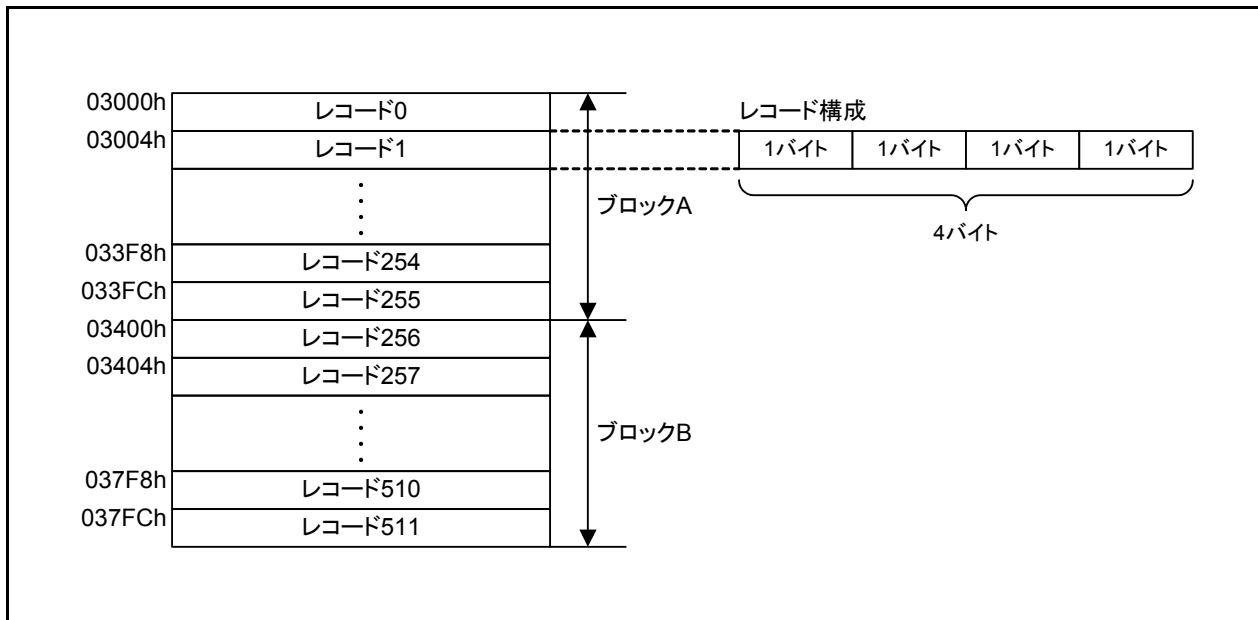


図 4.1 データフラッシュとレコードの関係図

### 4.2.1 空レコード検索(データ“FFh”サーチ)

データフラッシュに書き込んだデータは、電源を切っても保持されます。そのため、リセットスタート後に全てのデータが“FFh”であるレコード(空レコード)を検索します。

以下に空レコードの検索方法について説明します。

- (1) レコード0の先頭番地にサーチポインタを設定します。

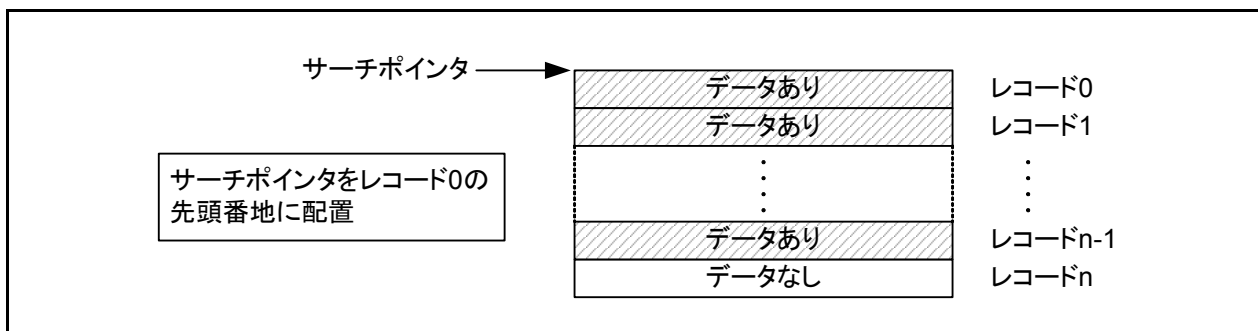


図 4.2 サーチポインタをレコード0の先頭番地に配置

- (2) サーチポインタが示すレコードが空レコード(ALL“FFh”)であるかをチェックします。

- (3) 空レコードでなかった場合は、サーチポインタを次のレコードの先頭番地に設定します。

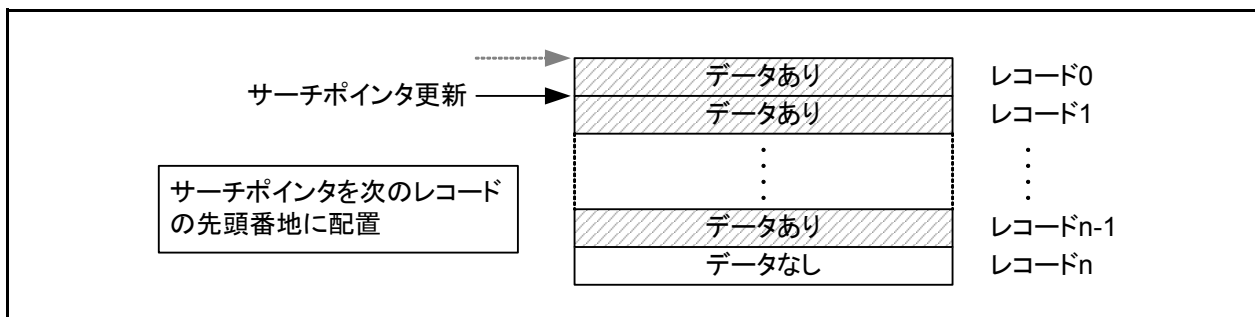


図 4.3 サーチポインタ更新

- (4) 空レコードが見つかるか、ブロック A の全てのレコードをチェックするまで(2)～(3)を繰り返して実行します。
- (5) 同様にブロック B(レコード256以降)に対して(1)～(4)を実行します。
- (6) ブロック A、ブロック B に対して検索した結果により、使用ブロックおよび書き込みアドレスを設定します。表 4.1 に検索結果と使用ブロック/書き込みアドレスを示します。

表 4.1 検索結果と使用ブロック/書き込みアドレス

| 検索結果                    |                         | 図参照   | 使用ブロック | 書き込みアドレス    | イレーズ実行ブロック       |
|-------------------------|-------------------------|-------|--------|-------------|------------------|
| ブロック A                  | ブロック B                  |       |        |             |                  |
| 全て空レコード                 | 全て空レコード                 | —     | ブロック A | ブロック A 先頭   | 実行しない            |
| 書き込みレコードあり              | 全て空レコード                 | 図 4.4 | ブロック A | ブロック A 検索結果 | 実行しない            |
| 全て書き込みレコード<br>(空レコードなし) | 全て空レコード                 | —     | ブロック B | ブロック B 先頭   | 実行しない            |
| 全て空レコード                 | 書き込みレコードあり              | 図 4.5 | ブロック B | ブロック B 検索結果 | 実行しない            |
| 全て空レコード                 | 全て書き込みレコード<br>(空レコードなし) | —     | ブロック A | ブロック A 先頭   | 実行しない            |
| 全て書き込みレコード<br>(空レコードなし) | 書き込みレコードあり              | 図 4.6 | ブロック B | ブロック B 検索結果 | ブロック A           |
| 書き込みレコードあり              | 全て書き込みレコード<br>(空レコードなし) | 図 4.7 | ブロック A | ブロック A 検索結果 | ブロック B           |
| 全て書き込みレコード<br>(空レコードなし) | 全て書き込みレコード<br>(空レコードなし) | 図 4.8 | ブロック A | ブロック A 先頭   | ブロック A<br>ブロック B |
| 書き込みレコードあり              | 書き込みレコードあり              | 図 4.9 | ブロック A | ブロック A 先頭   | ブロック A<br>ブロック B |

