

## RX ファミリ

### 低消費電力モードへの移行例

---

#### 要旨

本アプリケーションノートでは、以下に示す対象デバイスの消費電力低減機能を使用し、各低消費電力モードへの移行設定例について説明します。

また、本アプリケーションノートを適用することにより、低消費電力モードに移行した際の電流値の測定を容易に実施することが出来ますのでご活用ください。

#### 対象デバイス

- ・RX ファミリ

#### 動作確認デバイス

- ・RX130 グループ
- ・RX140 グループ
- ・RX231 グループ
- ・RX65N 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応じて変更し、十分評価してください。

#### 関連ドキュメント

RX210、RX21A、RX220 グループ 消費電力低減機能を使用した各低消費電力モードへの移行例 (R01AN1482)

RX63N グループ、RX631 グループ 消費電力低減機能を使用した各低消費電力モードへの移行例 (R01AN1920)

RX140 グループ スヌーズモードの使用例(R01AN5914)

## 目次

1. 概要	3
2. 動作確認条件	5
3. ハードウェア説明	6
3.1 使用端子一覧	6
3.2 ハードウェア構成	7
3.2.1 ハードウェアの改造	7
4. ソフトウェア説明	8
4.1 動作概要	9
4.1.1 スリープモード	9
4.1.2 ディープスリープモード	9
4.1.3 全モジュールクロックストップモード	10
4.1.4 ソフトウェアスタンバイモード	10
4.1.5 ディープソフトウェアスタンバイモード	11
4.2 モジュール構成	12
4.3 プロジェクト構成	13
4.4 ファイル構成	16
4.5 オプション設定メモリ	17
4.6 定数一覧	17
4.7 変数一覧	18
4.8 関数一覧	18
4.9 関数仕様	19
4.10 フローチャート	21
4.10.1 メイン処理	21
4.10.2 ポートの初期設定	23
4.10.3 周辺機能の初期設定	24
4.10.4 低消費モード移行トリガ待ち	25
4.10.5 RTC アラーム割り込みの設定	26
4.10.6 10進数への変換処理	27
4.10.7 IRQ4 割り込み処理	27
5. スマート・コンフィグレータのコンフィグレーションについて	28
6. 低消費電力モードへ移行する際のノウハウ	28
7. 付録	29
7.1 プロジェクト毎の動作周波数	29
8. プロジェクトをインポートする方法	35
8.1 e <sup>2</sup> studio での手順	35
8.2 CS+での手順	36
9. サンプルコード	37
10. 参考ドキュメント	37
改訂記録	38

## 1. 概要

本アプリケーションノートでは、スマート・コンフィグレータ(SC)のコード生成(CG)機能によって生成される消費電力低減機能を使用して、低消費電力モードへの移行、および解除する方法を示します。

表 1.1 に各デバイスがサポートしているモードを示します。

表 1.1 各デバイスがサポートしているモード

モード	デバイス (○ : 対応、- : 非対応)			
	RX130	RX140	RX231	RX65N
スリープ	○	○	○	○
ディープスリープ	○	○	○	-
全モジュール クロックストップ	-	-	-	○
ソフトウェア スタンバイ	○	○	○	○
ディープ ソフトウェア スタンバイ	-	-	-	○
スヌーズ	-	○	-	-

本アプリケーションノートで扱う、低消費電力モードのサンプルプログラムの条件を表 1.2、表 1.3 に示します。また RX140 のスヌーズモード使用方法に関しては「RX140 グループ スヌーズモード使用例 (R01AN5914)」をご確認ください。

表 1.2 RX130、RX140、RX231 のサンプルプログラムの条件

低消費電力モード	モジュールストップの条件	
スリープ	DMAC/DTC (注 1)、RAM0 を除きモジュールストップ状態	
ディープスリープ	DMAC/DTC (注 1)、RAM0 を除きモジュールストップ状態(注 2)	
ソフトウェア スタンバイ	RTC 未使用時	DMAC/DTC (注 1)、RAM0 を除きモジュールストップ状態
	RTC 使用時	DMAC/DTC (注 1)、RAM0 を除きモジュールストップ状態

注.1 RX130、RX140 には DMAC が搭載されていないため、DTC のみがモジュールストップの対象となります。

注.2 DMAC/DTC は消費電力低減機能においてモジュールストップ状態にしています。

表 1.3 RX65N のサンプルプログラムの条件(注 1)

