

RX21Aグループ

R01AN1486JJ0110

Rev.1.10

2014.07.01

初期設定例

要旨

本アプリケーションノートでは、クロックの設定やリセット後に動作している周辺機能の停止など、リセット後に必要な設定について説明します。

対象デバイス

- ・RX21Aグループ 100ピン版 ROM容量：256KB～512KB
- ・RX21Aグループ 80ピン版 ROM容量：256KB～512KB
- ・RX21Aグループ 64ピン版 ROM容量：256KB～512KB

本アプリケーションノートの他の使い方

RX200シリーズのアプリケーションノートには対象製品がRX210グループだけのもの、複数のグループを対象としているものなどが存在します。しかし、それらに付属しているサンプルコードは、「動作確認条件」に明記されている製品のスタートアッププログラムが組み込まれおり、そのままではRX21Aグループ上では動きません。

その場合、当該スタートアッププログラムを本アプリケーションノート付属のスタートアッププログラムに差し替えることにより、RX21Aグループでも動作するようにできます。

詳細は「5. RX200シリーズのアプリケーションノートをRX21Aグループに適用する方法」を参照してください。

目次

| | |
|--|----|
| 1. 仕様 | 3 |
| 1.1 リセット後に動作している周辺機能の停止 | 3 |
| 1.2 存在しないポートの設定 | 3 |
| 1.3 クロックの設定 | 4 |
| 1.3.1 概要 | 4 |
| 1.3.2 サンプルコードで想定しているクロックの仕様 | 4 |
| 1.3.3 クロックの選択 | 5 |
| 2. 動作確認条件 | 6 |
| 3. ソフトウェア説明 | 7 |
| 3.1 リセット後に動作している周辺機能の停止 | 7 |
| 3.2 存在しないポートの設定 | 8 |
| 3.2.1 処理概要 | 8 |
| 3.2.2 ピン数の選択方法 | 8 |
| 3.3 クロックの設定 | 9 |
| 3.3.1 クロックの設定手順 | 9 |
| 3.3.2 各クロックの発振安定待機時間の考え方 | 10 |
| 3.4 ファイル構成 | 14 |
| 3.5 オプション設定メモリ | 14 |
| 3.6 定数一覧 | 15 |
| 3.7 関数一覧 | 19 |
| 3.8 関数仕様 | 20 |
| 3.9 フローチャート | 24 |
| 3.9.1 メイン処理 | 24 |
| 3.9.2 リセット後に動作している周辺機能の停止 | 24 |
| 3.9.3 存在しないポートの初期設定 | 25 |
| 3.9.4 クロック初期設定 | 26 |
| 3.9.5 メインクロックの発振設定 | 28 |
| 3.9.6 PLL クロックの発振設定 | 28 |
| 3.9.7 HOCO クロックの発振設定 | 29 |
| 3.9.8 サブクロックの発振設定 | 30 |
| 3.9.9 サブクロックの停止設定 | 31 |
| 3.9.10 CMT0によるソフトウェアウエイト | 32 |
| 4. 付録 | 33 |
| 4.1 クロックの発振安定待機時間の考え方 | 33 |
| 4.1.1 メインクロックの発振安定を待たずに PLL の発振を行う場合 | 33 |
| 5. RX200 シリーズのアプリケーションノートを RX21A グループに適用する方法 | 34 |
| 6. サンプルコード | 37 |
| 7. 参考ドキュメント | 37 |

1. 仕様

リセット後に動作している周辺機能の停止、存在しないポートの設定、クロックの設定を行います。本アプリケーションノートでは、電源投入時(コールドスタート時)の処理を想定しています。

1.1 リセット後に動作している周辺機能の停止

周辺機能によっては、電源投入後から動作しているものや、モジュールストップ機能が無効になっているものがあります。この項目に該当する処理として、

- ・DMAC、DTC、RAM0 の機能を停止する処理

を用意しています。なお、サンプルコードでは上記の処理は実行させていません。必要に応じて定数を書き換えて、処理を実行させてください。

1.2 存在しないポートの設定

端子数が 100 ピン未満の製品では、端子に接続されていないポートを出力に設定する必要があります。サンプルコードでは、端子数が 100 ピンの製品に合わせて初期値を設定しています。お使いの製品に応じて定数を書き換えてください。

1.3 クロックの設定

1.3.1 概要

クロックの設定は、下記の手順で行います。

- (1) サブクロック設定
- (2) メインクロック設定
- (3) PLLクロック設定
- (4) HOCOクロック設定
- (5) システムクロックの切り替え

本アプリケーションノートでは、`r_init_clock.h` で定義している定数を変更することで、各クロックの設定内容を切り替えます。

サンプルコードでは、システムクロックをPLLクロックとし、サブクロックを使用しない処理を実行させています。必要に応じて定数を書き換えて、使用するクロックを選択してください。

1.3.2 サンプルコードで想定しているクロックの仕様

表 1.1 にサンプルコードで想定しているクロックの仕様を示します。この表に記載している仕様の発振子に合わせて、発振安定待機時間などを算出しています。

表 1.2 に使用する周辺機能と用途を示します。

表 1.1 サンプルコードで想定しているクロックの仕様

| クロック | 発振周波数 | 発振安定時間 | 備考 |
|------------|--------------------------------|----------------------|--------|
| メインクロック発振子 | 20MHz | 4.2ms (注 2) | 水晶 |
| サブクロック発振子 | 32.768kHz (注 1) | 1.3s (注 2) | 低 CL 用 |
| PLL クロック | 100MHz (メインクロック 2 分周 10 逡倍) | 最大 500 μ s (注 3) | |
| HOCO クロック | 50MHz (注 1) | 最大 175 μ s (注 3) | |

注 1. サンプルコードでは発振を停止させています。

注 2. 発振子の発振安定時間は実際のシステムにおける配線パターン、発振定数などの条件により異なります。発振安定時間は、お客様が実際に使用されるシステムの評価を発振子メーカーに依頼して入手してください。

注 3. ユーザーズマニュアル ハードウェア編の「電気的特性」を参照してください。

表 1.2 使用する周辺機能と用途

| 周辺機能 | 用途 |
|------------------------------|----------------------|
| コンペアマッチタイマ チャンネル 0 (以下、CMT0) | クロック発振安定待機時間の測定(注 1) |

注 1. OS を使用する場合は、OS が使用しないタイマのチャンネルを選択してください。

1.3.3 クロックの選択

サンプルコードでは、`r_init_clock.h`で定義している定数を変更することで、システムクロックのクロックソース、各クロックの発振/停止などを選択できます。変更可能な定数は、表 3.10、表 3.11のサンプルコードで使用する定数(ユーザ変更可)を参照してください。

表 1.3にクロックの選択例を示します。サンプルコードでは、システムクロックをPLLとし、サブクロックを使用しない処理(No.1)を選択しています。

表1.3 クロックの選択例

| No | 1 | 2 | 3 | 4 | 5 | 6 | |
|-----------------|------------------|----------------|---------------|----------------|----------------|----------------|-------------|
| システムクロック | PLL | PLL | HOCO | HOCO | メインクロック | メインクロック | |
| PLL クロック | 発振 | 発振 | 停止 | 停止 | 停止 | 停止 | |
| メインクロック | 発振 | 発振 | 停止 | 停止 | 発振 | 発振 | |
| HOCO クロック | 停止 | 停止 | 発振 (50MHz) | 発振 (50MHz) | 停止 | 停止 | |
| サブクロック (注 2) | 停止 | 発振 (RTC 使用) | 停止 | 発振 (RTC 使用) | 停止 | 発振 (RTC 使用) | |
| 動作モード | 高速動作 モード | 高速動作 モード | 高速動作 モード | 高速動作 モード | 中速動作 モード 1A | 中速動作 モード 1A | |
| 定数 | SEL_SYSCLK | CLK_PLL | CLK_PLL | CLK_HOCO | CLK_HOCO | CLK_MAIN | CLK_MAIN |
| | SEL_PLL | B_USE | B_USE | B_NOT_USE | B_NOT_USE | B_NOT_USE | B_NOT_USE |
| | SEL_MAIN | B_USE | B_USE | B_NOT_USE | B_NOT_USE | B_USE | B_USE |
| | SEL_HOCO | B_NOT_USE | B_NOT_USE | B_USE | B_USE | B_NOT_USE | B_NOT_USE |
| | SEL_SUB (注 1) | B_NOT_USE | B_NOT_USE | B_NOT_USE | B_NOT_USE | B_NOT_USE | B_NOT_USE |
| | SEL_RTC (注 1) | B_NOT_USE | B_USE | B_NOT_USE | B_USE | B_NOT_USE | B_USE |
| | REG_OPCCR | OPCM_HIGH | OPCM_HIGH | OPCM_HIGH | OPCM_HIGH | OPCM_MID_1A | OPCM_MID_1A |

注1. システムクロックとしてサブクロックを発振させる場合は、SEL_SUB を B_USE (使用する)に、RTC のカウントソースとしてサブクロックを発振させる場合は、SEL_RTC を B_USE に設定してください。

SEL_SUB または SEL_RTC のどちらか、または両方を B_USE に設定すると、サブクロックが発振します。

注2. サブクロックの発振は、SOSCCR.SOSTP ビット (サブクロックをシステムクロックとして使用する場合に制御)、RCR3.RTCEN ビット (サブクロックを RTC のカウントソースとして使用する場合に制御) の両方で制御されます。そのため、サブクロックをシステムクロックとして使用するか否かで、サブクロックの初期設定処理が変わります。また、サブクロックは電源投入時から発振を開始しています。そのため、サブクロックを使用しない場合でもサブクロックを停止する処理を行います。

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 | R5F521A8BDFP (RX21A グループ) |
| 動作周波数 | <ul style="list-style-type: none"> システムクロックにPLLを選択した場合 (表 1.3の No.1, No2) メインクロック: 20MHz サブクロック: 32.768kHz (使用しないを選択した場合は停止) PLL: 100MHz (メインクロック 2分周 10 通倍) HOCO: 停止 システムクロック(ICLK): 50MHz (PLL 2 分周) 周辺モジュールクロック A (PCLKA): 50MHz (PLL 2 分周) 周辺モジュールクロック B (PCLKB): 25MHz (PLL 4 分周) 周辺モジュールクロック C (PCLKC): 25MHz (PLL 4 分周) 周辺モジュールクロック D (PCLKD): 25MHz (PLL 4 分周) 外部バスクロック (BCLK): 25MHz(PLL 4 分周)(注 1) FlashIF クロック (FCLK): 25MHz(PLL 4 分周) |
| | <ul style="list-style-type: none"> システムクロックにHOCOを選択した場合 (表 1.3の No.3, No4) メインクロック: 停止 サブクロック: 32.768kHz (使用しないを選択した場合は停止) PLL: 停止 HOCO: 50MHz システムクロック(ICLK): 50MHz (HOCO 1 分周) 周辺モジュールクロック A (PCLKA): 50MHz (HOCO 1 分周) 周辺モジュールクロック B (PCLKB): 25MHz (HOCO 2 分周) 周辺モジュールクロック C (PCLKC): 25MHz (HOCO 2 分周) 周辺モジュールクロック D (PCLKD): 25MHz (HOCO 2 分周) 外部バスクロック (BCLK): 25MHz(HOCO 2 分周)(注 1) FlashIF クロック (FCLK): 25MHz(HOCO 2 分周) |
| | <ul style="list-style-type: none"> システムクロックにメインクロックを選択した場合 (表 1.3の No.5, No6) メインクロック: 20MHz サブクロック: 32.768kHz (使用しないを選択した場合は停止) PLL: 停止 HOCO: 停止 システムクロック(ICLK): 20MHz (メインクロック 1 分周) 周辺モジュールクロック A (PCLKA): 20MHz (メインクロック 1 分周) 周辺モジュールクロック B (PCLKB): 20MHz (メインクロック 1 分周) 周辺モジュールクロック C (PCLKC): 20MHz (メインクロック 1 分周) 周辺モジュールクロック D (PCLKD): 20MHz (メインクロック 1 分周) 外部バスクロック (BCLK): 20MHz(メインクロック 1 分周)(注 1) FlashIF クロック (FCLK): 20MHz(メインクロック 1 分周) |
| 動作電圧 | 3.3V |
| 統合開発環境 | ルネサスエレクトロニクス製 High-performance Embedded Workshop Version 4.09.01 |
| Cコンパイラ | ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family V.1.02 Release 01 コンパイルオプション -cpu=rx200 -output=obj="\$(CONFIGDIR)\\$(FILELEAF).obj" -debug -nologo (統合開発環境のデフォルト設定を使用しています) |
| iodef.hのバージョン | Version 1.0 |
| エンディアン | リトルエンディアン |
| 動作モード | シングルチップモード |
| 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 | 株式会社北斗電子製 HSB シリーズ マイコンボード (製品型名: HSBRX21AP-B) |

注1. RX21Aグループには外部バスがありませんが、外部バスクロック(BCLK)選択ビットの設定が必要です。

3. ソフトウェア説明

リセット後に動作している周辺機能の停止、存在しないポートの設定を行ったのち、クロックの設定を行います。

3.1 リセット後に動作している周辺機能の停止

リセット後に動作している周辺機能の停止を行います。

リセット解除後、以下の周辺モジュールに限り、モジュールストップ状態が解除されています。モジュールストップ状態へ遷移する場合は、モジュールストップビットを“1” (モジュールストップ状態へ遷移)に設定してください。このモジュールストップを行うことで消費電力を低減できます。