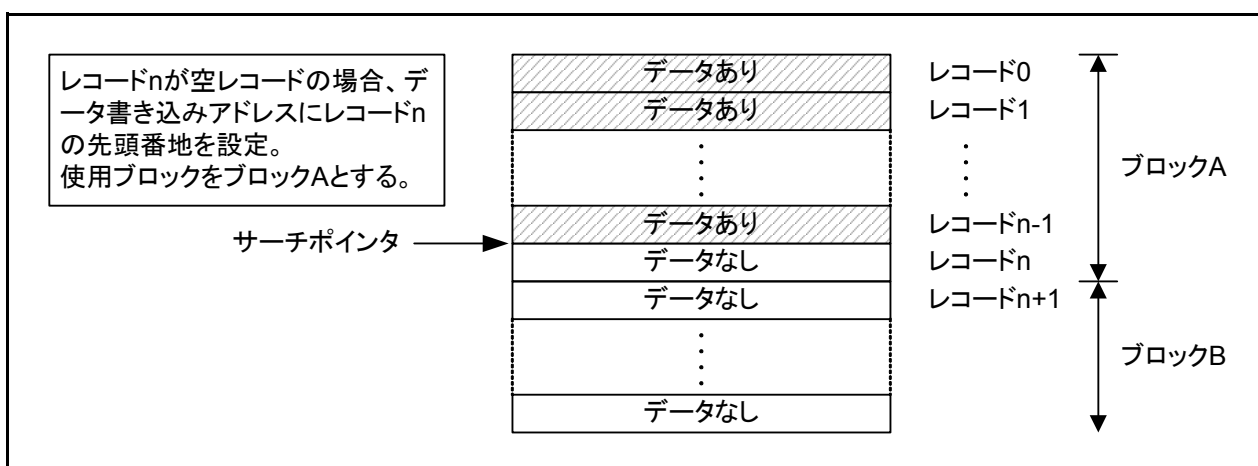


図 4.4 ブロック A：書き込みレコードあり／ブロック B：全て空レコード

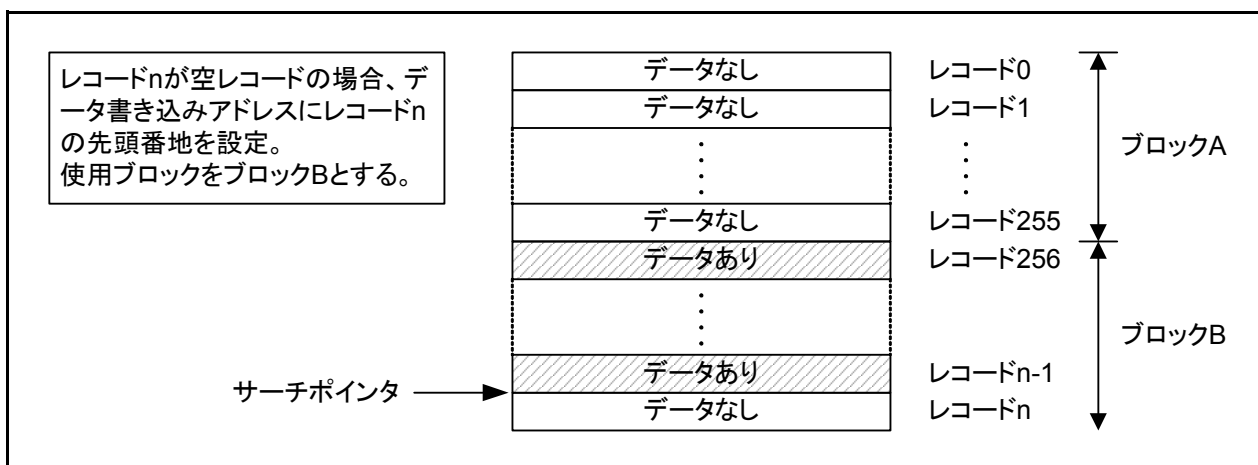


図 4.5 ブロック A：全て空レコード／ブロック B：書き込みレコードあり

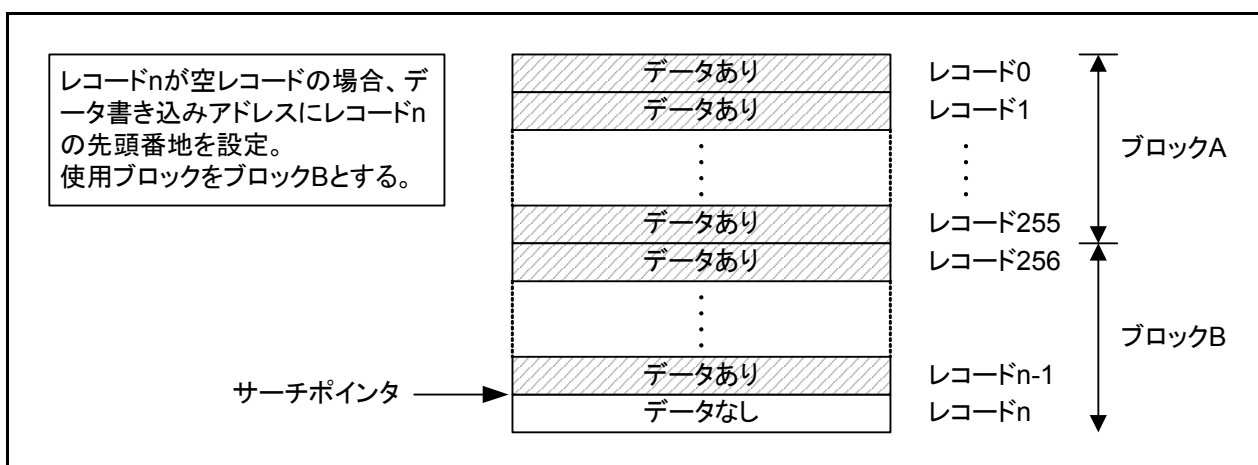


図 4.6 ブロック A：全て書き込みレコード／ブロック B：書き込みレコードあり

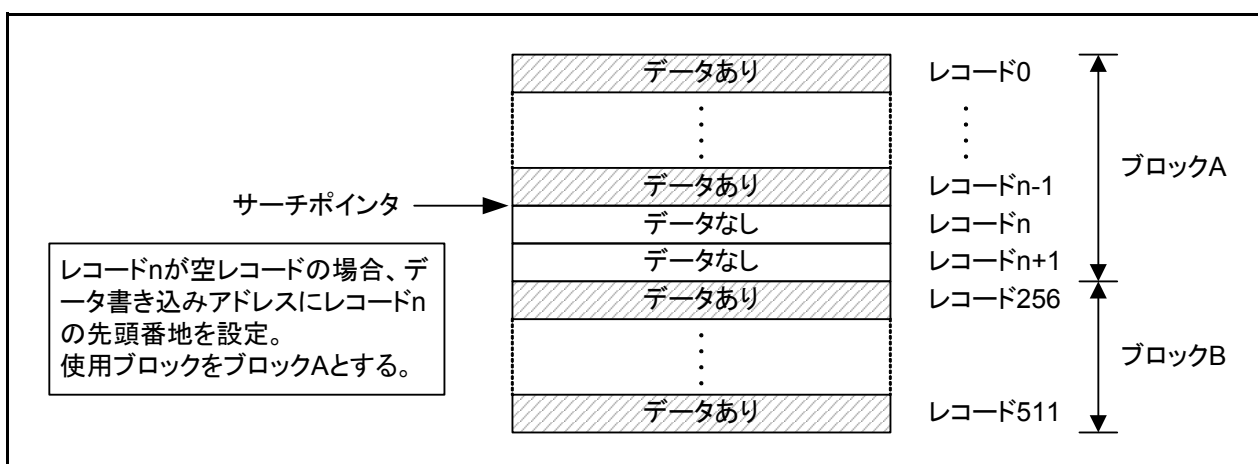


図 4.7 ブロックA：書き込みレコードあり／ブロックB：全て書き込みレコード

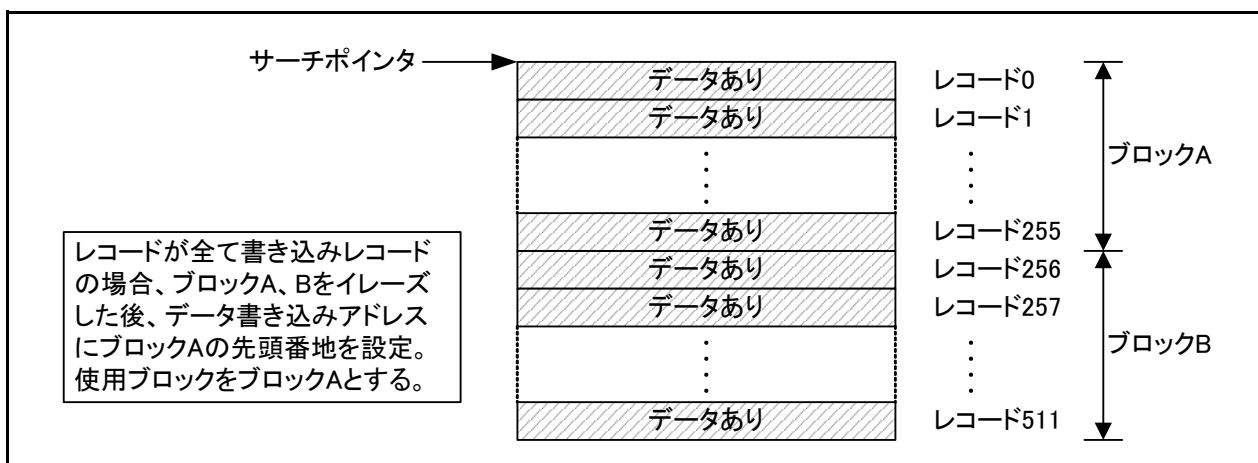


図 4.8 ブロックA：全て書き込みレコード／ブロックB：全て書き込みレコード

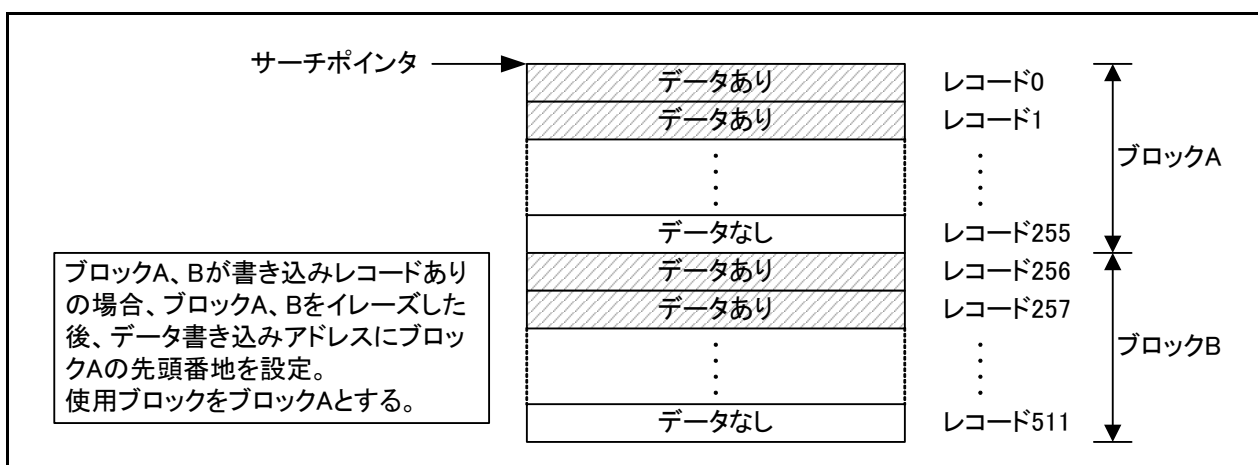


図 4.9 ブロックA：書き込みレコードあり／ブロックB：書き込みレコードあり

### 4.3 必要メモリサイズ

表 4.2 に必要メモリサイズを示します。

表 4.2 必要メモリサイズ

| 使用メモリ        | サイズ      | 備考                   |
|--------------|----------|----------------------|
| ROM          | 1179 バイト | r01an0366_src モジュール内 |
| RAM          | 14 バイト   | r01an0366_src モジュール内 |
| 最大使用ユーザスタック  | 33 バイト   |                      |
| 最大使用割り込みスタック | 18 バイト   |                      |

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。

### 4.4 定数一覧

表 4.3 にサンプルコードで使用する定数を示します。

表 4.3 サンプルコードで使用する定数

| 定数名               | 設定値                          | 内容                   |
|-------------------|------------------------------|----------------------|
| COLD_HARDWARE_RST | 0                            | ハードウェアリセット(コールドスタート) |
| LVD_RST           | 1                            | 電圧監視0リセット            |
| SOFTWARE_RST      | 2                            | ソフトウェアリセット           |
| WDT_RST           | 3                            | ウォッチドッグタイマリセット       |
| WARM_HARDWARE_RST | 4                            | ハードウェアリセット(ウォームスタート) |
| POWER_OFF_MODE    | 5                            | パワーオフ1モード解除          |
| BLOCK_A           | (unsigned char *)<br>0x03000 | ブロックAの先頭アドレス         |
| BLOCK_A_END       | (unsigned char *)<br>0x033FF | ブロックAの最終アドレス         |
| BLOCK_B           | (unsigned char *)<br>0x03400 | ブロックBの先頭アドレス         |
| BLOCK_B_END       | (unsigned char *)<br>0x037FF | ブロックBの最終アドレス         |
| RECORD_SIZE       | 4                            | 1レコードのサイズ            |
| BLOCK_A_SELECT    | 0                            | 使用ブロックがブロックA         |
| BLOCK_B_SELECT    | 1                            | 使用ブロックがブロックB         |
| NORMAL            | 0x00                         | 正常終了                 |
| CMD_SEQ_ERROR     | 0x01                         | コマンドシーケンスエラー         |
| ERS_BLK_CHK_ERROR | 0x02                         | イレーズ/ブランクチェックエラー     |
| PROGRAM_ERROR     | 0x03                         | プログラムエラー             |
| ALL_FF_BLOCK_A    | 0x00                         | ブロックA全て空レコード         |
| WRITE_BLOCK_A     | 0x01                         | ブロックA書き込みレコードあり      |
| ALL_XX_BLOCK_A    | 0x02                         | ブロックA全て書き込みレコード      |
| ALL_FF_BLOCK_B    | 0x00                         | ブロックB全て空レコード         |
| WRITE_BLOCK_B     | 0x10                         | ブロックB書き込みレコードあり      |
| ALL_XX_BLOCK_B    | 0x20                         | ブロックB全て書き込みレコード      |