低消費電力モード		モジュールストップの条件
スリープ		TSIP のみモジュールストップ状態
全モジュールクロックストップ		DMAC/DTC、EXDMAC、RAM、拡張 RAM、スタンバイ RAM を除きモジュールストップ状態(注 2)
ソフトウェアスタンバイ		DMAC/DTC、EXDMAC、RAM、拡張 RAM、スタンバイ RAM を除きモジュールストップ状態
ディープソフトウェアスタンバイ	RTC 未使用時	DMAC/DTC、EXDMAC、RAM、拡張 RAM、スタンバイ RAM を除きモジュールストップ状態(注 3)
	RTC 使用時	DMAC/DTC、EXDMAC、RAM、拡張 RAM、スタンバイ RAM を除きモジュールストップ状態(注 3)

注.1 RX651 の消費電流を計測する場合は RX65N 用のサンプルプログラムを使用してください。

注.2 DMAC/DTC、EXDMAC は消費電力低減機能においてモジュールストップ状態にしています。

注.3 DMAC/DTC は消費電力低減機能においてモジュールストップ状態にしています。

本アプリケーションノートのサンプルプログラムでは、表 1.4 に示す幾つかの周辺機能を使用し、低消費電力モードの移行トリガが発生することにより、通常動作の状態から低消費電力モードへ移行し、低消費電力モードへ移行した状態で低消費電力モードの移行トリガが発生することにより、低消費電力モードの解除を行います。

表 1.4 使用する周辺機能と用途

周辺機能	用途
消費電力低減機能	消費電力の低減
外部端子割り込み(以下、IRQ)	低消費電力モードへの移行、低消費電力モードの解除
リアルタイムクロック(以下、RTC)	RTC 使用時の電流値測定(注 1)、アラーム割り込みによる低消費電力モードの解除(注 2)
I/O ポート	SW、LED 制御

注.1 RTC を使用したソフトウェアスタンバイモードの場合のみ使用します。

注.2 RTC を使用したディープソフトウェアスタンバイモードの場合のみ使用します。

## 2. 動作確認条件

本アプリケーションノートのサンプルプログラムは、表 2.1 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	R5F51308ADFP (RX130 グループ) R5F51406BDFN(RX140 グループ) R5F52318ADFP (RX231 グループ) R5F565NEDDFC(RX65N グループ)
動作周波数	動作周波数はプロジェクトごとに異なります。プロジェクト毎の動作周波数については 7.1 章を参照ください。
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 e <sup>2</sup> studio Version 2022-07
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.3.04.00 コンパイラオプション -lang = C99
iodefine.h のバージョン	Version 2.0 (RX130 グループ) Version 1.10C (RX140 グループ) Version 1.0I (RX231 グループ) Version 2.30 (RX65N グループ、RX651 グループ)
エミュレータ(注 1)	E2 Lite
エンディアン	リトルエンディアンとビッグエンディアン両方
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
サンプルプログラムのバージョン	Version 1.20
使用ボード(以下、RSK)	Renesas Starter Kit for RX130-512KB (製品型名: RTK5051308SxxxxxBE) Renesas Starter Kit for RX140 (製品型名: RTK551406BSxxxxxBE) Renesas Starter Kit for RX231 (製品型名: R0K505231SxxxBE) Renesas Starter Kit for RX65N-2MB (製品型名: RTK50565NSxxxxxBE)

注.1 電流計測時はエミュレータを外した状態で測定してください。

## 3. ハードウェア説明

## 3.1 使用端子一覧

表 3.1 ~ 表 3.4 に使用端子と機能を示します。

表 3.1 RX130-512KB の使用端子と機能

端子名	入出力	内容
P32/IRQ2	入力	SW2 入力(低消費電力モードへの移行および解除)
PD3	出力	LED0 出力 <ul style="list-style-type: none"> <li>・ 初期設定終了後、通常動作状態時に点灯</li> <li>・ 低消費電力モード移行時に消灯</li> <li>・ 低消費電力モード解除後に点灯</li> </ul>

表 3.2 RX140 の使用端子と機能

端子名	入出力	内容
P32/IRQ2	入力	SW2 入力(低消費電力モードへの移行および解除)
P21	出力	LED0 出力 <ul style="list-style-type: none"> <li>・ 初期設定終了後、通常動作状態時に点灯</li> <li>・ 低消費電力モード移行時に消灯</li> <li>・ 低消費電力モード解除後に点灯</li> </ul>

表 3.3 RX231 の使用端子と機能

端子名	入出力	内容
P34/IRQ4	入力	SW2 入力(低消費電力モードへの移行および解除)
P17	出力	LED0 出力 <ul style="list-style-type: none"> <li>・ 初期設定終了後、通常動作状態時に点灯</li> <li>・ 低消費電力モード移行時に消灯</li> <li>・ 低消費電力モード解除後に点灯</li> </ul>