サンプルコードでは定数「MSTP_STATE_対象モジュール名」の値を“0 (MODULE_STOP_DISABLE)”とし、対象モジュールはモジュールストップ状態に遷移していません。使用するシステムに応じてモジュールストップ状態へ遷移したい場合は、r_init_stop_module.hの定数の値を“1 (MODULE_STOP_ENABLE)”に設定してください。

表3.1にリセット後にモジュールストップ状態が解除されている周辺モジュール一覧を示します。

表3.1 リセット後にモジュールストップ状態が解除されている周辺モジュール一覧

| 周辺モジュール | モジュールストップ設定ビット | リセット後の値 | このモジュールを使用しない場合の設定 |
|----------|---------------------|-----------------------|-----------------------|
| DMAC/DTC | MSTPCRA.MSTPA28 ビット | 0 (モジュールストップ状態の解除) | 1 (モジュールストップ状態へ遷移) |
| RAM0 | MSTPCRC.MSTPC0 ビット | | |

3.2 存在しないポートの設定

3.2.1 処理概要

100ピン版未満の製品を使用する場合、存在しないポートのPDRレジスタの対応ビットを“1”(出力)に設定します。本関数をコールした後に、存在しないポートを含むPDRレジスタまたはPODRレジスタへバイト単位で書き込む場合、存在しないポートの方向制御ビットには“1”、ポート出力データ格納ビットには“0”を設定してください。

表3.2に存在しないポート一覧を示します。

表3.2 存在しないポート一覧

| ポート シンボル | 80ピン版の製品 | 本数 | 64ピン版の製品 | 本数 |
|-------------|----------|----|-------------|----|
| PORT0 | — | — | P07 | 1 |
| PORT1 | — | — | P12、P13 | 2 |
| PORT2 | P22~P25 | 4 | P20~P25 | 6 |
| PORT3 | P33 | 1 | P33、P34 | 2 |
| PORT4 | — | — | P42、P43 | 2 |
| PORT5 | P50~P53 | 4 | P50~P53 | 4 |
| PORTA | PA7 | 1 | PA2、PA5、PA7 | 3 |
| PORTB | — | — | PB2、PB4 | 2 |
| PORTC | PC0、PC1 | 2 | PC0、PC1 | 2 |
| PORTE | PE6、PE7 | 2 | PE6、PE7 | 2 |
| PORTH | — | — | — | — |
| PORTJ | PJ3 | 1 | PJ1、PJ3 | 2 |

3.2.2 ピン数の選択方法

サンプルコードでは、100ピン版(PIN_SIZE=100)に設定しています。また、本アプリケーションノートで対応しているピン数は、100ピン、80ピン、64ピンです。100ピン以外の製品を使用する場合は、r_init_non_existent_port.hのPIN_SIZEを使用するピン数に変更してください。

3.3 クロックの設定

3.3.1 クロックの設定手順

表 3.3にクロックの設定手順とそれぞれの処理内容、およびサンプルコードでの設定を示します。サンプルコードでは、メインクロック、PLL を動作させて、HOCO、サブクロックを停止させています。

表3.3 クロックの設定手順

| 手順 | 処理 | 処理内容 | | サンプルコードの設定 |
|----|----------------------|-------------------------------------|---|--|
| 1 | サブクロック 設定(注 2) | 使用しない | サブクロック制御回路の初期化を行います。 | サブクロックを 使用しない |
| | | 使用する | サブクロック制御回路の初期化、駆動能力の設定、およびサブクロックの出力を内部クロックに供給するまでの待機時間を SOSCWTCCR レジスタに設定してから、サブクロックを発振します。 その後、ソフトウェアによる発振安定待機時間待ち(注 1)を行います。 | |
| 2 | メインクロック 設定(注 2) | 使用しない | 設定は不要です。 | メインクロックを 使用する |
| | | 使用する | メインクロックのドライブ能力の設定、およびメインクロックの出力を内部クロックに供給するまでの待機時間を MOSCWTCR レジスタに設定してから、メインクロックを発振します。 その後、ソフトウェアによる発振安定待機時間待ち(注 1)を行います。 | |
| 3 | PLL クロック 設定(注 2) | 使用しない | PLL の電源を OFF にします。 | PLL クロックを 使用する |
| | | 使用する | PLL 入力分周比および周波数通倍率の設定、および PLL クロックの出力を内部クロックに供給するまでの待機時間を PLLWTCR に設定してから、PLL クロックを発振します。 その後、ソフトウェアによる発振安定待機時間待ち(注 1)を行います。 | |
| 4 | HOCO クロック 設定(注 2) | 使用しない | HOCO の電源を OFF にします。 | HOCO クロックを 使用しない |
| | | 使用する | HOCO の周波数の設定、および HOCO クロックの出力を内部クロックに供給するまでの待機時間を HOCOWTCR2 レジスタに設定してから、HOCO クロックを発振します。 その後、ソフトウェアによる発振安定待機時間待ち(注 1)を行います。 | |
| 5 | 動作電力制御 モード設定 | 使用する動作周波数、動作電圧に応じて、動作電力制御モードを設定します。 | | 高速動作モードに設定 |
| 6 | クロック分周比 設定 | クロック分周の変更を行います。 | | ・ ICLK, PCLKA: 2 分周 ・ PCLKB, PCLKC, PCLKD, BCLK, FCLK: 4 分周 |
| 7 | システムクロック 切り替え | 使用するシステムに応じて切り替えます。 | | PLL クロックに 切り替え |

注1. 各クロックの発振安定待機時間の考え方は「3.3.2 各クロックの発振安定待機時間の考え方」を参照してください。

注2. 各クロックを使用する／使用しないの選択は、必要に応じて `r_init_clock.h` の定数を変更してください。

3.3.2 各クロックの発振安定待機時間の考え方

メインクロック、PLLクロック、サブクロック、HOCOクロックのウェイトコントロールレジスタおよび発振安定待機時間の考え方を示します。

3.3.2.1 メインクロックの発振安定待機時間の考え方

図3.1にメインクロックの発振安定待機時間の考え方、表3.4にメインクロックのウェイトコントロールと発振安定待機時間の算出方法と設定値を示します。

メインクロックのウェイトコントロールレジスタ(MOSCWTCR)は「発振子メーカーが推奨するメインクロックの発振安定時間以上の値」を設定します。また、メインクロックの発振安定待機時間は「MOSCWTCRレジスタで設定したサイクル期間の2倍以上」を設定します。

サンプルコードで使用するメインクロックの発振安定時間は4.2msのため、サンプルコードでのウェイトコントロールレジスタ設定値は0Dh(約6.55ms)、発振安定待機時間は約13.1msです。

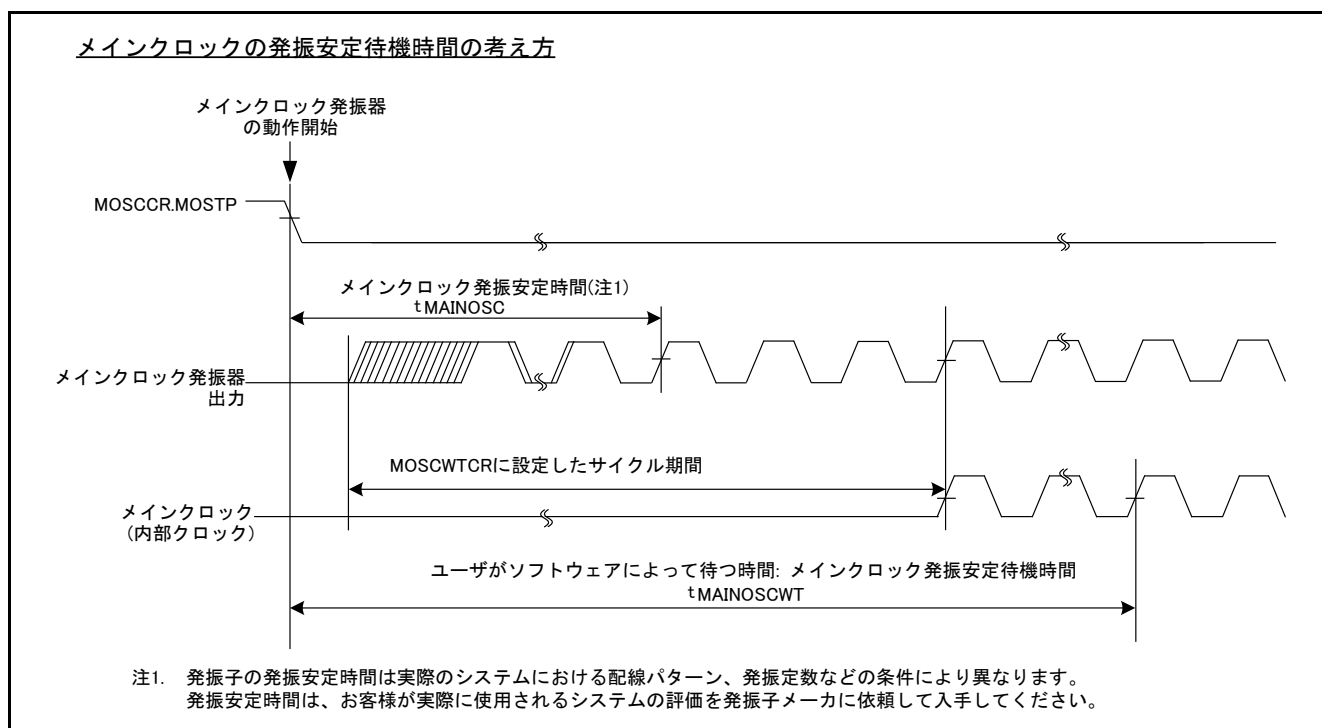


図3.1 メインクロックの発振安定待機時間の考え方

表3.4 メインクロックのウェイトコントロールと発振安定待機時間の算出方法と設定値

| ウェイトコントロール ／発振安定待機時間 | 算出方法 | サンプルコード の設定値 |
|---------------------------------------|------------------------------------|--------------------|
| ウェイトコントロール レジスタ (MOSCWTCR.MSTS) | 発振子メーカーが推奨するメインクロックの発振安定時間以上の 値 | 0Dh (約6.5536ms) |
| 発振安定待機時間 ($t_{MAINOSCWT}$) | MOSCWTCRレジスタに設定したサイクル期間の2倍以上の値 | 約13.1072ms |

3.3.2.2 PLL の発振安定待機時間の考え方(メインクロックの発振安定待機時間経過後に PLL の発振を行う場合)

図 3.2にPLL の発振安定待機時間の考え方、表 3.5にPLL のウェイトコントロールと発振安定待機時間の算出方法と設定値を示します。

PLL のウェイトコントロールレジスタ(PLLWTCR)は「 t_{PLL1} (最大 $500\mu s$)以上の値」を設定します。また、PLL の発振安定待機時間は「 t_{PLLWT1} (最小 $1.5ms$)以上の値」を設定します。

PLL の発振安定時間は最大 $500\mu s$ のため、サンプルコードでの設定値は、ウェイトコントロールレジスタは $09h$ (約 $655.36\mu s$)、発振安定待機時間は約 $1.50ms$ となります。

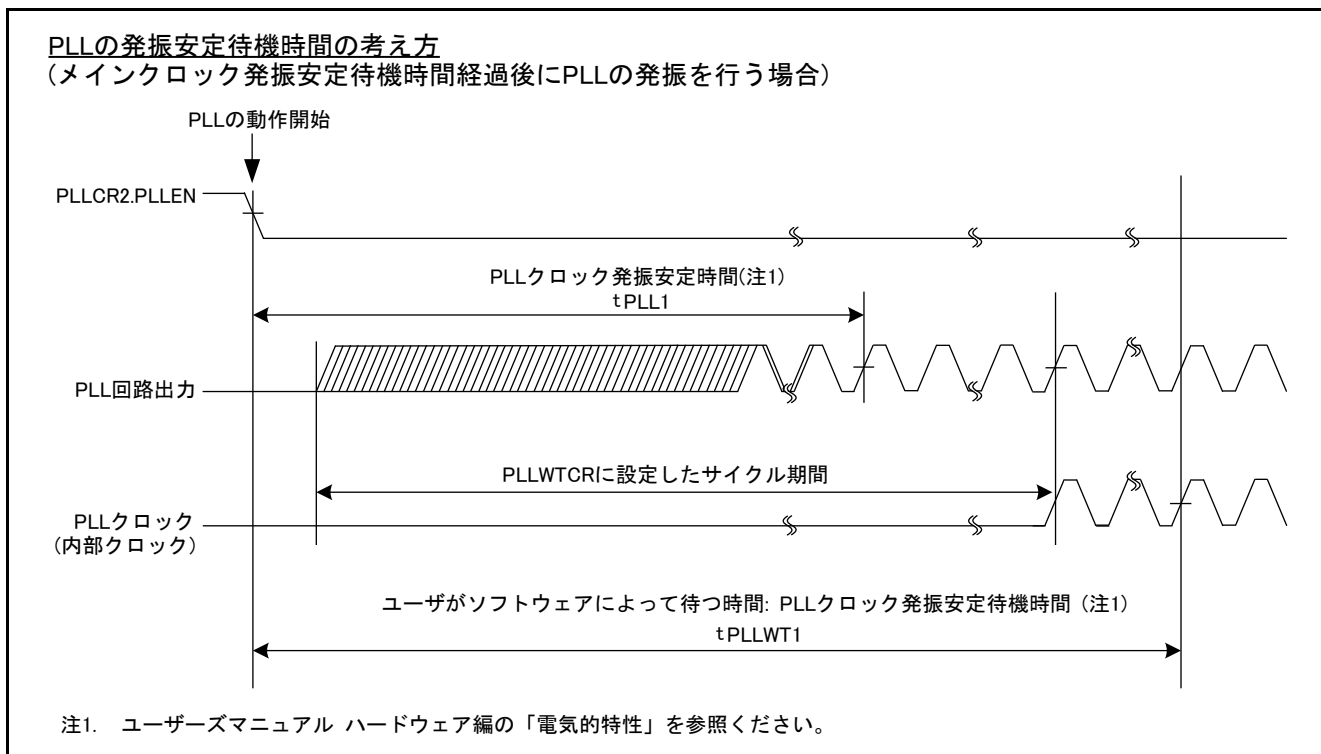


図3.2 PLL の発振安定待機時間の考え方

表3.5 PLL のウェイトコントロールと発振安定待機時間の算出方法と設定値

| ウェイトコントロール ／発振安定待機時間 | 算出方法 | サンプルコード の設定値 |
|--------------------------------------|---------------------------------|-----------------------------|
| ウェイトコントロール レジスタ (PLLWTCR.PSTS) | t_{PLL1} (最大 $500\mu s$)以上の値 | $09h$ (約 $655.36\mu s$) |
| 発振安定待機時間 (t_{PLLWT1}) | t_{PLLWT1} (最小 $1.5ms$)以上の値 | 約 $1.50ms$ |