## 4.5 構造体/共用体一覧

図 4.10 にサンプルコードで使用する構造体/共用体を示します。

```
typedef union{
    unsigned long all;
    struct{
        unsigned char lower;           /* 下位データ */
        unsigned char middle_lower;    /* 中下位データ */
        unsigned char middle_upper;    /* 中上位データ */
        unsigned char upper;           /* 上位データ */
    }byte;
}byte_dt;
```

図 4.10 サンプルコードで使用する構造体/共用体

## 4.6 変数一覧

表 4.4 にグローバル変数を示します。

表 4.4 グローバル変数

| 型             | 変数名                          | 内容                                  | 使用関数                                             |
|---------------|------------------------------|-------------------------------------|--------------------------------------------------|
| byte_dt       | DataStr                      | カウントデータ                             | user_program、<br>write_address_init、<br>set_data |
| unsigned char | block_select                 | 使用ブロック選択                            | write_address_init、<br>data_write                |
| unsigned char | *read_addr                   | 読み出しアドレス                            | write_address_init                               |
| unsigned char | *write_addr                  | 書き込みアドレス                            | write_address_init、<br>data_write                |
| unsigned char | record_data<br>[RECORD_SIZE] | 書き込み用レコード                           | write_control                                    |
| unsigned char | write_req                    | 書き込み要求 1 : 書き込み要求あり<br>0 : 書き込み要求なし | main、user_program                                |

## 4.7 関数一覧

表 4.5 に関数を示します。

表 4.5 関数

| 関数名                | 概要                 |
|--------------------|--------------------|
| reset_check        | リセット要因判別処理         |
| mcu_init           | システムクロック設定処理       |
| user_program       | ユーザプログラム処理         |
| timer_re_init      | タイマRE初期設定処理        |
| power_control      | パワーコントロール処理        |
| write_address_init | レコード書き込みアドレス初期設定処理 |
| write_control      | データ書き込み制御処理        |
| set_data           | 書き込みデータ設定処理        |
| block_erase        | ブロックイレーズ処理         |
| data_write         | プログラムコマンド処理        |
| full_sts_chk       | フルステータスチェック処理      |

## 4.8 関数仕様

サンプルコードの関数仕様を示します。

|             |                                                                                                                                                                                                              |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reset_check |                                                                                                                                                                                                              |
| 概 要         | リセット要因判別処理                                                                                                                                                                                                   |
| ヘッダ         | なし                                                                                                                                                                                                           |
| 宣 言         | unsigned char reset_check(void)                                                                                                                                                                              |
| 説 明         | リセット要因を判別し結果を返します。                                                                                                                                                                                           |
| 引 数         | なし                                                                                                                                                                                                           |
| リターン値       | ●コールドスタートハードウェアリセット：COLD_HARDWARE_RST(0)<br>●電圧監視0リセット：LVD_RST(1)<br>●ソフトウェアリセット：SOFTWARE_RST(2)<br>●ウォッチドッグタイマリセット：WDT_RST(3)<br>●ウォームスタートハードウェアリセット：WARM_HARDWARE_RST(4)<br>●パワーオフ1モード解除：POWER_OFF_MODE(5) |
| 備 考         | —                                                                                                                                                                                                            |

| mcu_init |                     |
|----------|---------------------|
| 概 要      | システムクロック設定処理        |
| ヘッダ      | なし                  |
| 宣 言      | void mcu_init(void) |
| 説 明      | クロックの設定を行います。       |
| 引 数      | なし                  |
| リターン値    | なし                  |
| 備 考      | —                   |

|       |                                                                                                                        |
|-------|------------------------------------------------------------------------------------------------------------------------|
| <hr/> |                                                                                                                        |
| 概要    | ユーザプログラム処理                                                                                                             |
| ヘッダ   | なし                                                                                                                     |
| 宣言    | void user_program(unsigned char reset_result)                                                                          |
| 説明    | ユーザプログラムの処理を行います。必要に応じて処理を追加してください。<br>本アプリケーションノートでは、パワーオフ1モード解除によるリセット回数をカウントして、データフラッシュ書き込み要求(write_req)に“1”を設定します。 |
| 引数    | 第一引数：reset_result                                 ： リセット要因判別結果                                                         |
| リターン値 | なし                                                                                                                     |
| 備考    | —                                                                                                                      |

| timer_re_init |                                       |
|---------------|---------------------------------------|
| 概 要           | タイマRE初期設定処理                           |
| ヘッダ           | なし                                    |
| 宣 言           | void timer_re_init(void)              |
| 説 明           | タイマREをリアルタイムクロックモードで使用するための初期設定を行います。 |
| 引 数           | なし                                    |
| リターン値         | なし                                    |
| 備 考           | —                                     |

---

power\_control

|       |                          |
|-------|--------------------------|
| 概 要   | パワーコントロール処理              |
| ヘッダ   | なし                       |
| 宣 言   | void power_control(void) |
| 説 明   | パワーオフ1モードに移行します。         |
| 引 数   | なし                       |
| リターン値 | なし                       |
| 備 考   | —                        |

---

write address init

|       |                                                                                                                      |
|-------|----------------------------------------------------------------------------------------------------------------------|
| 概 要   | レコード書き込みアドレス初期設定処理                                                                                                   |
| ヘッダ   | なし                                                                                                                   |
| 宣 言   | void write_address_init(void)                                                                                        |
| 説 明   | 各ブロックの空レコードを検索し、ブロックの使用状況により使用ブロックを選択します。その後、空レコードの先頭番地を書き込みアドレス(write_addr)に設定します。また、最終書き込みレコードを読み出し、カウントデータに設定します。 |
| 引 数   | なし                                                                                                                   |
| リターン値 | なし                                                                                                                   |
| 備 考   | —                                                                                                                    |

---

write control

|       |                                                                                                                                       |
|-------|---------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | データ書き込み制御処理                                                                                                                           |
| ヘッダ   | なし                                                                                                                                    |
| 宣 言   | unsigned char write_control(void)                                                                                                     |
| 説 明   | 書き込みデータ設定処理で設定したカウントデータの書き込みを行います。                                                                                                    |
| 引 数   | なし                                                                                                                                    |
| リターン値 | ●正常終了：NORMAL(0x00)<br>●コマンドシーケンスエラー：CMD_SEQ_ERROR(0x01)<br>●イレーズ/ブランクチェックエラー：ERS_BLK_CHK_ERROR(0x02)<br>●プログラムエラー：PROGRAM_ERROR(0x03) |
| 備 考   | —                                                                                                                                     |

---

set data

|       |                                    |
|-------|------------------------------------|
| 概 要   | 書き込みデータ設定処理                        |
| ヘッダ   | なし                                 |
| 宣 言   | void set_data(unsigned char *data) |
| 説 明   | 引数のアドレスにカウントデータを設定します。             |
| 引 数   | 第一引数：data<br>：書き込みデータ先頭アドレス        |
| リターン値 | なし                                 |
| 備 考   | —                                  |

---

block\_erase

|       |                                                                                                                                                                                                                       |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | ブロックイレース処理                                                                                                                                                                                                            |
| ヘッダ   | なし                                                                                                                                                                                                                    |
| 宣 言   | unsigned char block_erase(unsigned char block_no)                                                                                                                                                                     |
| 説 明   | 指定したブロックをCPU書き換えモード(EW1モード)で消去します。<br>イレース終了後、フルステータスチェック処理を行い、結果を返します。                                                                                                                                               |
| 引 数   | 第一引数 : block_no : イレースブロック番号                                                                                                                                                                                          |
| リターン値 | <ul style="list-style-type: none"> <li>● 正常終了 : NORMAL(0x00)</li> <li>● コマンドシーケンスエラー : CMD_SEQ_ERROR(0x01)</li> <li>● イレース/ブランクチェックエラー : ERS_BLK_CHK_ERROR(0x02)</li> <li>● プログラムエラー : PROGRAM_ERROR(0x03)</li> </ul> |
| 備 考   | —                                                                                                                                                                                                                     |

---

data\_write

|       |                                                                                                                                                                                                                       |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | プログラムコマンド処理                                                                                                                                                                                                           |
| ヘッダ   | なし                                                                                                                                                                                                                    |
| 宣 言   | unsigned char data_write(unsigned char *data)                                                                                                                                                                         |
| 説 明   | 書き込みアドレス(write_addr)から1レコード分のデータをCPU書き換えモード(EW1モード)で書き込みます。1バイト書き込む毎にフルステータスチェック処理を行い、結果を返します。                                                                                                                        |
| 引 数   | 第一引数 : data<br>: 書き込みデータ先頭アドレス                                                                                                                                                                                        |
| リターン値 | <ul style="list-style-type: none"> <li>● 正常終了 : NORMAL(0x00)</li> <li>● コマンドシーケンスエラー : CMD_SEQ_ERROR(0x01)</li> <li>● イレーズ/ブランクチェックエラー : ERS_BLK_CHK_ERROR(0x02)</li> <li>● プログラムエラー : PROGRAM_ERROR(0x03)</li> </ul> |
| 備 考   | —                                                                                                                                                                                                                     |

---

full\_sts\_chk

|       |                                                                                                                                                                                                                   |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | フルステータスチェック処理                                                                                                                                                                                                     |
| ヘッダ   | なし                                                                                                                                                                                                                |
| 宣 言   | unsigned char full_sts_chk(unsigned char *chk_adr)                                                                                                                                                                |
| 説 明   | フルステータスチェックを行い、結果を返します。                                                                                                                                                                                           |
| 引 数   | 第一引数 : chk_adr<br>: イレーズコマンド、プログラムコマンドを書き込んだアドレス                                                                                                                                                                  |
| リターン値 | <ul style="list-style-type: none"> <li>●正常終了 : NORMAL(0x00)</li> <li>●コマンドシーケンスエラー : CMD_SEQ_ERROR(0x01)</li> <li>●イレーズ/ブランクチェックエラー : ERS_BLK_CHK_ERROR(0x02)</li> <li>●プログラムエラー : PROGRAM_ERROR(0x03)</li> </ul> |
| 備 考   | —                                                                                                                                                                                                                 |