表 3.4 RX65N-2MB の使用端子と機能

端子名	入出力	内容
P05/IRQ13	入力	SW2 入力(低消費電力モードへの移行および解除(注 1))
RES#	入力	低消費電力モードの解除(注 1)
P73	出力	LED0 出力 <ul style="list-style-type: none"> <li>・ 初期設定終了後、通常動作状態時に点灯</li> <li>・ 低消費電力モード移行時に消灯</li> <li>・ 低消費電力モード解除後に点灯</li> </ul>

注.1 RSK の SW 端子に割り当てられている IRQ はディープソフトウェアスタンバイの解除に対応していません。このため、RES#端子によるリセットを使用して復帰させてください。

## 3.2 ハードウェア構成

### 3.2.1 ハードウェアの改造

本アプリケーションノートでは消費電力低減機能の低消費電力モードについての動作を確認するため、RSK ボードを改造して実施しています。

RX231 の RSK ボードの場合、MCU 消費電流測定用にジャンパ J8 が用意されていますが、製品出荷時、ジャンパ J8 はボードに実装されていません。抵抗 R55 によって"Shorted Pin1-2"の設定になっています。このため、以下の改造が必要となります。

- 1)抵抗 R55 を外す。
- 2)ジャンパ J8 にジャンパピンを立てる。

図 3.1、図 3.2 に RSK ボードの改造に必要な資料の抜粋を示します。

RSKRX231
6. コンフィグレーション

### 6.3 電源設定

電源設定に関連するオプションリンクを表 6-3、表 6-4 に示します。

Reference	設定	説明	関連
J8 *1	Shorted Pin1-2	Board_VCC を UC_VCC に接続	R55
	All open	MCU 消費電流測定用設定 (J8 の間に電流メータを接続)	R55
J9	Shorted Pin1-2	EXT_BATT を 5V 電源ラインに接続	-
	Shorted Pin2-3	VBUS0 を 5V 電源ラインに接続	J15
	All open	EXT_BATT、VBUS0 を 5V 電源ラインから接続解除	-
R244	実装	レギュレータ出力を 1.8V に設定	U6
	未実装	レギュレータ出力を 3.3V に設定	U6
J11	Shorted Pin1-2	レギュレータ出力を Board_VCC に接続	U6
	Shorted Pin2-3	レギュレータ出力を Board_VCC から接続解除	U6
	All open	設定しないでください	U6

**表 6-3: 電源設定オプションリンク (1)**

\*1: 製品出荷時、ジャンパ J8 はボードに実装されていませんが、抵抗 R55 によって"Shorted Pin1-2"の設定になっています。

図 3.1 RX231 の RSK ボードのユーザーズマニュアルの抜粋

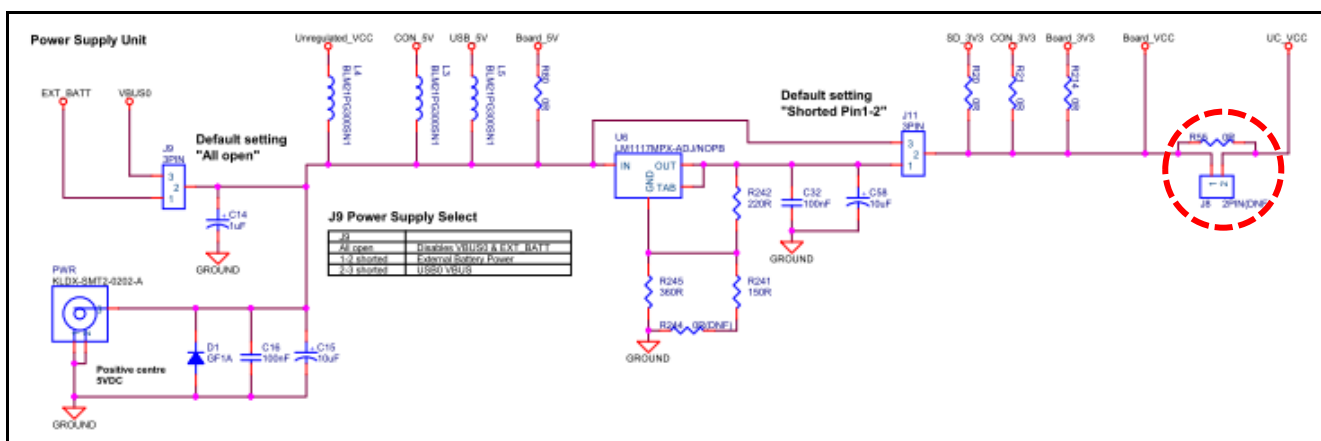


図 3.2 RX231 の RSK ボードの回路図の抜粋

詳細はアプリケーションノート「Renesas Starter Kit for RX231 ユーザーズマニュアル (R20UT3027)」、  
「Renesas Starter Kit for RX231 CPU ボード回路図 (R20UT3026)」を参照してください。