3.3.2.3 サブクロックの発振安定待機時間の考え方

図 3.3にサブクロックの発振安定待機時間の考え方を、表 3.6にサブクロックのウェイトコントロールと発振安定待機時間の算出方法と設定値を示します。

サブクロックのウェイトコントロールレジスタ(SOSCWTCR)は「発振子メーカーが推奨するサブクロックの発振安定時間以上の値」を設定します。また、サブクロックの発振安定待機時間は「SOSCWTCR レジスタで設定した時間の2倍以上」を設定します。

サンプルコードで使用するサブクロックの発振安定時間は 1.3s のため、サンプルコードでの設定値は、ウェイトコントロールレジスタは 00h (2s + 約 61 μ s)、発振安定待機時間は約 4s となります。

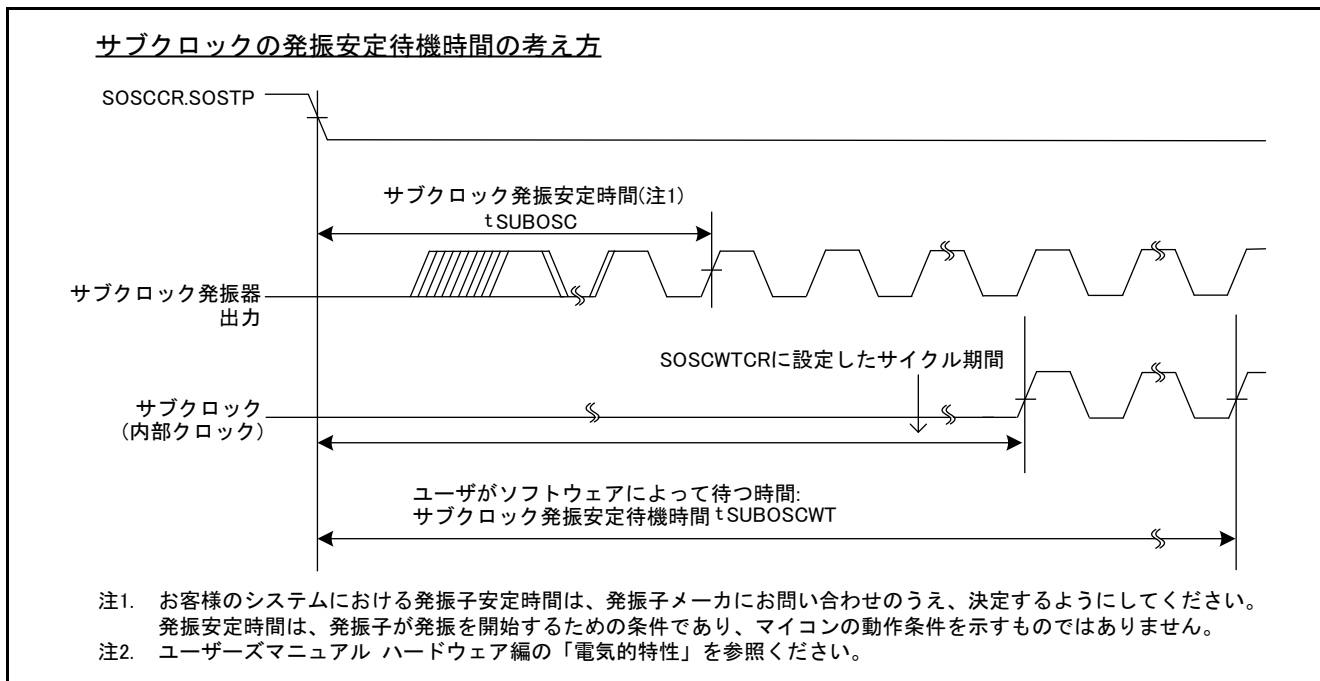


図3.3 サブクロックの発振安定待機時間の考え方

表3.6 サブクロックのウェイトコントロールと発振安定待機時間の算出方法と設定値

| ウェイトコントロール ／発振安定待機時間 | 算出方法 | サンプルコード の設定値 |
|---------------------------------------|---|----------------------------|
| ウェイトコントロール レジスタ (SOSCWTCR.SSTS) | 発振子メーカーが推奨するサブクロックの発振安定時間以上の値 | 00h (2s + 約 61 μ s) |
| 発振安定待機時間 ($t_{SUBOSCWT}$) | ウェイトコントロールレジスタで設定した時間(2s + SOSCWTCR レジスタで設定したサイクル期間)の2倍以上の値 | 約 4s |

3.3.2.4 HOCO クロックの発振安定待機時間の考え方

図 3.4にHOCO クロックの発振安定待機時間の考え方を、表 3.7にHOCO クロックのウェイトコントロールと発振安定待機時間の算出方法と設定値を示します。

HOCO クロックのウェイトコントロールレジスタ(HOCOWTCR2)は HOCO クロックの周波数が 50MHz 以外(32MHz / 36.864MHz / 40MHz)の場合は「02h (7168 サイクル)」を、50MHz の場合は「03h (9216 サイクル)」を設定します。また、HOCO クロックの発振安定待機時間は「tHOCOWT (最大 350μs)以上」を設定します。

サンプルコードで使用する HOCO クロックの周波数は 50MHz のため、サンプルコードでの設定値は、ウェイトコントロールレジスタは 03h (約 184.32μs)、発振安定待機時間は約 350μs となります。

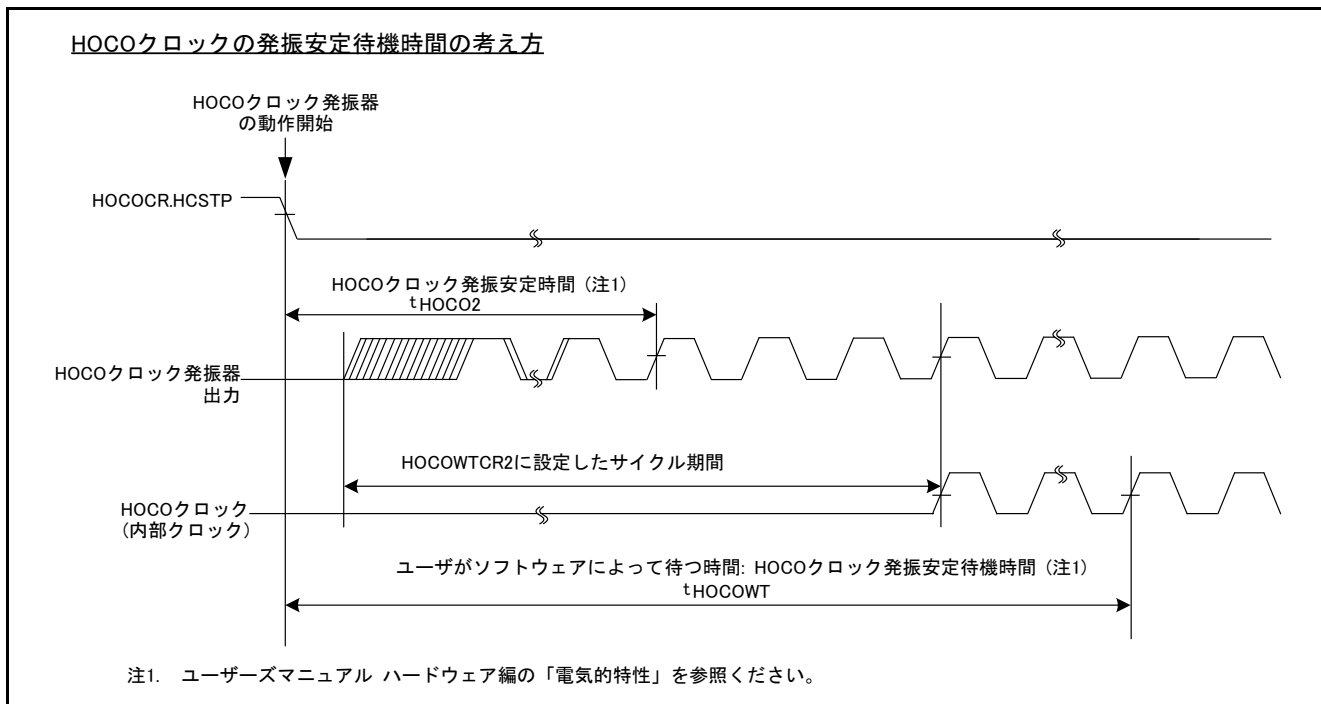


図3.4 HOCO クロックの発振安定待機時間の考え方

表3.7 HOCO クロックのウェイトコントロールと発振安定待機時間の算出方法と設定値

| ウェイトコントロール ／発振安定待機時間 | 算出方法 | サンプルコード の設定値 |
|---|--|---------------------|
| ウェイトコントロール レジスタ (HOCOWTCR2.HSTS2) | HOCO クロックの周波数が 50MHz 以外の場合 : 02h (7168 サイクル) HOCO クロックの周波数が 50MHz の場合 : 03h (9216 サイクル) | 03h (約 184.32μs) |
| 発振安定待機時間 (tHOCOWT) | tHOCOWT (最大 350μs)以上の値 | 約 350μs |

3.4 ファイル構成

表 3.8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3.8 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 |
|----------------------------|-------------------------------------|----|
| main.c | メイン処理 | |
| r_init_stop_module.c | リセット後に動作している周辺機能の停止 | |
| r_init_stop_module.h | r_init_stop_module.c のヘッダファイル | |
| r_init_non_existent_port.c | 存在しないポートの初期設定 | |
| r_init_non_existent_port.h | r_init_non_existent_port.c のヘッダファイル | |
| r_init_clock.c | クロック初期設定 | |
| r_init_clock.h | r_init_clock.c のヘッダファイル | |

3.5 オプション設定メモリ

表 3.9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。

表3.9 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、IWDT 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 0 リセット無効 リセット後、HOCO 発振が無効 |
| MDES | FFFF FF83h~FFFF FF80h | FFFF FFFFh | リトルエンディアン |

3.6 定数一覧

表 3.10、表 3.11にサンプルコードで使用する定数(ユーザ変更可)を、表 3.12にサンプルコードで使用する定数(ユーザ変更不可)を、表 3.13に100ピン版(PIN_SIZE=100)の場合での定数を、表 3.14に80ピン版(PIN_SIZE=80)の場合での定数を、表 3.15に64ピン版(PIN_SIZE=64)の場合での定数を示します。

表3.10 サンプルコードで使用する定数(ユーザ変更可) (1/2)

| 定数名 | 設定値 | 内容 |
|--------------------------------------|-----------------|---|
| SEL_MAIN (注 1) | B_USE | メインクロックの発振/停止選択 B_USE : 使用する (メインクロック発振) B_NOT_USE : 使用しない (メインクロック停止) |
| MAIN_CLOCK_Hz (注 1) | 20,000,000 L | メインクロックの発振子周波数(Hz) |
| REG_MOFPCR (注 1) | 30h | メインクロック発振器のドライブ能力の設定 (MOFPCR レジスタの設定値) |
| REG_MOSCWTCR(注 1) | 0Dh | メインクロックのウェイトコントロールレジスタの設定値 |
| WAIT_TIME_FOR_MAIN_oscillation (注 1) | 13,107,200 L | メインクロックの発振安定待機時間(ns) |
| SEL_SUB (注 1、注 2) | B_NOT_USE | サブクロック使用選択(システムクロックとして使用) B_USE : 使用する B_NOT_USE : 使用しない |
| SEL_RTC (注 1、注 2) | B_NOT_USE | サブクロック使用選択(RTC のカウントソースとして使用) B_USE : 使用する B_NOT_USE : 使用しない |
| SUB_CLOCK_Hz (注 1) | 32,768 L | サブクロックの発振子周波数(Hz) |
| REG_SOSCWTCR (注 1) | 00h | サブクロックのウェイトコントロールレジスタの設定値 |
| WAIT_TIME_FOR_SUB_oscillation (注 1) | 4,000,000,000 L | サブクロックの発振安定待機時間(ns) |
| REG_RCR3 (注 1) | CL_LOW | サブクロック発振器のドライブ能力の選択 CL_STD : 標準 CL 用ドライブ能力 CL_LOW : 低 CL 用ドライブ能力 |
| SEL_CNTMD (注 1) | CNTMD_CAL | リアルタイムクロックのカウントモード選択 CNTMD_CAL: カレンダーカウントモード CNTMD_BIN: バイナリカウントモード |
| SEL_PLL (注 1) | B_USE | PLL クロックの発振/停止選択 B_USE : 使用する (PLL クロック発振) B_NOT_USE : 使用しない (PLL クロック停止) |
| REG_PLLCR (注 1) | 0901h | PLL の入力分周比、周波数通倍率の設定 (PLLCR レジスタの設定値) |
| REG_PLLWTCR (注 1) | 09h | PLL のウェイトコントロールレジスタの設定値 |
| WAIT_TIME_FOR_PLL_oscillation (注 1) | 1,500,000 L | PLL クロックの発振安定待機時間(ns) |