## 4.9 フローチャート

### 4.9.1 メイン処理

図 4.11 にメイン処理のフローチャートを示します。

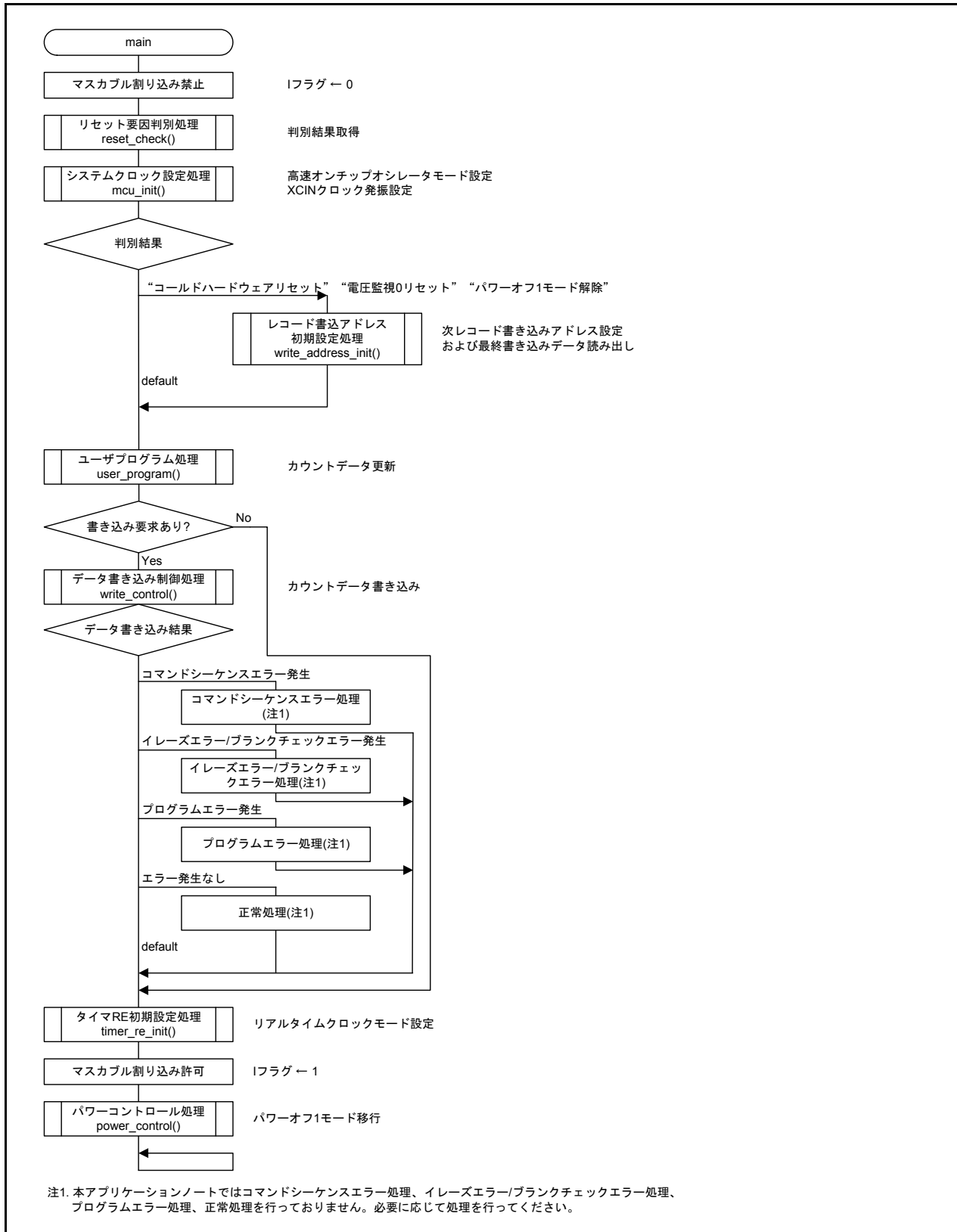


図 4.11 メイン処理

## 4.9.2 リセット要因判別処理

図 4.12 にリセット要因判別処理のフローチャートを示します。

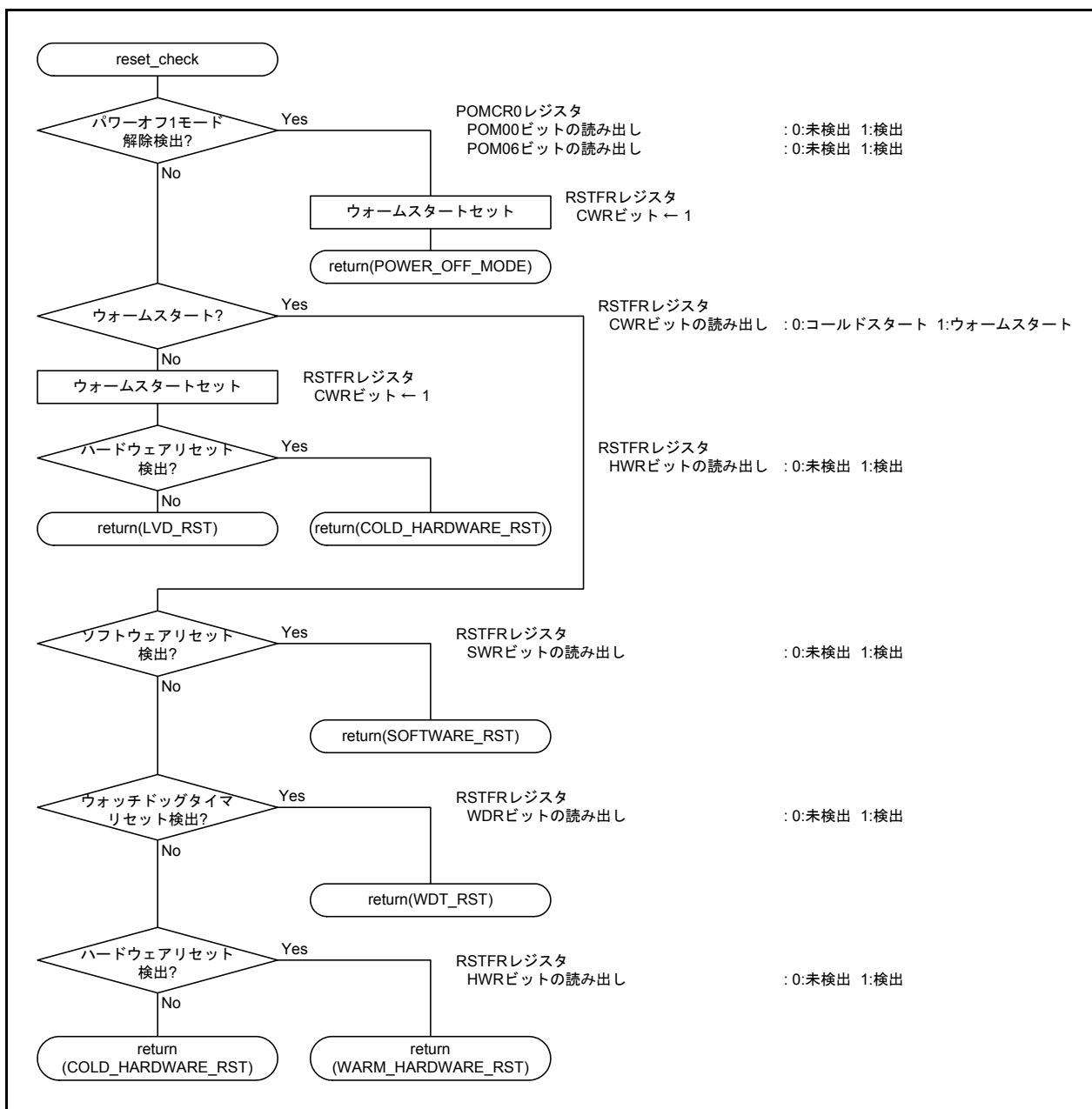


図 4.12 リセット要因判別処理

### 4.9.3 システムクロック設定処理

図 4.13 にシステムクロック設定処理のフローチャートを示します。

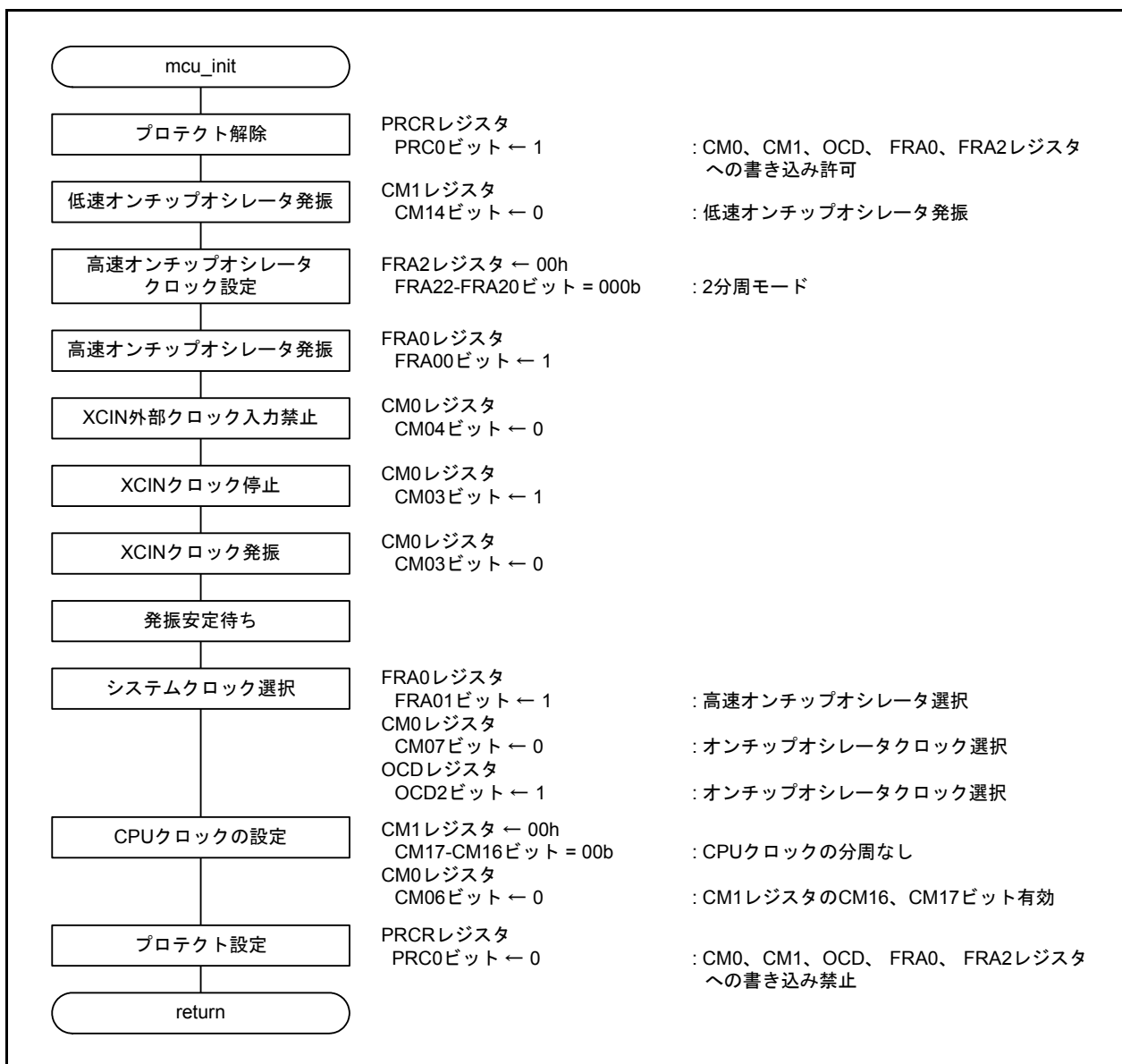