#### 4. ソフトウェア説明

本アプリケーションノートのサンプルプログラムは、低消費電力モードごとにプロジェクトが分かれています。詳細は 4.3 章を参照ください。

本アプリケーションノートのサンプルプログラムでは、低消費電力モードの移行トリガが発生すると低消費電力モードへ移行し、低消費モードへ移行した状態で低消費電力モードの移行トリガが発生すると低消費電力モードを解除します。

低消費電力モードの移行トリガには IRQ を用いています。表 4.1 に IRQ の設定内容を示します。

表 4.1 IRQ の設定内容

設定項目	設定内容
検出タイプ	立ち下がリエッジ
デジタルフィルタ	無効
優先順位	レベル 15 (最高)



## 4.1 動作概要

### 4.1.1 スリープモード

初期設定後、LED を点灯し、低消費電力モードの移行トリガの発生を待ちます。

RX65N の場合、初期設定後、TSIP を除くモジュールストップ状態を解除しています。

低消費電力モードの移行トリガが発生すると、LED を消灯し、スリープモードへ移行します。

スリープモード中に低消費電力モードの移行トリガが発生すると、スリープモードを解除し、LED を点灯します。

図 4.1 にスリープモードの動作概要を示します。

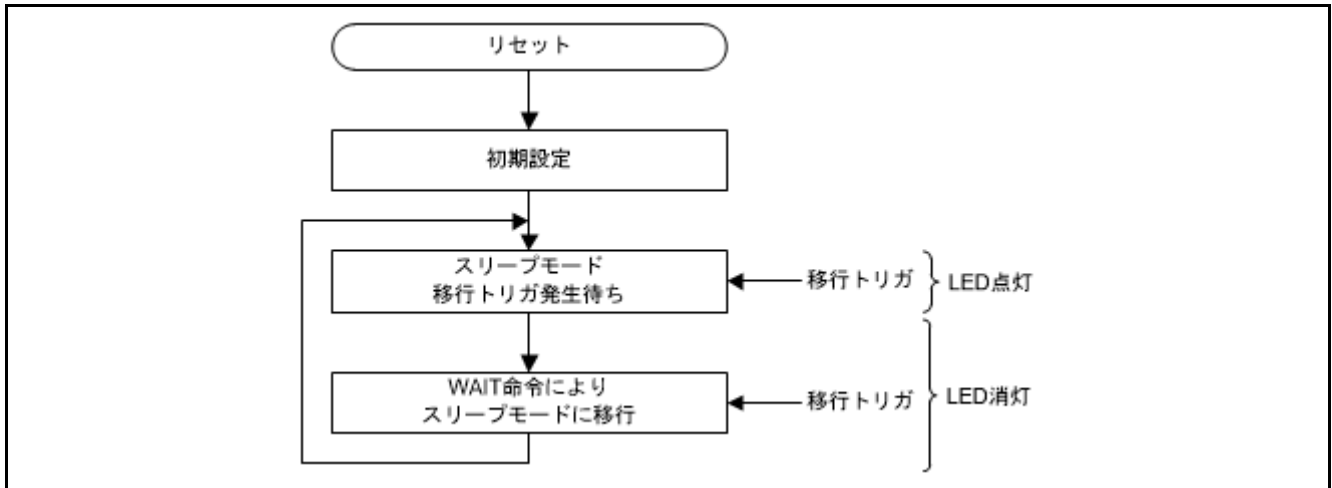


図 4.1 スリープモードの動作概要