注1. 使用するシステムに応じて「r_init_clock.h」で設定値を変更してください。

注2. SEL_SUB と SEL_RTC のどちらか、または両方に B_USE(使用する)を設定すると、サブクロックの発振設定を行います。

表3.11 サンプルコードで使用する定数(ユーザ変更可) (2/2)

| 定数名 | 設定値 | 内容 |
|--------------------------------------|---------------------|---|
| SEL_HOCO (注 1) | B_NOT_USE | HOCO クロックの発振/停止選択 B_USE : 使用する (HOCO クロック発振) B_NOT_USE : 使用しない (HOCO クロック停止) |
| REG_HOCOOCR2 (注 1) | FREQ_50MHz | HOCO クロックの周波数の選択 FREQ_32MHz: 32MHz FREQ_36MHz: 36.864MHz FREQ_40MHz: 40MHz FREQ_50MHz: 50MHz |
| WAIT_TIME_FOR_HOCO_oscillation (注 1) | 350,000 L | HOCO クロックの発振安定待機時間(ns) |
| SEL_SYSCLK (注 1) | CLK_PLL | システムクロックのクロックソース選択 CLK_PLL : PLL CLK_HOCO : HOCO CLK_MAIN : メインクロック CLK_SUB : サブクロック |
| REG_OPCCR (注 1) | OPCM_HIGH | 動作電力制御モード選択(注 6) OPCM_HIGH : 高速動作モード OPCM_MID_1A : 中速動作モード 1A OPCM_MID_1B : 中速動作モード 1B OPCM_MID_2A : 中速動作モード 2A OPCM_MID_2B : 中速動作モード 2B OPCM_LOW_1 : 低速動作モード 1 (注 4) OPCM_LOW_2 : 低速動作モード 2 (注 5) |
| MSTP_STATE_DMADCDC (注 2) | MODULE_STOP_DISABLE | DMAC、DTC のモジュールストップ状態選択 MODULE_STOP_DISABLE : 解除 MODULE_STOP_ENABLE : 遷移 |
| MSTP_STATE_RAM0 (注 2) | MODULE_STOP_DISABLE | RAM0 のモジュールストップ状態選択 MODULE_STOP_DISABLE : 動作 MODULE_STOP_ENABLE : 停止 |
| PIN_SIZE (注 3) | 100 | 使用する製品のピン数 |

注1. 使用するシステムに応じて「r_init_clock.h」で設定値を変更してください。

注2. 使用するシステムに応じて「r_init_stop_module.h」で設定値を変更してください。

注3. 使用するシステムに応じて「r_init_non_existent_port.h」で設定値を変更してください。

注4. PLL を発振する設定にした場合、低速動作モード 1 は選択できません。

注5. PLL を発振する、または HOCO を発振する設定にした場合、低速動作モード 2 は選択できません。

注6. 動作モードによって、動作周波数範囲および動作電圧範囲が異なります。詳細はユーザーズマニュアル ハードウェア編を参照ください。

表3.12 サンプルコードで使用する定数(ユーザ変更不可)

| 定数名 | 設定値 | 内容 |
|------------------------|--|--|
| B_NOT_USE | 0 | 使用しない |
| B_USE | 1 | 使用する |
| CL_LOW | 02h | サブクロック: 低 CL 用ドライブ能力 |
| CL_STD | 0Ch | サブクロック: 標準 CL 用ドライブ能力 |
| CNTMD_CAL | 0 | RTC: カレンダカウントモード |
| CNTMD_BIN | 1 | RTC: バイナリカウントモード |
| FREQ_32MHz | 00h | HOCO 周波数: 32MHz |
| FREQ_36MHz | 01h | HOCO 周波数: 36.864MHz |
| FREQ_40MHz | 02h | HOCO 周波数: 40MHz |
| FREQ_50MHz | 03h | HOCO 周波数: 50MHz |
| CLK_PLL | 0400h | クロックソース: PLL |
| CLK_HOCO | 0100h | クロックソース: HOCO |
| CLK_SUB | 0300h | クロックソース: サブクロック |
| CLK_MAIN | 0200h | クロックソース: メインクロック |
| REG_HOCOWTCR2 (注 1) | 03h (50MHz 選択時) 02h (50MHz 以外選択時) | HOCO のウェイトコントロールレジスタの 設定値 |
| REG_SCKCR (注 2) | 2102 1222h (PLL 選択時) 1001 0111h (HOCO 選択時) 0000 0000h (上記以外) | 内部クロック分周比の設定(SCKCR レジスタ の設定値) |
| OPCM_HIGH | 00h | 動作電力制御モード: 高速動作モード |
| OPCM_MID_1A | 02h | 動作電力制御モード: 中速動作モード 1A |
| OPCM_MID_1B | 03h | 動作電力制御モード: 中速動作モード 1B |
| OPCM_MID_2A | 04h | 動作電力制御モード: 中速動作モード 2A |
| OPCM_MID_2B | 05h | 動作電力制御モード: 中速動作モード 2B |
| OPCM_LOW_1 | 06h | 動作電力制御モード: 低速動作モード 1 |
| OPCM_LOW_2 | 07h | 動作電力制御モード: 低速動作モード 2 |
| MAIN_CLOCK_CYCLE | (1,000,000,000L / MAIN_CLOCK_Hz) | メインクロックの周期(ns) |
| SUB_CLOCK_CYCLE | (1,000,000,000L / SUB_CLOCK_Hz) | サブクロックの周期(ns) |
| FOR_CMT0_TIME | 232727 | 発振安定待機時間待ち用タイマ(CMT0)の カウント周期(ns)= 1/LOCO(137.5kHz) × 32 (LOCO=137.5kHz(max)、PCLKB の 32 分周) |
| MODULE_STOP_ENABLE | 1 | モジュールストップ状態へ遷移 |
| MODULE_STOP_DISABLE | 0 | モジュールストップ状態の解除 |

注1. 選択した HOCO 周波数により、設定値が異なります。

注2. 選択したシステムクロックのクロックソースにより、設定値が異なります。

表3.13 100ピン版(PIN_SIZE=100)の場合での定数

| 定数名 | 設定値 | 内容 |
|-----------|------|-------------------|
| DEF_P0PDR | 0x00 | ポート P0 の方向レジスタ設定値 |
| DEF_P1PDR | 0x00 | ポート P1 の方向レジスタ設定値 |
| DEF_P2PDR | 0x00 | ポート P2 の方向レジスタ設定値 |
| DEF_P3PDR | 0x00 | ポート P3 の方向レジスタ設定値 |
| DEF_P4PDR | 0x00 | ポート P4 の方向レジスタ設定値 |
| DEF_P5PDR | 0x00 | ポート P5 の方向レジスタ設定値 |
| DEF_PAPDR | 0x00 | ポート PA の方向レジスタ設定値 |
| DEF_PBPDR | 0x00 | ポート PB の方向レジスタ設定値 |
| DEF_PCPDR | 0x00 | ポート PC の方向レジスタ設定値 |
| DEF_PEPDR | 0x00 | ポート PE の方向レジスタ設定値 |
| DEF_PHPDR | 0x00 | ポート PH の方向レジスタ設定値 |
| DEF_PJPDR | 0x00 | ポート PJ の方向レジスタ設定値 |

表3.14 80ピン版(PIN_SIZE=80)の場合での定数

| 定数名 | 設定値 | 内容 |
|-----------|------|-------------------|
| DEF_P0PDR | 0x00 | ポート P0 の方向レジスタ設定値 |
| DEF_P1PDR | 0x00 | ポート P1 の方向レジスタ設定値 |
| DEF_P2PDR | 0x3C | ポート P2 の方向レジスタ設定値 |
| DEF_P3PDR | 0x08 | ポート P3 の方向レジスタ設定値 |
| DEF_P4PDR | 0x00 | ポート P4 の方向レジスタ設定値 |
| DEF_P5PDR | 0x0F | ポート P5 の方向レジスタ設定値 |
| DEF_PAPDR | 0x80 | ポート PA の方向レジスタ設定値 |
| DEF_PBPDR | 0x00 | ポート PB の方向レジスタ設定値 |
| DEF_PCPDR | 0x03 | ポート PC の方向レジスタ設定値 |
| DEF_PEPDR | 0xC0 | ポート PE の方向レジスタ設定値 |
| DEF_PHPDR | 0x00 | ポート PH の方向レジスタ設定値 |
| DEF_PJPDR | 0x08 | ポート PJ の方向レジスタ設定値 |

表3.15 64ピン版(PIN_SIZE=64)の場合での定数

| 定数名 | 設定値 | 内容 |
|-----------|------|-------------------|
| DEF_P0PDR | 0x80 | ポート P0 の方向レジスタ設定値 |
| DEF_P1PDR | 0x0C | ポート P1 の方向レジスタ設定値 |
| DEF_P2PDR | 0x3F | ポート P2 の方向レジスタ設定値 |
| DEF_P3PDR | 0x18 | ポート P3 の方向レジスタ設定値 |
| DEF_P4PDR | 0x0C | ポート P4 の方向レジスタ設定値 |
| DEF_P5PDR | 0x0F | ポート P5 の方向レジスタ設定値 |
| DEF_PAPDR | 0xA4 | ポート PA の方向レジスタ設定値 |
| DEF_PBPDR | 0x14 | ポート PB の方向レジスタ設定値 |
| DEF_PCPDR | 0x03 | ポート PC の方向レジスタ設定値 |
| DEF_PEPDR | 0xC0 | ポート PE の方向レジスタ設定値 |
| DEF_PHPDR | 0x00 | ポート PH の方向レジスタ設定値 |
| DEF_PJPDR | 0x0A | ポート PJ の方向レジスタ設定値 |

3.7 関数一覧

表 3.16にサンプルコードで使用する関数を示します。

表3.16 サンプルコードで使用する関数

| 関数名 | 概要 |
|-------------------------|------------------------------|
| main | メイン処理 |
| R_INIT_StopModule | リセット後に動作している周辺機能の停止 |
| R_INIT_NonExistentPort | 存在しないポートの初期設定 |
| R_INIT_Clock | クロック初期設定 |
| CGC_oscillation_main | メインクロックの発振設定 |
| CGC_oscillation_PLL | PLL クロックの発振設定 |
| CGC_oscillation_HOCO | HOCO クロックの発振設定 |
| CGC_oscillation_sub | サブクロックの発振設定 |
| CGC_disable_subclk | サブクロックの停止設定 |
| oscillation_subclk | サブクロックの発振 |
| no_use_subclk_as_sysclk | サブクロックをシステムクロックとして使用しない場合の設定 |
| resetting_wtcr_subclk | サブクロックのウェイトコントロールレジスタの再設定 |
| enable_RTC | RTC を使用する場合の初期化 |
| cmt0_wait | CMT0 によるソフトウェアウェイト |

3.8 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 | リセット後に動作している周辺機能の停止設定関数、存在しないポートの初期設定関数およびクロックの初期設定関数を呼び出します。 |
| 引数 | なし |
| リターン値 | なし |

| R_INIT_StopModule | |
|-------------------|------------------------------------|
| 概要 | リセット後に動作している周辺機能の停止 |
| ヘッダ | r_init_stop_module.h |
| 宣言 | void R_INIT_StopModule (void) |
| 説明 | モジュールストップ状態へ遷移する設定を行います。 |
| 引数 | なし |
| リターン値 | なし |
| 備考 | サンプルコードでは、モジュールストップ状態への遷移は行っていません。 |

| R_INIT_NonExistentPort | |
|------------------------|--|
| 概要 | 存在しないポートの初期設定 |
| ヘッダ | r_init_non_existent_port.h |
| 宣言 | void R_INIT_NonExistentPort (void) |
| 説明 | 100ピン未満の製品に対して、存在しないポートの端子に対応するポート方向レジスタの初期設定を行います。 |
| 引数 | なし |
| リターン値 | なし |
| 備考 | サンプルコードでは、100ピン版(PIN_SIZE=100)に設定しています。 本関数をコールした後に、存在しないポートを含む PDR、PODR レジスタへバイト単位で書き込む場合、存在しないポートの方向制御ビットには“1”、ポート出力データ格納ビットには“0”を設定してください。 |

| R_INIT_Clock | |
|--------------|--|
| 概要 | クロック初期設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void R_INIT_Clock (void) |
| 説明 | クロックの初期設定を行います。 |
| 引数 | なし |
| リターン値 | なし |
| 備考 | サンプルコードでは、システムクロックを PLL とし、サブクロックを使用しない処理を選択しています。 |

CGC_oscillation_main

| | |
|-------|---|
| 概要 | メインクロックの発振設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void CGC_oscillation_main (void) |
| 説明 | メインクロックのドライブ能力、および MOSCWTCR レジスタを設定してから、メインクロックを発振します。その後、ソフトウェアによるメインクロックの発振安定待機時間待ちを行います。 |
| 引数 | なし |
| リターン値 | なし |