図 4.13 システムクロック設定処理

#### 4.9.4 ユーザプログラム処理

図 4.14 にユーザプログラム処理のフローチャートを示します。

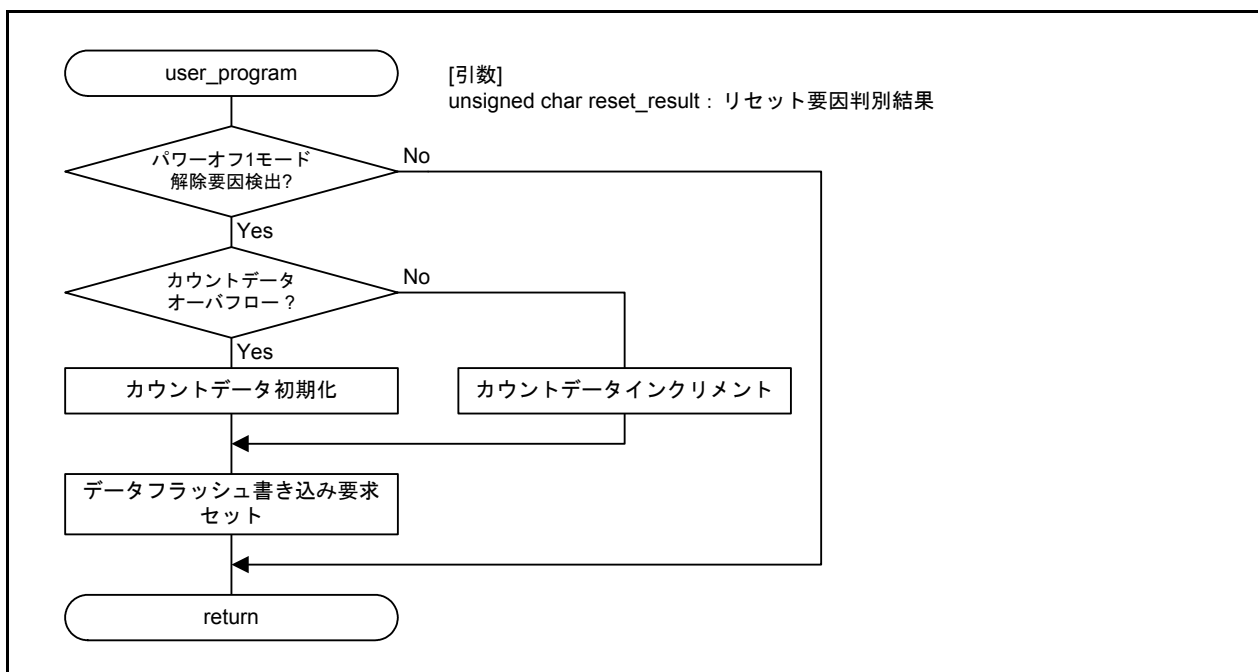


図 4.14 ユーザプログラム処理

#### 4.9.5 タイマRE初期設定処理

図 4.15 にタイマRE初期設定処理のフローチャートを示します。

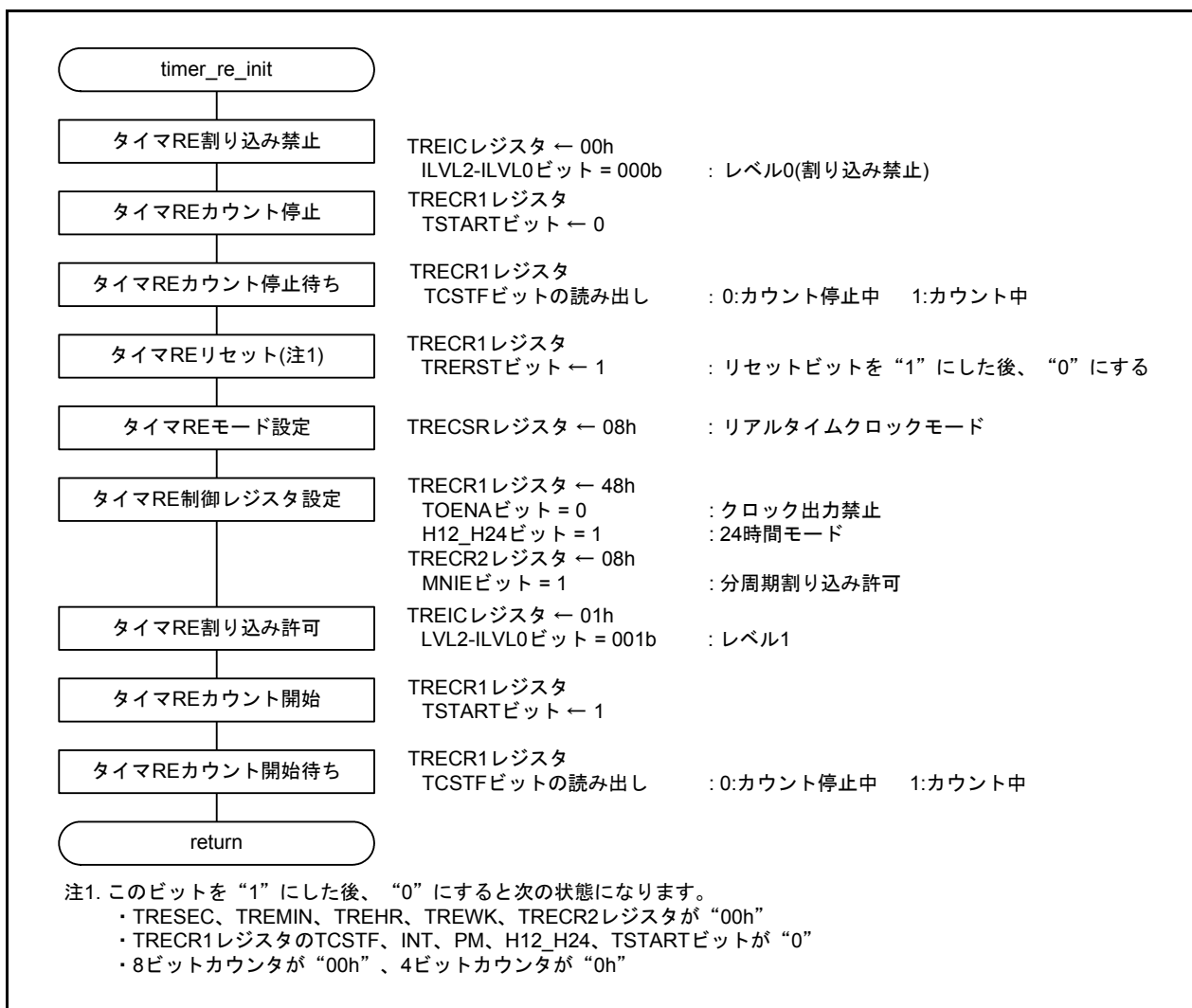


図 4.15 タイマRE初期設定処理

#### 4.9.6 パワーコントロール処理

図 4.16 にパワーコントロール処理のフローチャートを示します。

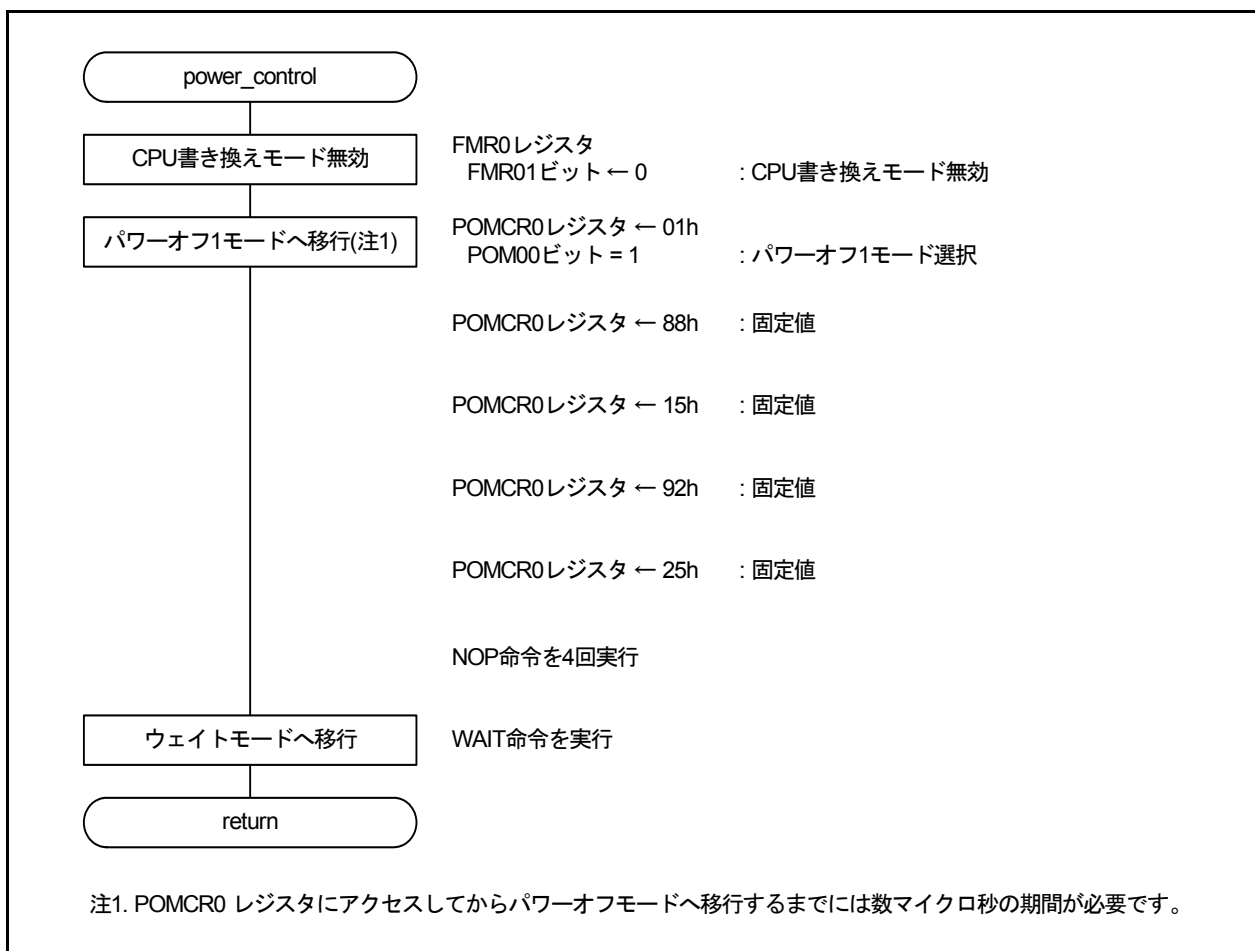


図 4.16 パワーコントロール処理

## 4.9.7 レコード書き込みアドレス初期設定処理

図 4.17～図 4.19にレコード書き込みアドレス初期設定処理のフローチャートを示します。

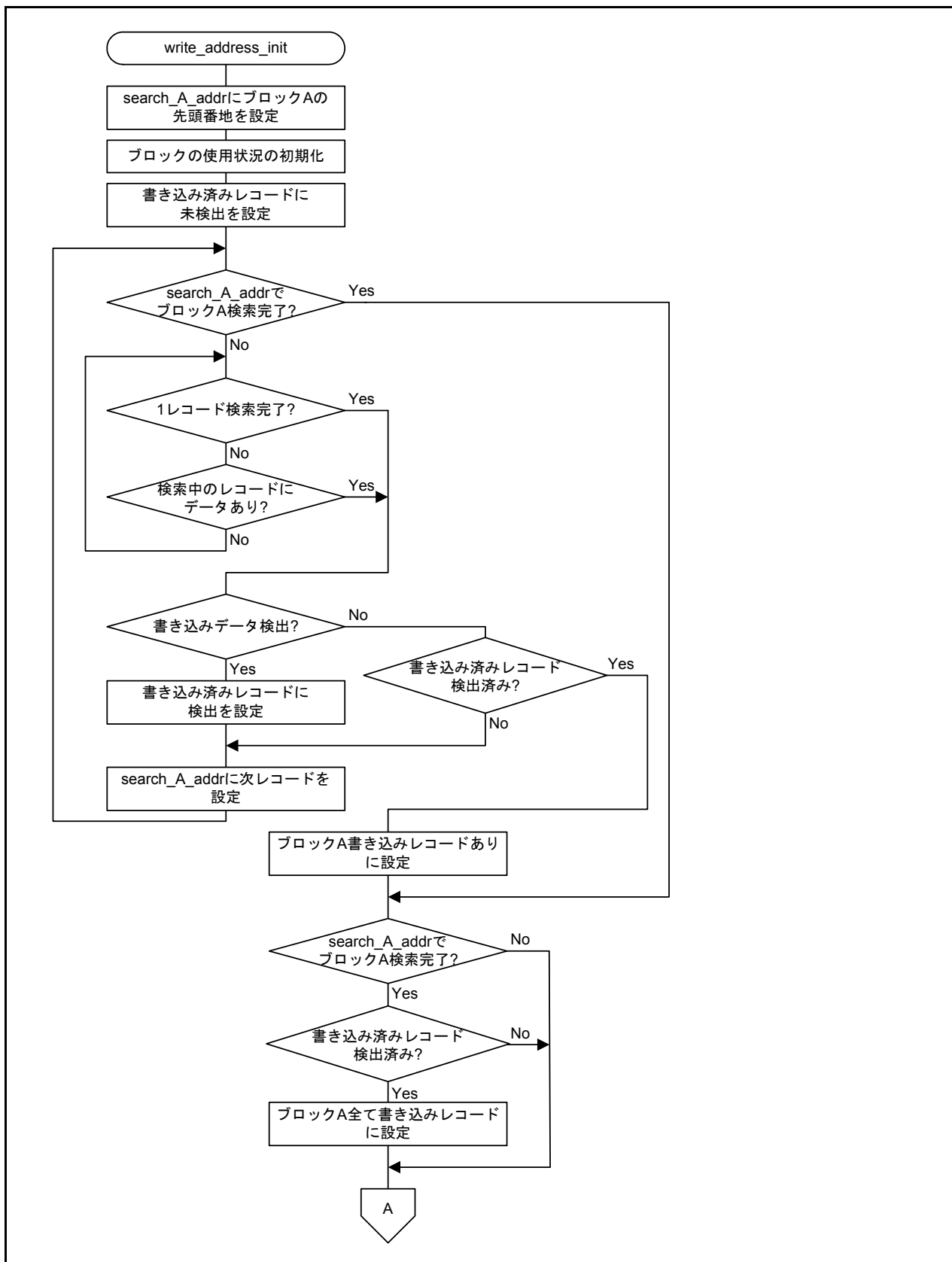


図 4.17 レコード書き込みアドレス初期設定処理 (1/3)

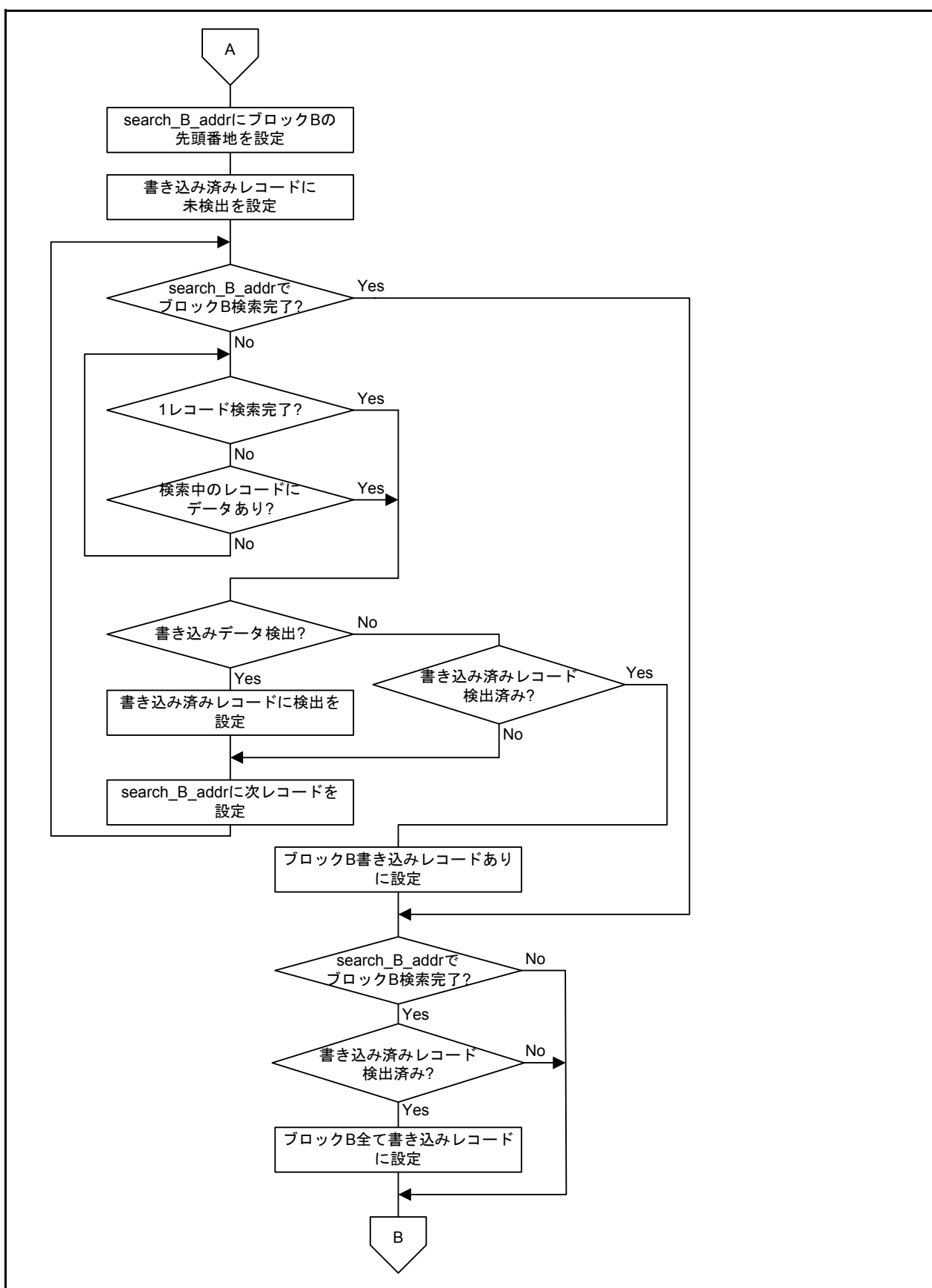


図 4.18 レコード書き込みアドレス初期設定処理 (2/3)

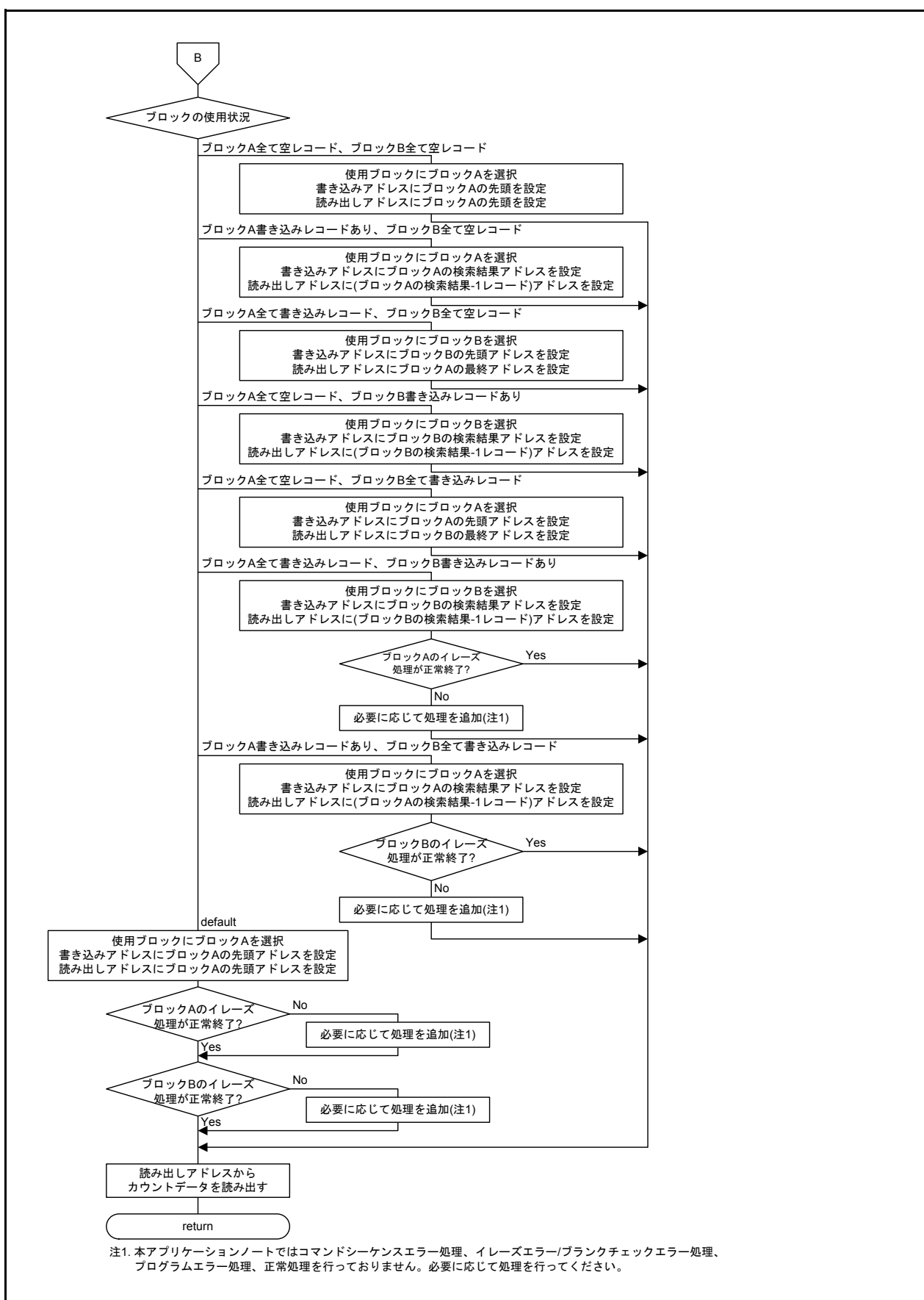


図 4.19 レコード書き込みアドレス初期設定処理 (3/3)