CGC_oscillation_PLL

| | |
|-------|--|
| 概要 | PLL クロックの発振設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void CGC_oscillation_PLL (void) |
| 説明 | PLL 入力分周比および周波数通倍率の設定、および PLLWTCR を設定してから、PLL クロックを発振します。その後、ソフトウェアによる PLL の発振安定時間待ちを行います。 |
| 引数 | なし |
| リターン値 | なし |

CGC_oscillation_HOCO

| | |
|-------|--|
| 概要 | HOCO クロックの発振設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void CGC_oscillation_HOCO (void) |
| 説明 | HOCO の周波数、および HOCOWTCR2 レジスタを設定してから、HOCO を発振します。その後、ソフトウェアによる HOCO の発振安定待機時間待ちを行います。 |
| 引数 | なし |
| リターン値 | なし |

CGC_oscillation_sub

| | |
|-------|---|
| 概要 | サブクロックの発振設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void CGC_oscillation_sub (void) |
| 説明 | サブクロックをシステムクロック、RTC のカウントソースのどちらか、または両方に使用する場合の設定を行います。 |
| 引数 | なし |
| リターン値 | なし |

CGC_disable_subclk

| | |
|-------|---|
| 概要 | サブクロックの停止設定 |
| ヘッダ | r_init_clock.h |
| 宣言 | void CGC_disable_subclk (void) |
| 説明 | サブクロックをシステムクロックにも、RTCのカウントソースにも使用しない場合の設定を行います。 |
| 引数 | なし |
| リターン値 | なし |

oscillation_subclk

| | |
|-------|---------------------------------------|
| 概要 | サブクロックの発振 |
| ヘッダ | なし |
| 宣言 | static void oscillation_subclk (void) |
| 説明 | サブクロックの発振設定を行います。 |
| 引数 | なし |
| リターン値 | なし |

no_use_subclk_as_sysclk

| | |
|-------|---|
| 概要 | サブクロックをシステムクロックとして使用しない場合の設定 |
| ヘッダ | なし |
| 宣言 | static void no_use_subclk_as_sysclk (void) |
| 説明 | サブクロックをRTCのカウントソースのみに使用する場合、システムクロックとしてのサブクロックを停止します。 |
| 引数 | なし |
| リターン値 | なし |

resetting_wtcr_subclk

| | |
|-------|---|
| 概要 | サブクロックのウェイトコントロールレジスタの再設定 |
| ヘッダ | なし |
| 宣言 | static void resetting_wtcr_subclk (void) |
| 説明 | ソフトウェアスタンバイモードからの復帰の際のウェイトコントロールレジスタの再設定を行います。この際、ウェイトコントロールレジスタは最小値に設定します。 |
| 引数 | なし |
| リターン値 | なし |

| | |
|------------|--|
| enable_RTC | |
| 概要 | RTC を使用する場合の初期化 |
| ヘッダ | なし |
| 宣言 | static void enable_RTC (void) |
| 説明 | RTC を使用する場合の初期化(クロック供給の設定と RTC ソフトウェアリセット)を行います。 |
| 引数 | なし |
| リターン値 | なし |

| | |
|-----------|--|
| cmt0_wait | |
| 概要 | CMT0 によるソフトウェアウェイト |
| ヘッダ | なし |
| 宣言 | static void cmt0_wait (uint32_t cnt) |
| 説明 | 発振安定待機時間待ちを行う際に使用します。 |
| 引数 | uint32_t cnt: 発振安定待機時間 cnt = 発振安定待機時間(ns)(注 1) ÷ FOR_CMT0_TIME (注 2) |
| リターン値 | なし |
| 備考 | 注 1. 発振安定待機時間は発振子によって異なります。3.3.2の算出方法に基づいて設定してください。 注 2. FOR_CMT0_TIME は LOCO = 137.5kHz(max)で算出します。 実際の待ち時間は LOCO の周波数により異なります。 |

3.9 フローチャート

3.9.1 メイン処理

図 3.5にメイン処理のフローチャートを示します。

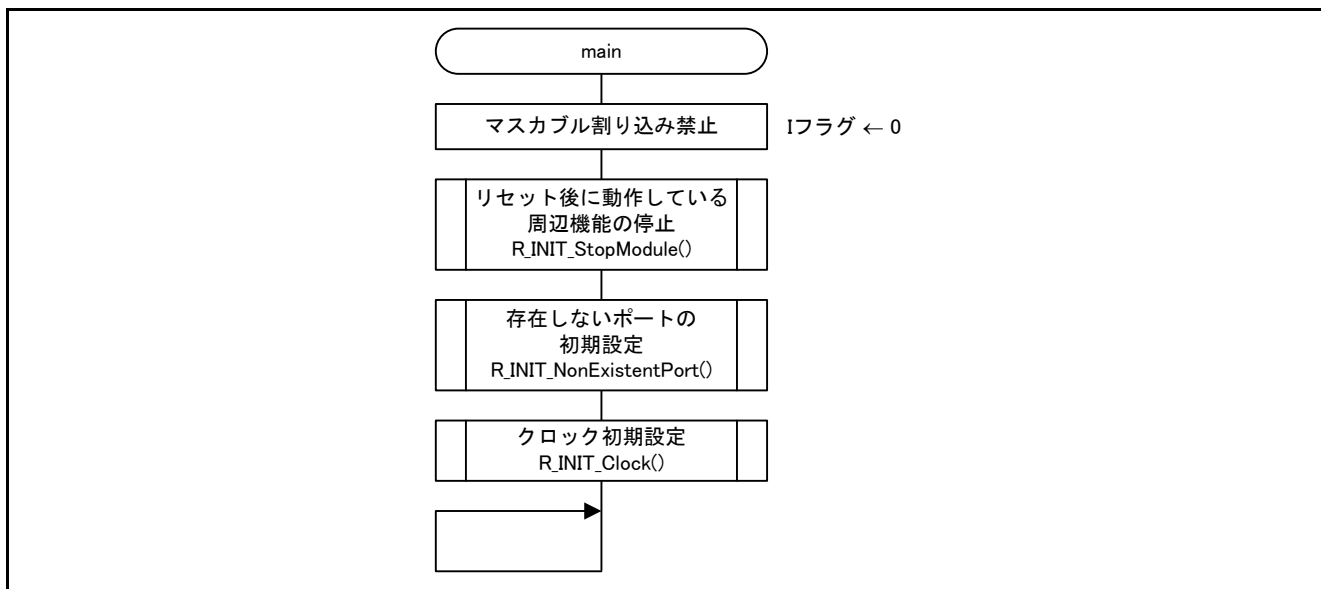


図3.5 メイン処理

3.9.2 リセット後に動作している周辺機能の停止

図 3.6にリセット後に動作している周辺機能の停止のフローチャートを示します。

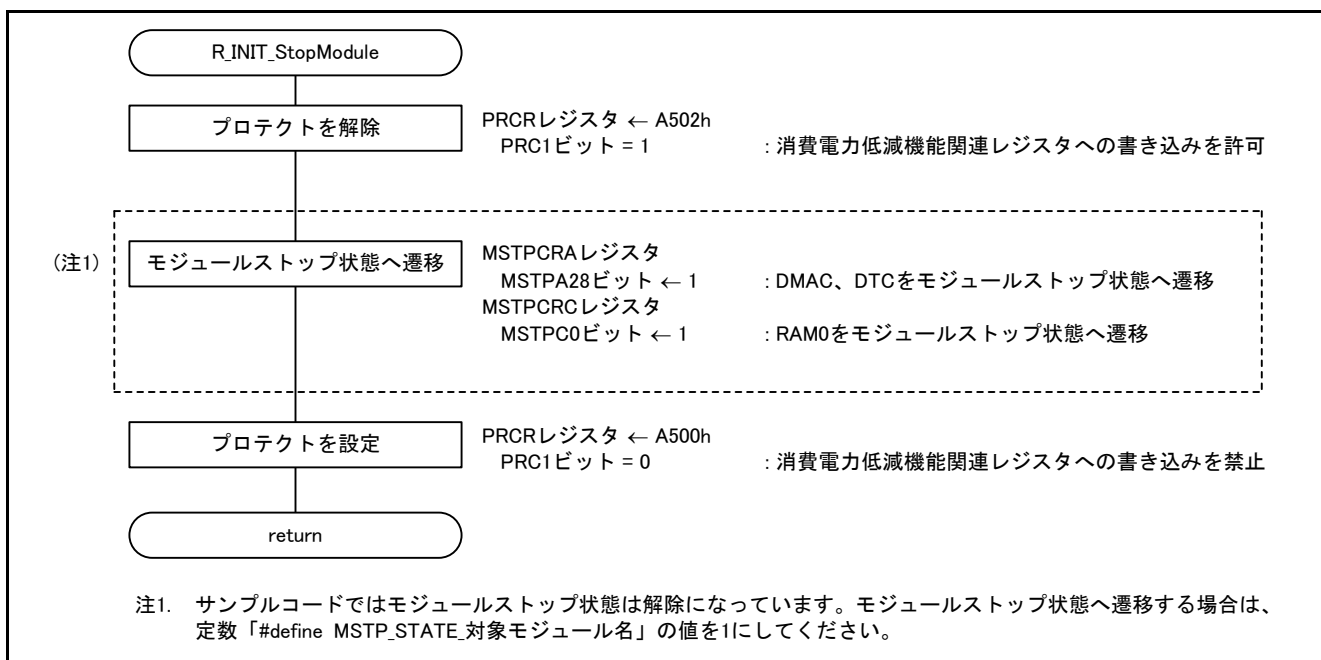


図3.6 リセット後に動作している周辺機能の停止

3.9.3 存在しないポートの初期設定

図 3.7に存在しないポートの初期設定のフローチャートを示します。

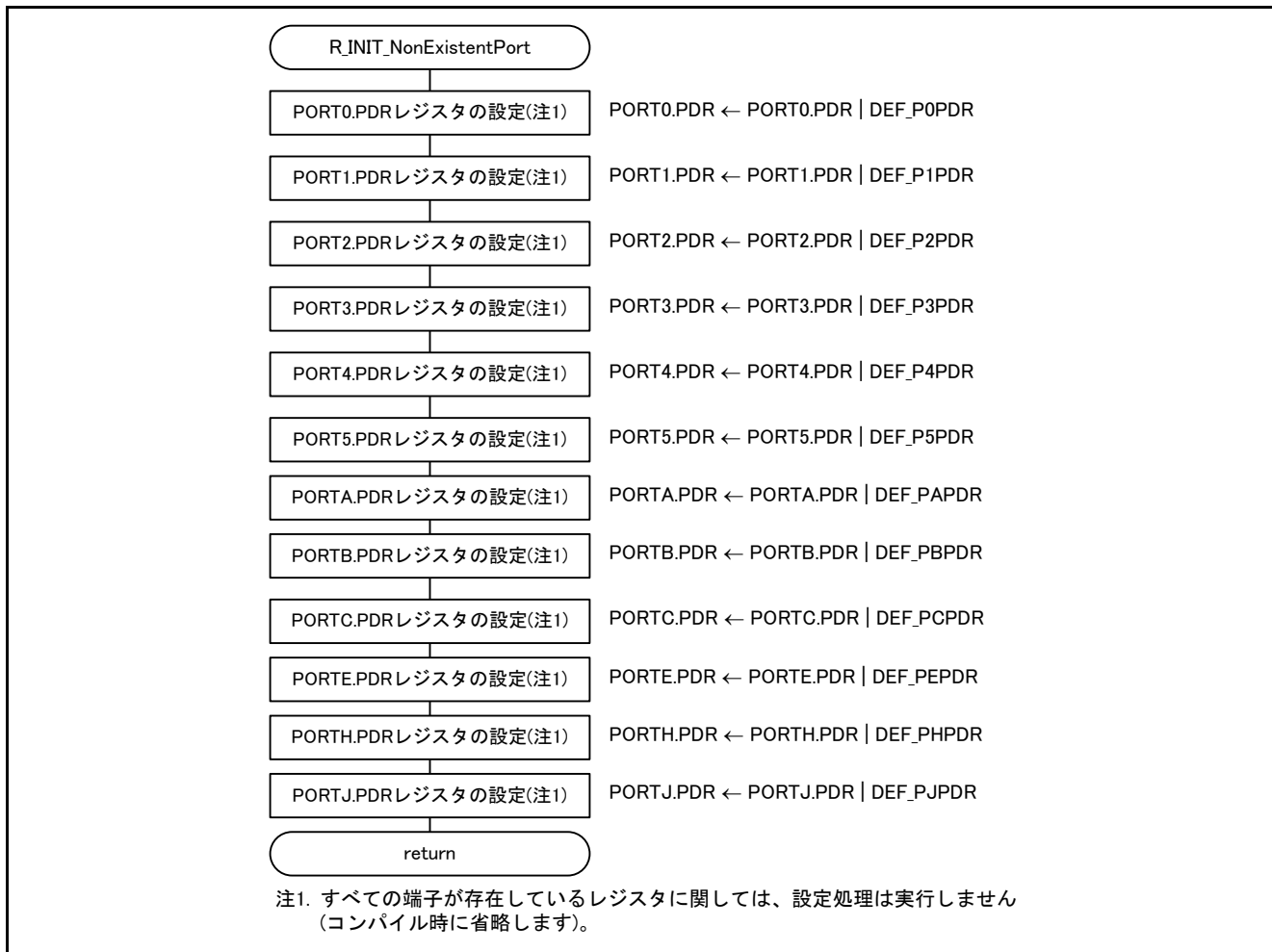


図3.7 存在しないポートの初期設定

3.9.4 クロック初期設定

図 3.8、図 3.9にクロック初期設定のフローチャートを示します。

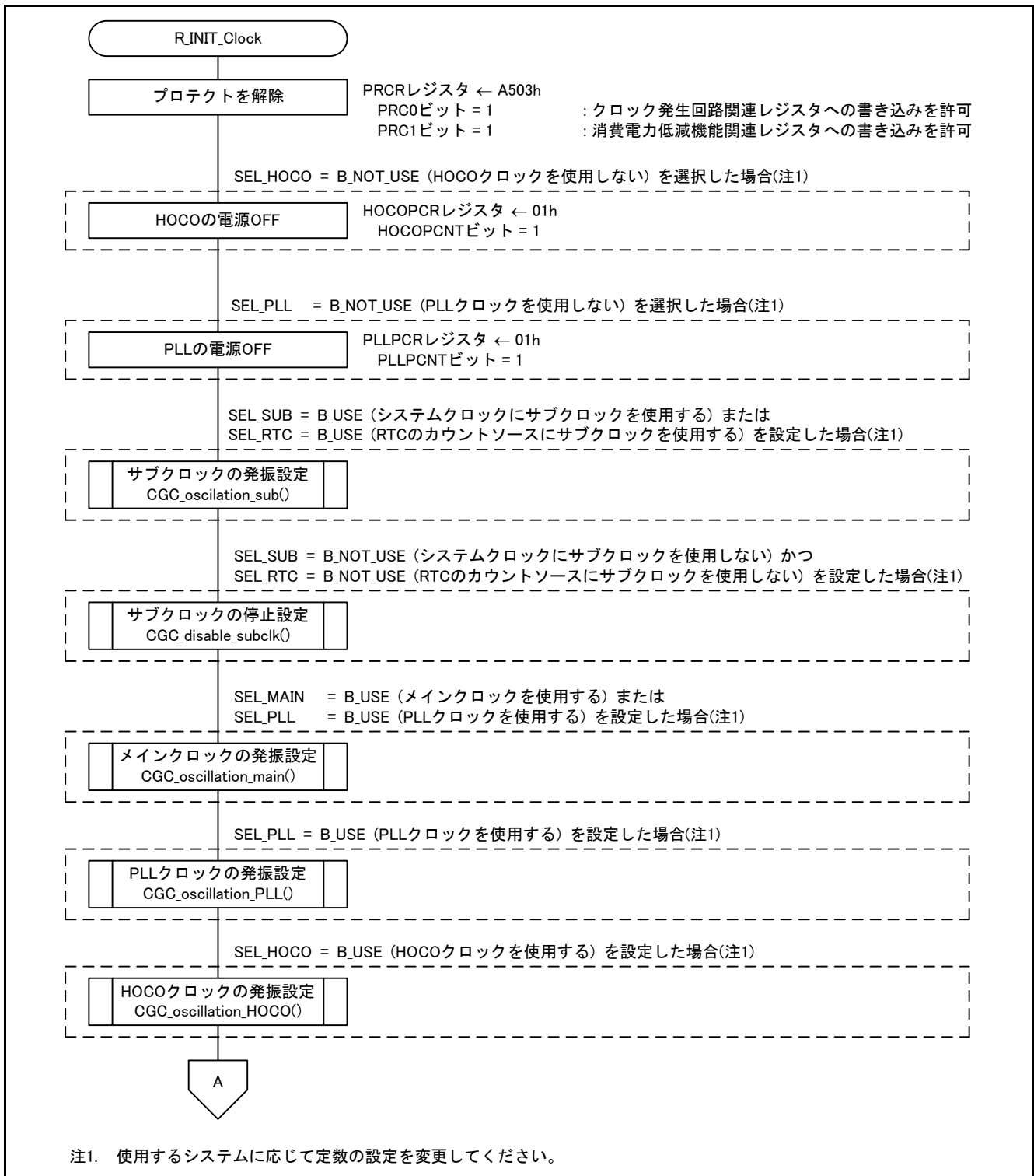


図3.8 クロック初期設定(1/2)

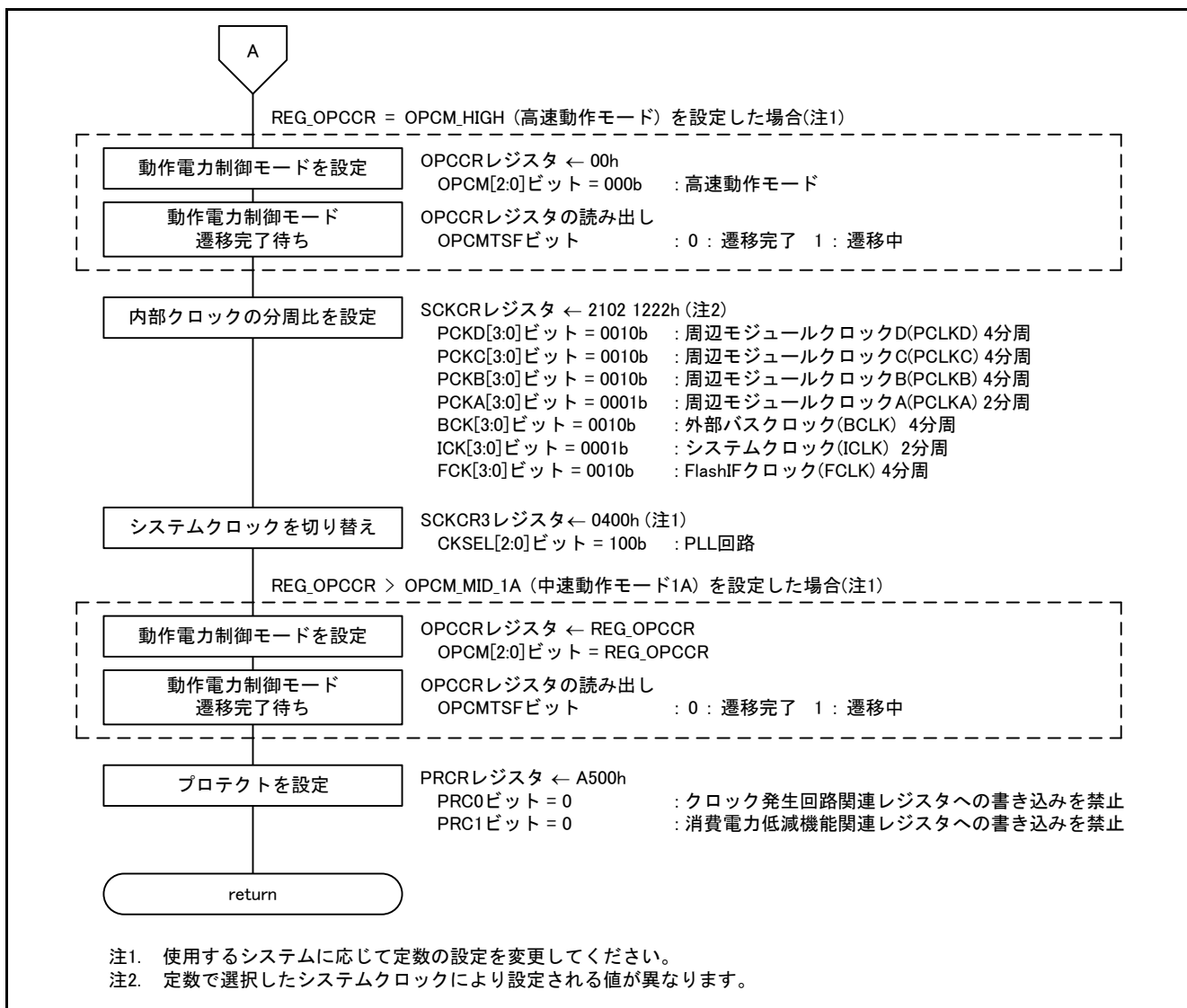


図3.9 クロック初期設定(2/2)