#### 4.9.8 データ書き込み制御処理

図 4.20 にデータ書き込み制御処理のフローチャートを示します。

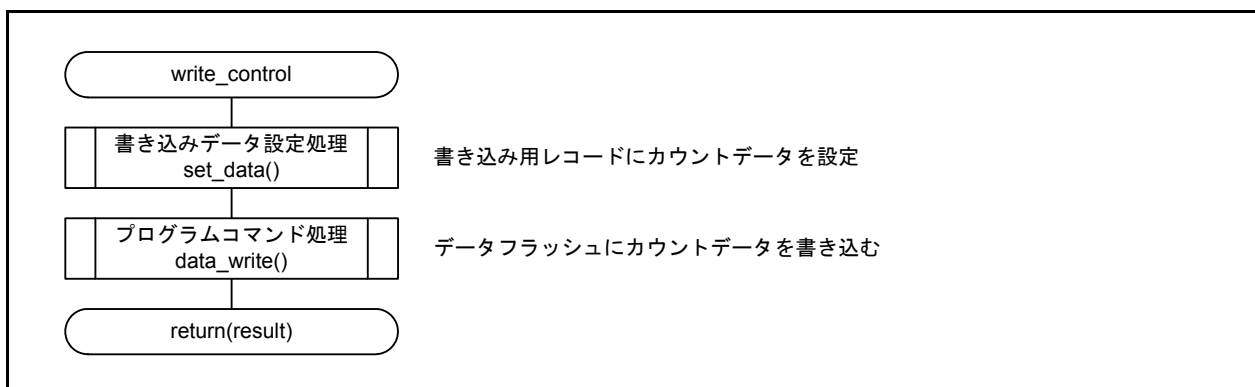


図 4.20 データ書き込み制御処理

#### 4.9.9 書き込みデータ設定処理

図 4.21 に書き込みデータ設定処理のフローチャートを示します。

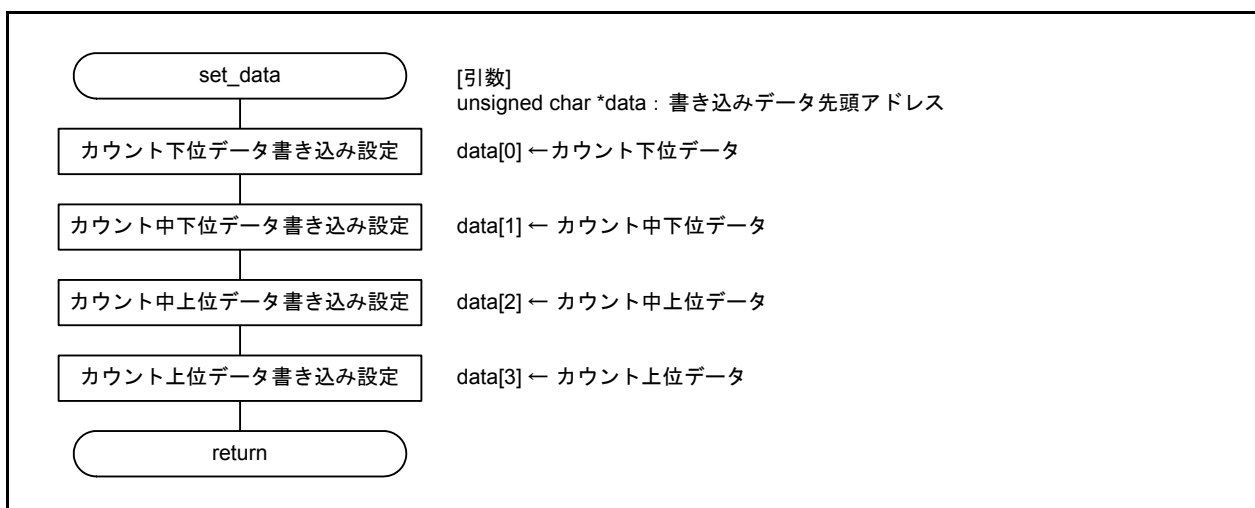


図 4.21 書き込みデータ設定処理

## 4.9.10 ブロックイレーズ処理

図 4.22 にブロックイレーズ処理のフローチャートを示します。

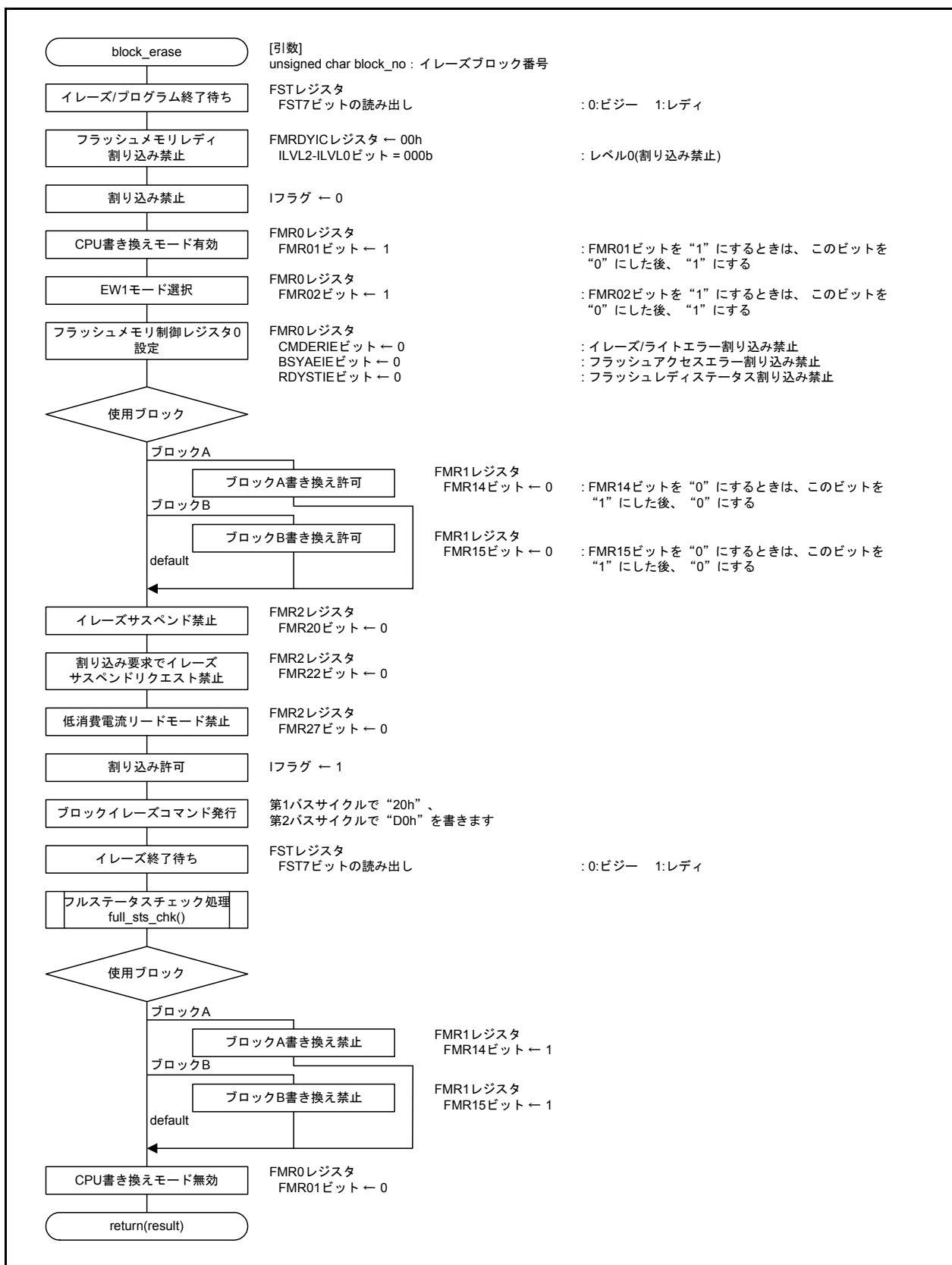


図 4.22 ブロックイレーズ処理

## 4.9.11 プログラムコマンド処理

図 4.23、図 4.24 にプログラムコマンド処理のフローチャートを示します。

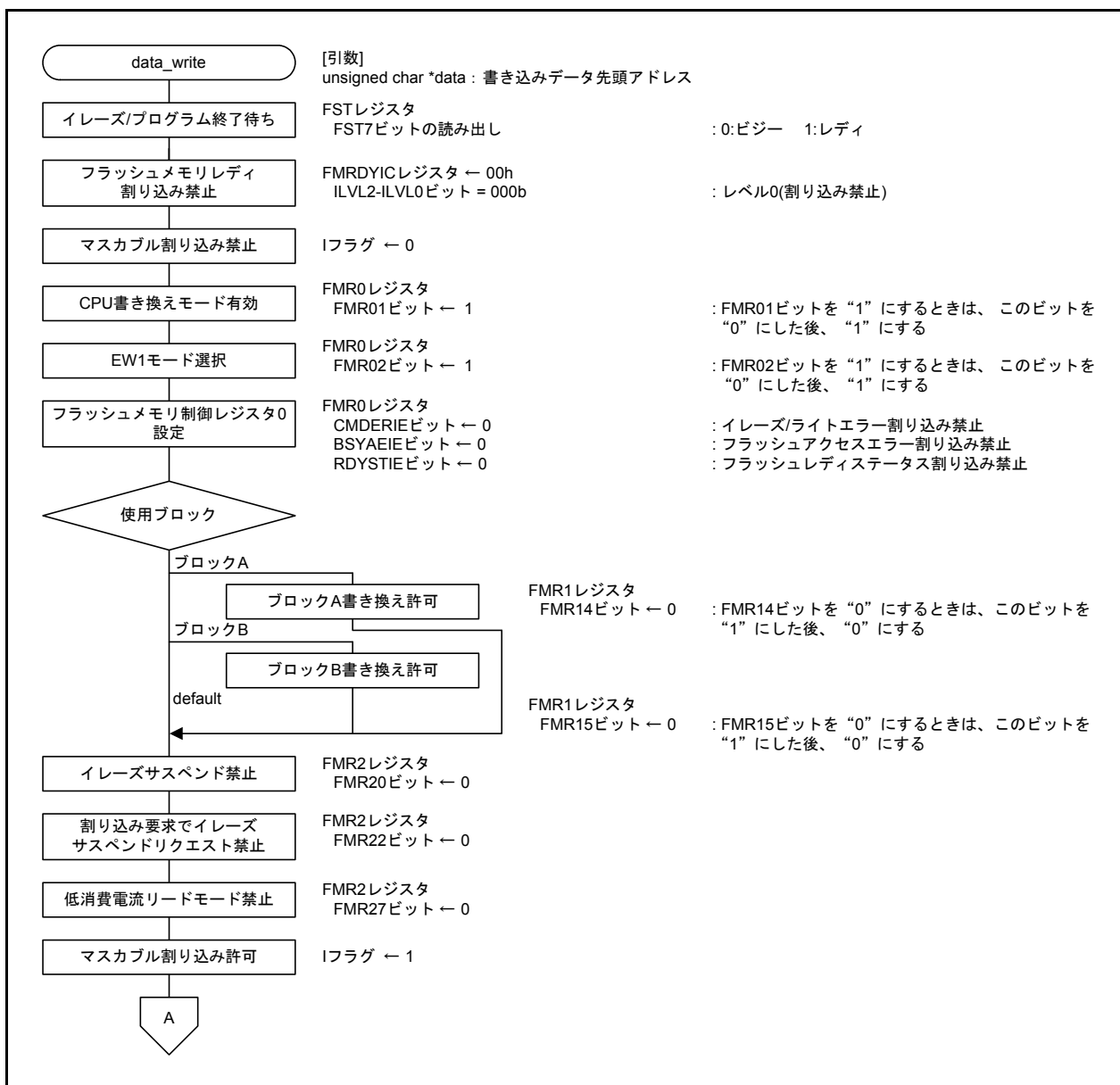


図 4.23 プログラムコマンド処理(1/2)

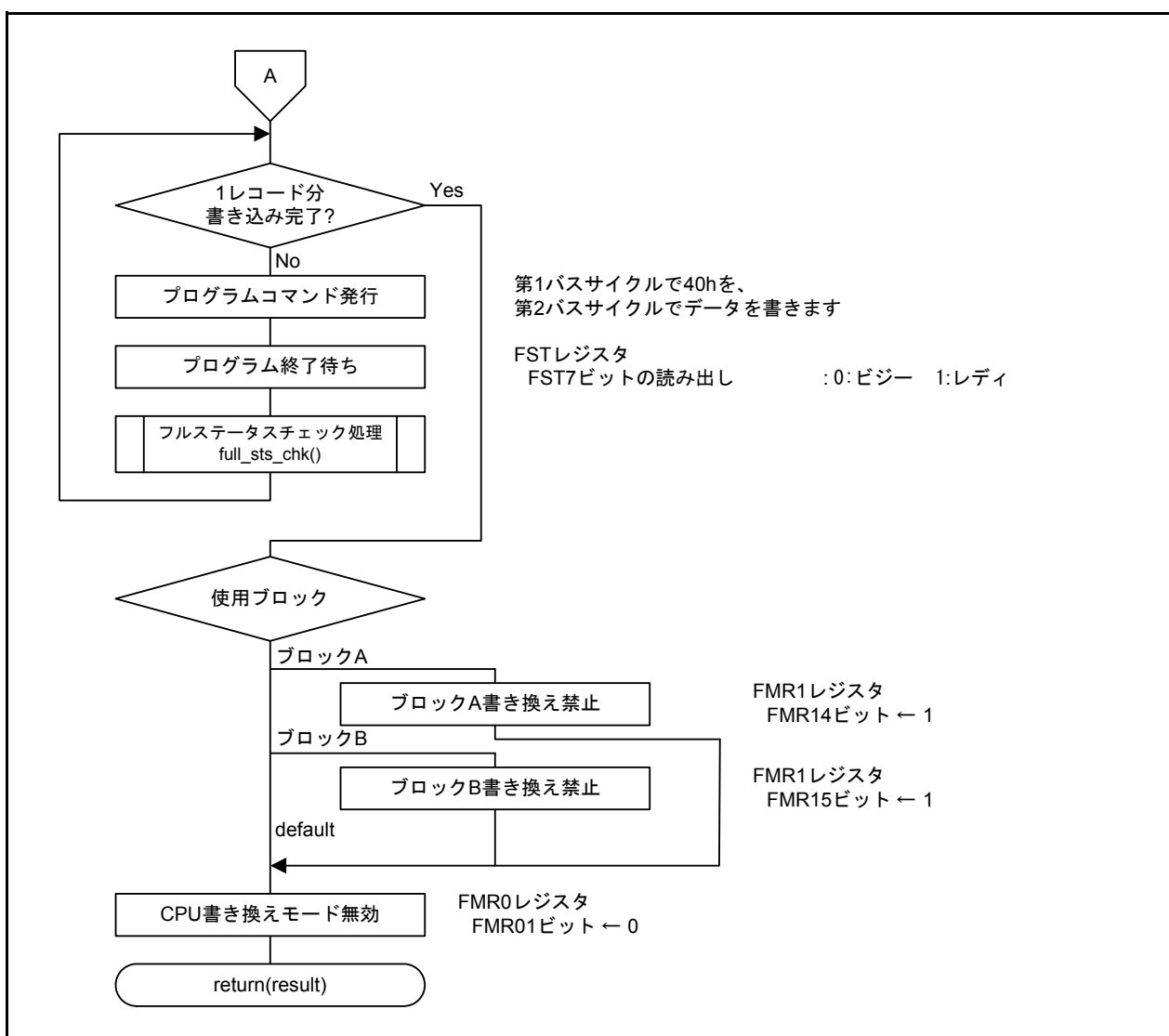


図 4.24 プログラムコマンド処理 (2/2)

## 4.9.12 フルステータスチェック処理

図 4.25 にフルステータスチェック処理のフローチャートを示します。

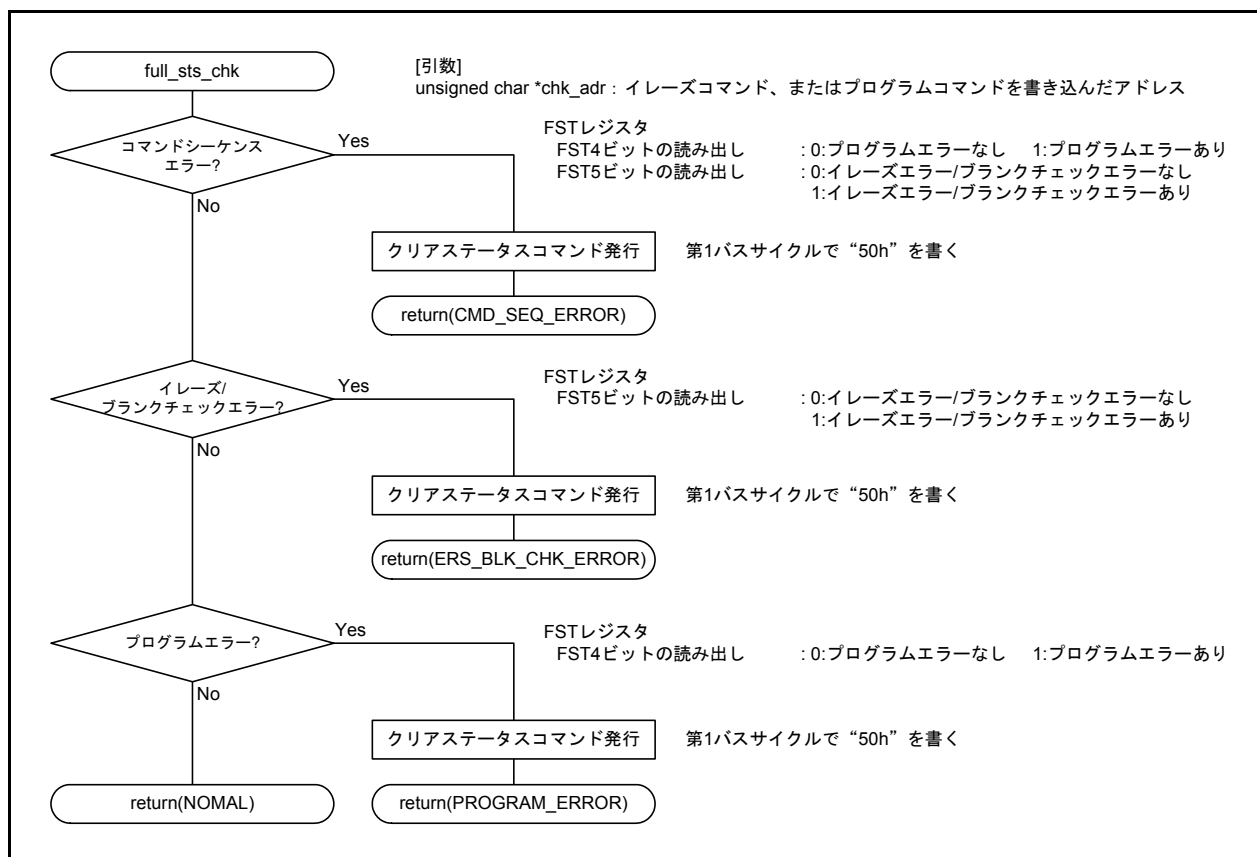


図 4.25 フルステータスチェック処理

## 5. 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。

## 6. 参考ドキュメント

R8C/L3AM 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。)

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。)

## 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>

|      |                                         |
|------|-----------------------------------------|
| 改訂記録 | R8C/L3AMグループ<br>パワーオフ1モードを使用したパワーコントロール |
|------|-----------------------------------------|

| Rev. | 発行日        | 改訂内容 |                        |
|------|------------|------|------------------------|
|      |            | ページ  | ポイント                   |
| 1.00 | 2011.11.30 | －    | 初版発行                   |
| 1.01 | 2012.1.6   | 20   | 図4.16一部削除（サンプルコード変更なし） |

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（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うと、内部 ROM、レイアウトパターンの相違などにより、電氣的特性の範囲で、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。型名が違う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. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが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・電話番号は変更することがあります。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>