3.9.5 メインクロックの発振設定

図 3.10にメインクロックの発振設定のフローチャートを示します。

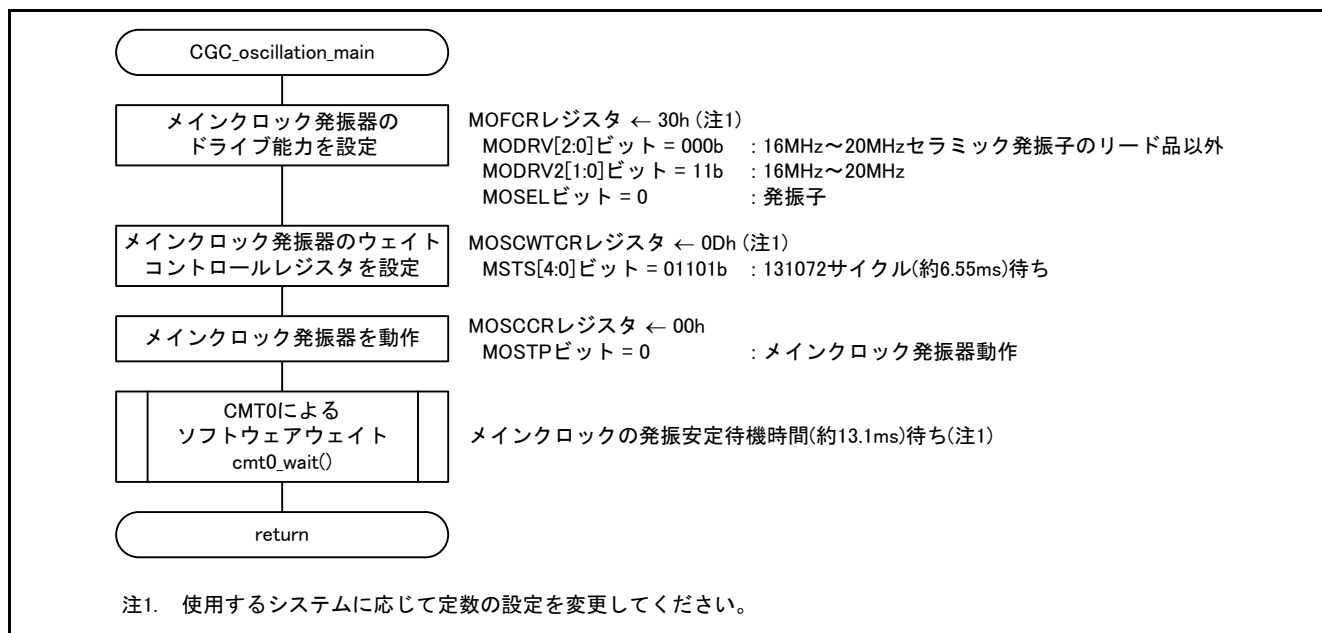


図3.10 メインクロックの発振設定

3.9.6 PLL クロックの発振設定

図 3.11にPLL クロックの発振設定のフローチャートを示します。

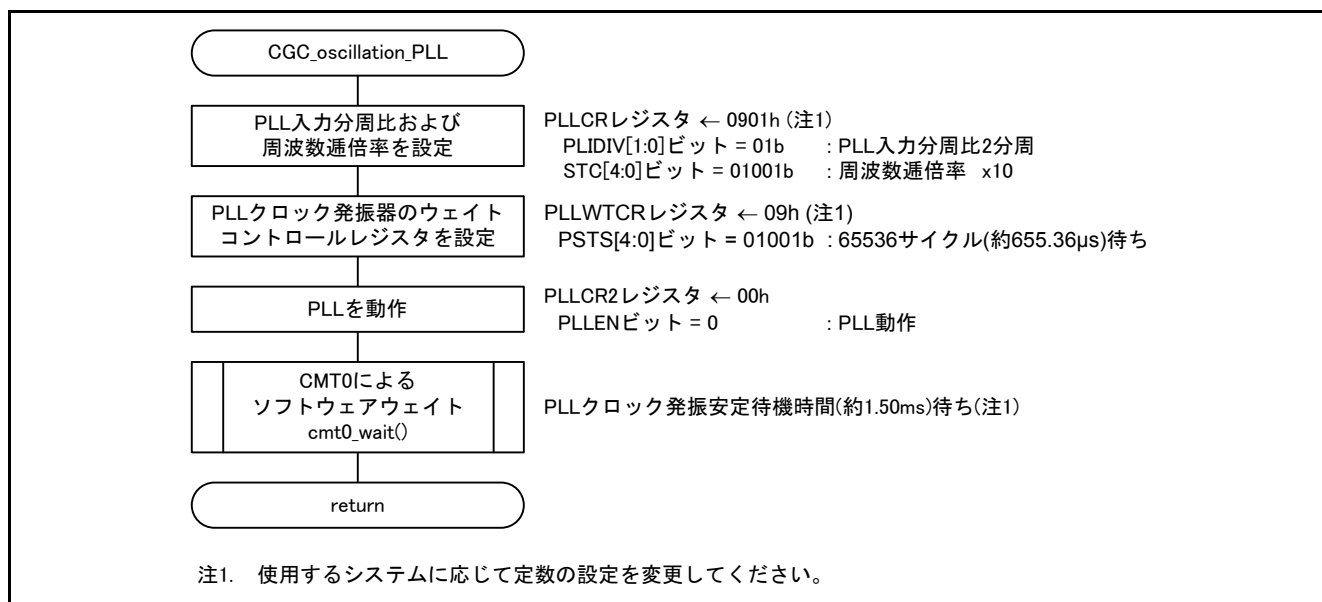


図3.11 PLL クロックの発振設定

3.9.7 HOCO クロックの発振設定

図 3.12にHOCO クロックの発振設定のフローチャートを示します。

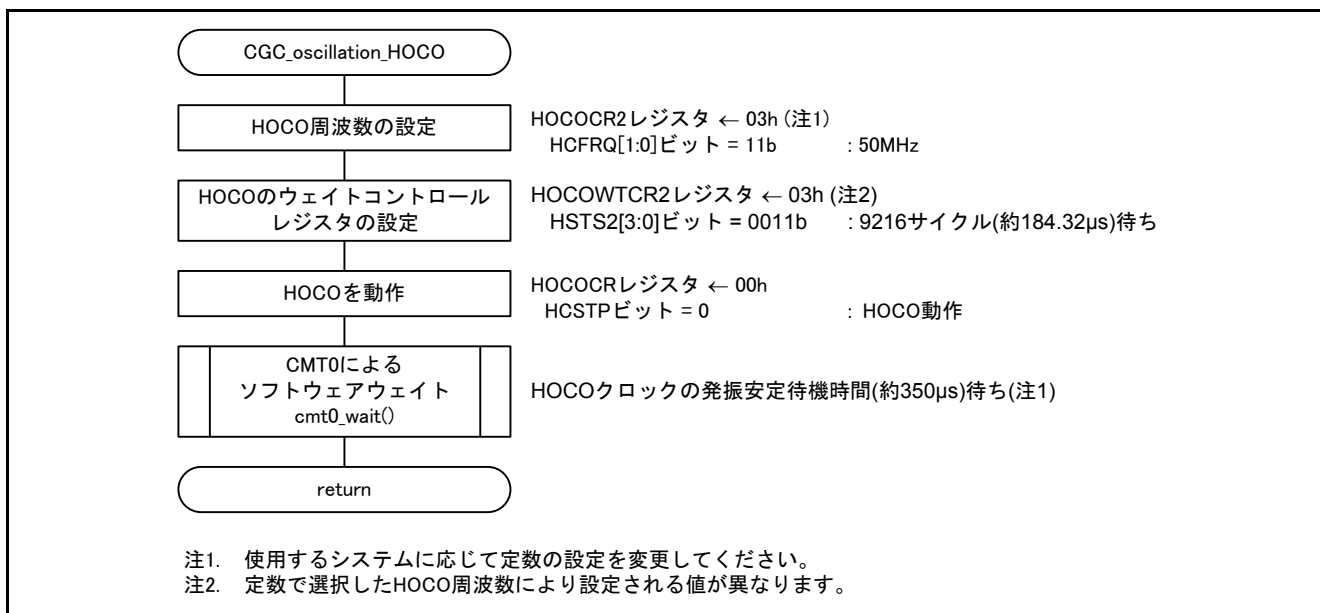


図3.12 HOCO クロックの発振設定

3.9.8 サブクロックの発振設定

図 3.13、図 3.14にサブクロックの発振設定を示します。

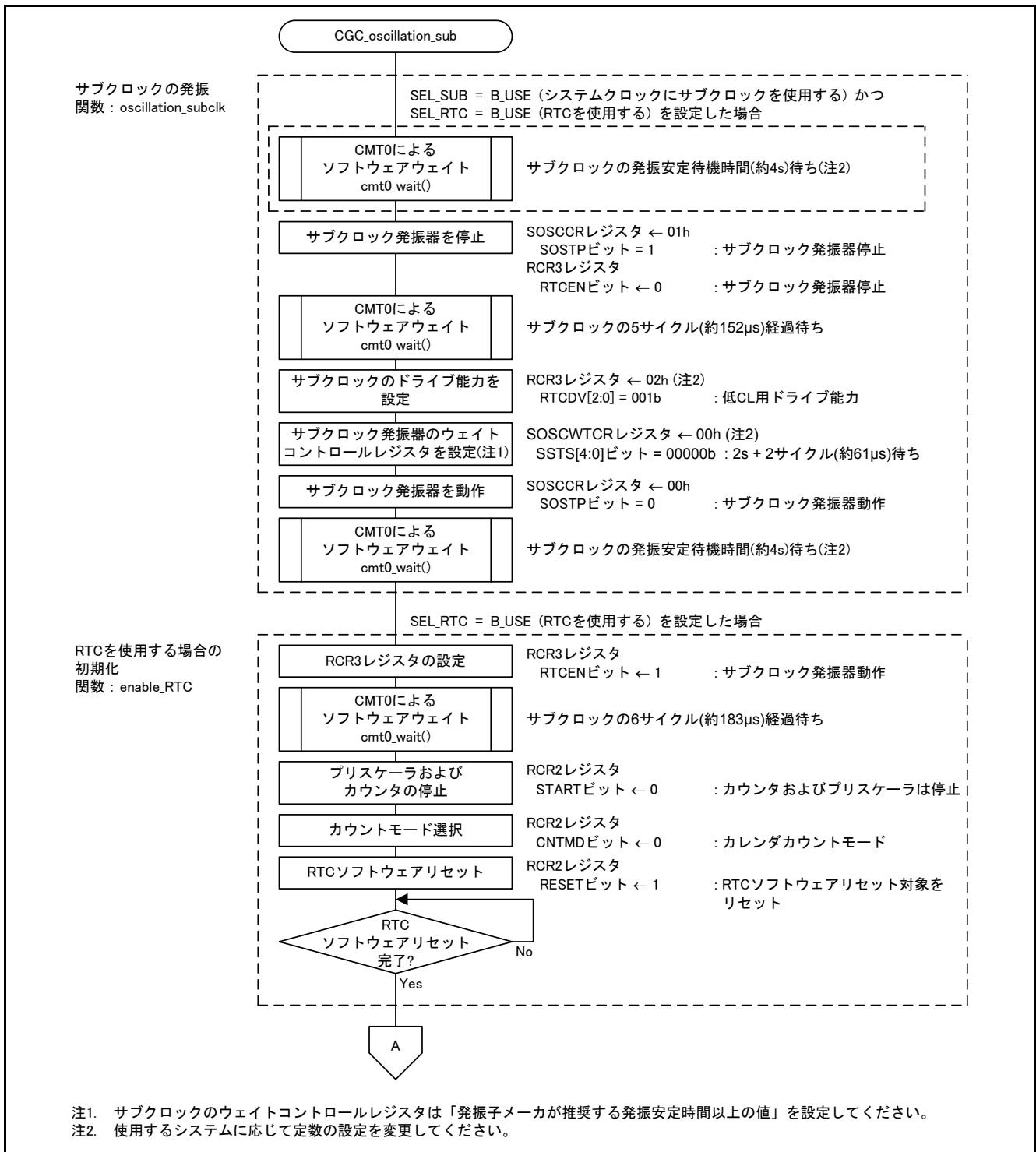


図3.13 サブクロックの発振設定(1/2)

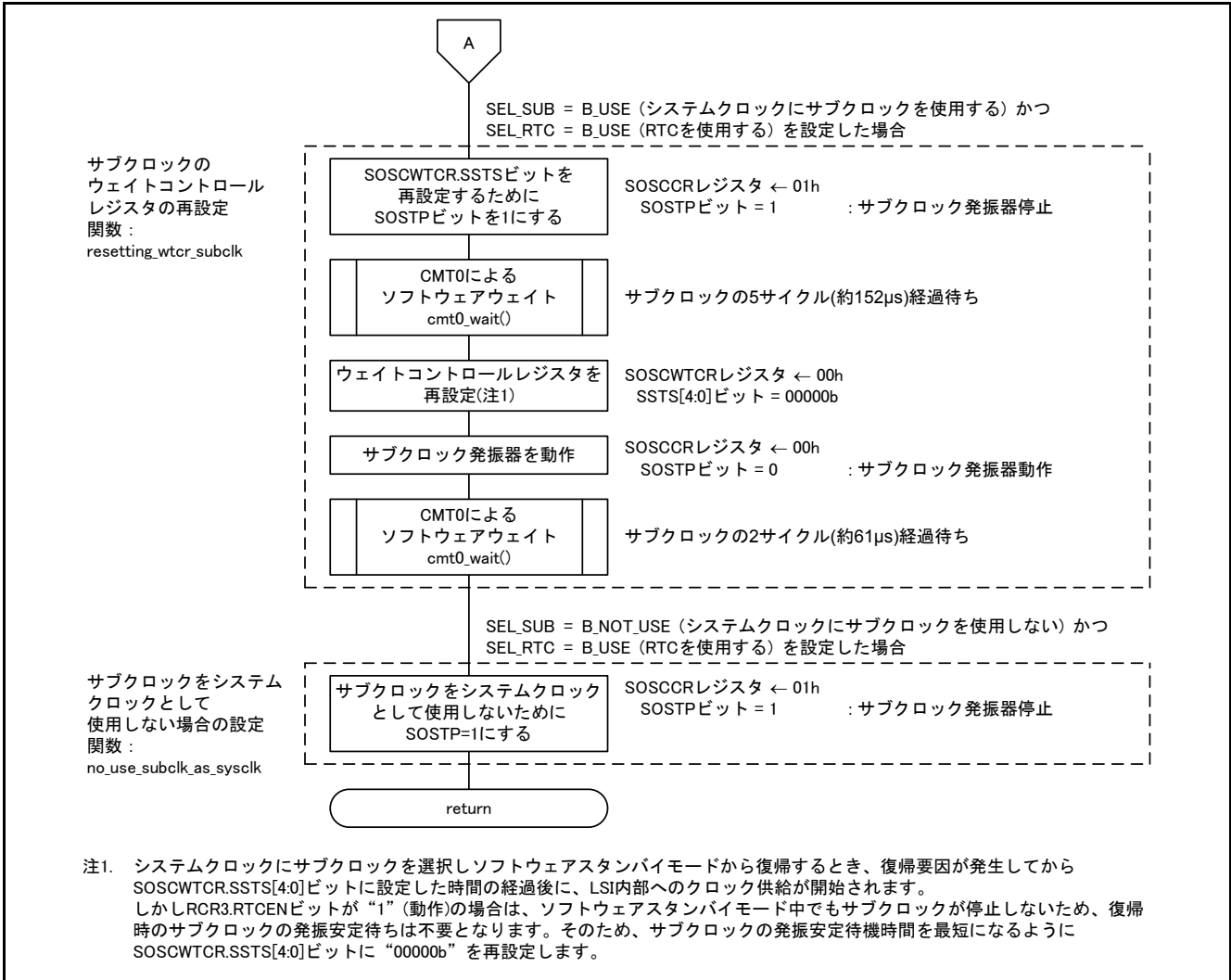


図3.14 サブクロックの発振設定(2/2)

3.9.9 サブクロックの停止設定

図 3.15にサブクロックの停止設定のフローチャートを示します。

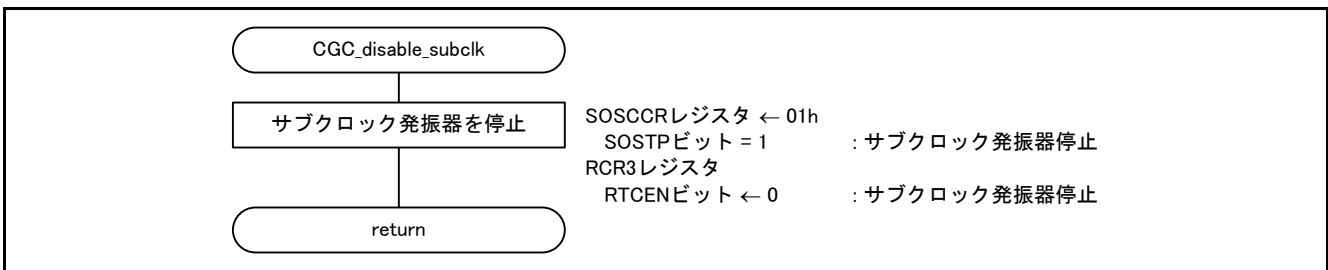


図3.15 サブクロックの停止設定

3.9.10 CMT0 によるソフトウェアウェイト

図 3.16にCMT0 によるソフトウェアウェイトのフローチャートを示します。

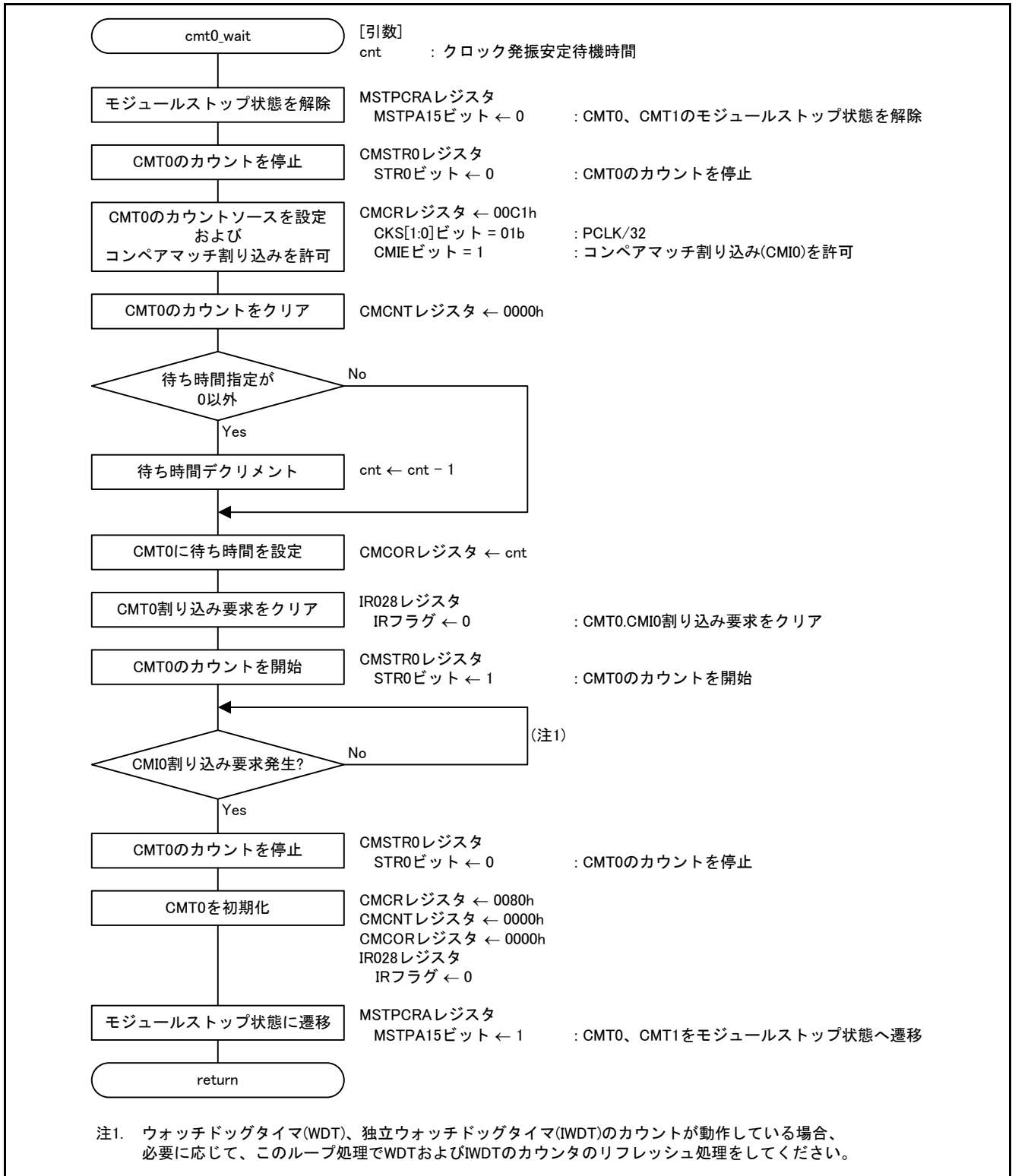


図3.16 CMT0 によるソフトウェアウェイト

4. 付録

4.1 クロックの発振安定待機時間の考え方

4.1.1 メインクロックの発振安定を待たずに PLL の発振を行う場合

メインクロックと PLL クロックの発振を行う場合は、メインクロックと PLL クロックの発振安定待機時間をまとめて待つことが可能です。

図 4.1にPLL の発振安定待機時間の考え方(メインクロックの発振安定を待たずに PLL の発振を行う場合)、表 4.1にPLL のウェイトコントロールと発振安定待機時間の算出方法と設定値(メインクロックの発振安定を待たずに PLL の発振を行う場合)を示します。

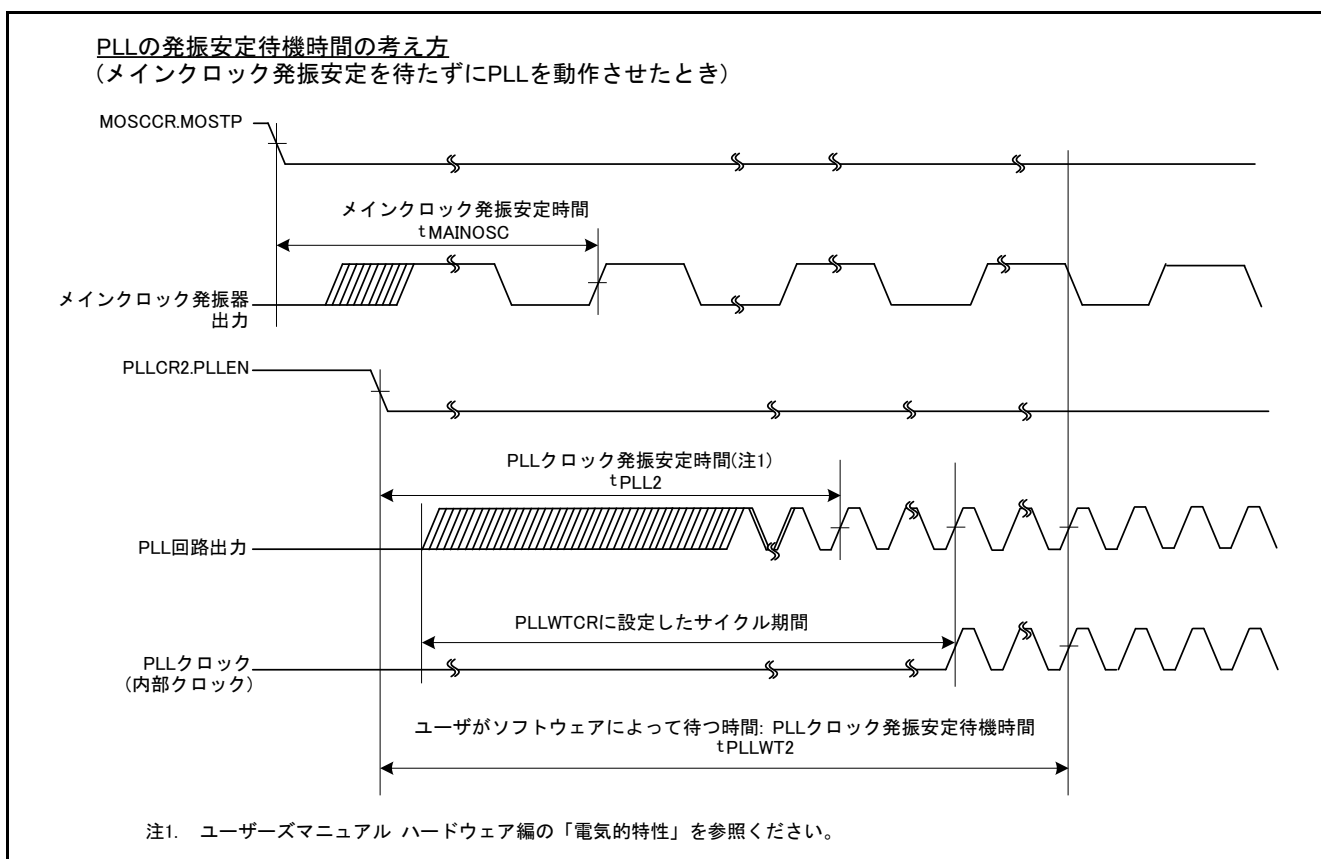


図4.1 PLL の発振安定待機時間の考え方(メインクロックの発振安定を待たずに PLL の発振を行う場合)

表4.1 PLL のウェイトコントロールと発振安定待機時間の算出方法と設定値(メインクロックの発振安定を待たずに PLL の発振を行う場合)

| 発振安定待機時間 | 算出方法 |
|-------------------------------|--|
| ウェイトコントロールレジスタ (PLLWTCR.PSTS) | 発振子メーカーが推奨するメインクロックの発振安定時間以上の値 + t_{PLL1} (最大 500 μ s)以上の値 |
| 発振安定待機時間 (t_{PLLWT2}) | PLLWTCR レジスタで設定したサイクル期間の 2 倍以上の値 |

5. RX200 シリーズのアプリケーションノートをRX21A グループに適用する方法

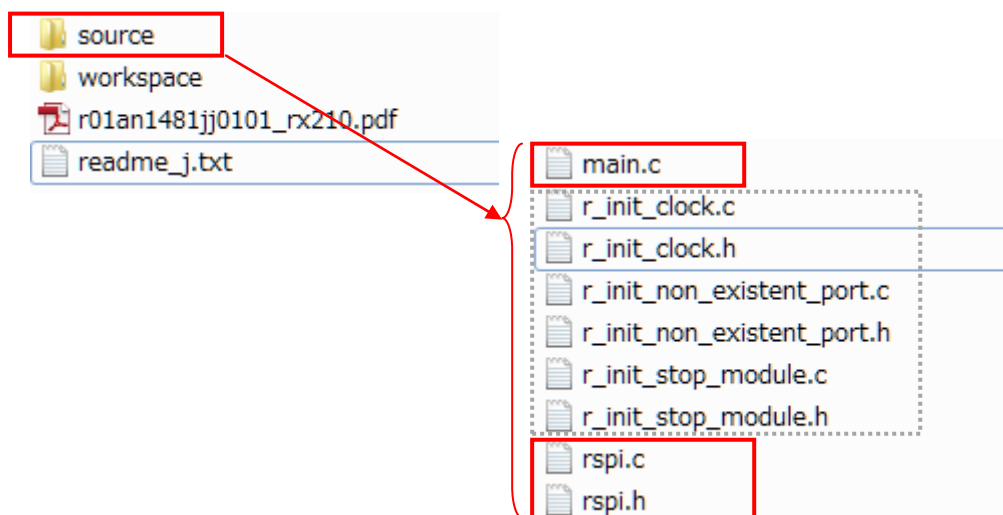
同じ周辺機能がRX21Aグループにあり、その周辺機能を使ったRX200シリーズアプリケーションノートは、スタートアッププログラムを「RX21Aグループ 初期設定例」に差し替えることで、RX21Aグループに適用することができます。以下に手順例を示します。

ここでは、RX200シリーズアプリケーションノートに「RX210、RX21A、RX220グループ アプリケーションノート RSPI を用いた通信設定例(R01AN1481JJ0101)」を使用します。

- (1) 初期設定例 アプリケーションノートのプロジェクトを開き、RX200シリーズアプリケーションノートのプロジェクトから以下のファイルをコピーします。

[source フォルダ]

- ・ r_init_clock、r_init_non_existent_port、r_init_stop_module **以外**のソースファイル、ヘッダファイル
(赤枠で囲ったファイル/main.c や rspi.c、rspi.h など使用する機能のソースファイル)



- (2) RX200 シリーズ アプリケーションノートの「関数一覧」項に”Excep_”から始まる関数名がないかを確認してください。”Excep_”から始まる関数名がある場合、rx21a_clock_port_r01an1486 フォルダにある intprg.c を開き、同じ名称の割り込み関数の定義を削除(あるいはコメントアウト)します。

表5.7 サンプルコードで使用する関数

| 関数名 | 概要 |
|------------------------|------------------------------|
| main | メイン処理 |
| port_init | ポート初期設定 |
| R_INIT_StopModule | リセット後に動作している周辺機能の停止 |
| R_INIT_NonExistentPort | 存在しないポートの初期設定 |
| R_INIT_Clock | クロック初期設定 |
| peripheral_init | 周辺機能初期設定 |
| cb_rspi_slave0_end | コールバック関数(スレーブ 0 へのRSPI送受信完了) |
| cb_rspi_slave1_end | コールバック関数(スレーブ 1 へのRSPI送受信完了) |
| cb_rspi_rx_error | コールバック関数(RSPI受信エラー) |
| RSPI_Init | ユーザI/F関数(RSPI初期設定) |
| RSPI_PreTrans | ユーザI/F関数(RSPI送受信開始) |
| RSPI_GetState | ユーザI/F関数(RSPI状態取得) |
| rspi_spti_isr | RSPI送信割り込み |
| rspi_spii_isr | RSPIアイドル割り込み |
| rspi_spri_isr | RSPI受信割り込み |
| rspi_spei_isr | RSPIエラー割り込み |
| Excep_RSPIO_SPEIO | RSPIO.SPEIO 割り込み処理 |
| Excep_RSPIO_SPRIO | RSPIO.SPRIO 割り込み処理 |
| Excep_RSPIO_SPTIO | RSPIO.SPTIO 割り込み処理 |
| Excep_RSPIO_SPIIO | RSPIO.SPIIO 割り込み処理 |

```

行番  S... ソース
82    void Excep_CAC_OVFF(void){ }
83
84    // RSPIO SPEIO
85    //void Excep_RSPIO_SPEIO(void){ }
86
87    // RSPIO SPRIO
88    //void Excep_RSPIO_SPRIO(void){ }
89
90    // RSPIO SPTIO
91    //void Excep_RSPIO_SPTIO(void){ }
92
93    // RSPIO SPIIO
94    //void Excep_RSPIO_SPIIO(void){ }
95
96    // DOC DOPCF
97    void Excep_DOC_DOPCF(void){ }
98
intprg.c*

```

(3) 使用するクロックに応じて、`r_init_clock_h` の設定を変更します。変更する内容は、本アプリケーションノートを参考にしてください。

(4) 適用するアプリケーションノートでの設定を見直します。

- ・使用する端子に同じ機能が割り付けられているか。
- ・ICLK や PCLK の変化によって、周辺機能の設定を変更する必要があるか。

PCLK の周波数が変わるため、通信ビットレートなどの設定値を、変更後の周波数に合わせて見直してください。

注. シリアルコミュニケーションインターフェースを使用するアプリケーションノートを適用する場合、チャンネル 1 がオンチップデバッグエミュレータと接続されている可能性があります。

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

RX21Aグループ ユーザーズマニュアル ハードウェア編 Rev.1.00(R01UH0251JJ)
(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(R20UT0570JJ)
(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 | RX21A グループ アプリケーションノート 初期設定例 |
|------|------------------------------|

| Rev. | 発行日 | 改訂内容 | |
|------|------------|-------|---|
| | | ページ | ポイント |
| 1.00 | 2013.10.01 | — | 初版発行 |
| 1.10 | 2014.07.01 | 34-36 | RX200 シリーズのアプリケーションノートを RX21A グループに適用する方法の説明を追加 |

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うと、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、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事情報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/>