### 4.1.2 ディープスリープモード

初期設定後、LED を点灯し、低消費電力モードの移行トリガの発生を待ちます。

低消費電力モードの移行トリガが発生すると、LED を消灯し、ディープスリープモードへ移行します。

ディープスリープモード中に低消費電力モードの移行トリガが発生すると、ディープスリープモードを解除し、LED を点灯します。

図 4.2 にディープスリープモードの動作概要を示します。

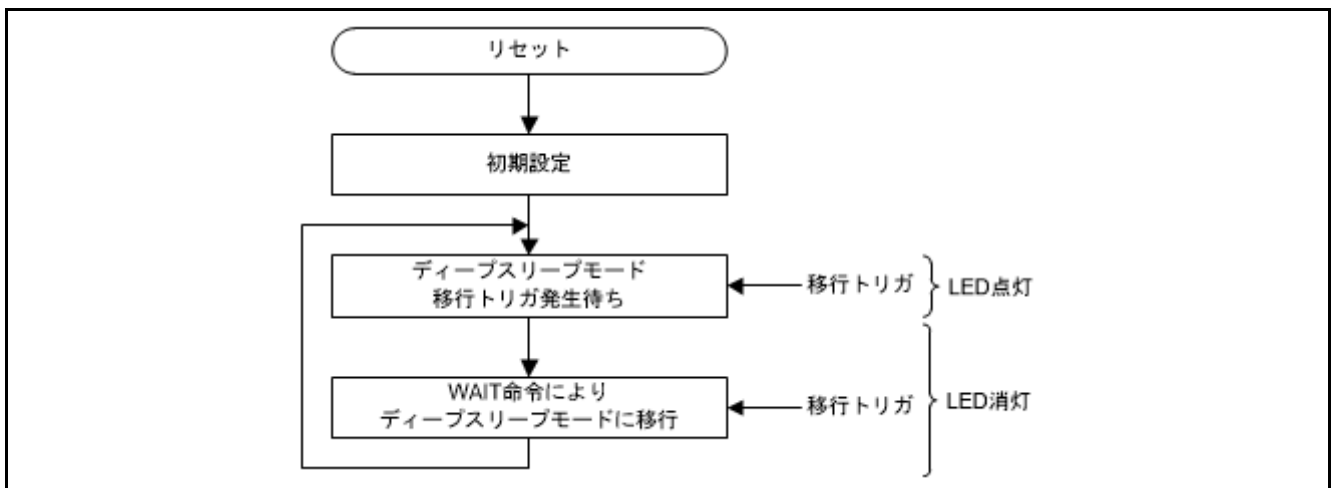


図 4.2 ディープスリープモードの動作概要

#### 4.1.3 全モジュールクロックストップモード

初期設定後、LED を点灯し、低消費電力モードの移行トリガの発生を待ちます。

低消費電力モードの移行トリガが発生すると、LED を消灯し、全モジュールクロックストップモードへ移行します。

全モジュールクロックストップモード中に低消費電力モードの移行トリガが発生すると、全モジュールクロックストップモードを解除し、LED を点灯します。

図 4.3 に全モジュールクロックストップモードの動作概要を示します。

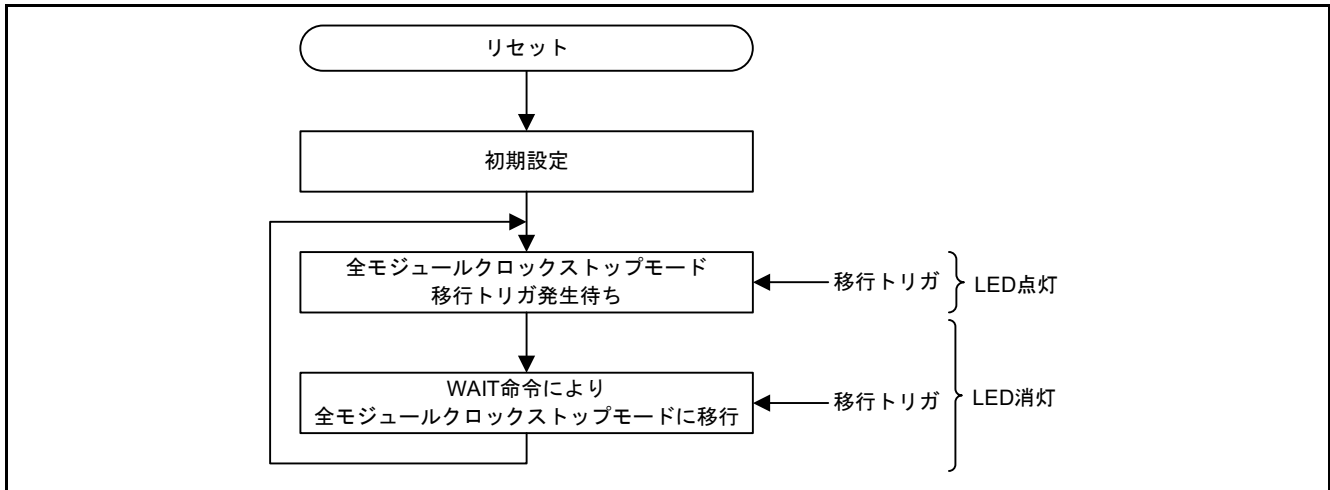


図 4.3 全モジュールクロックストップモードの動作概要

#### 4.1.4 ソフトウェアスタンバイモード

初期設定後、LED を点灯し、低消費電力モードの移行トリガの発生を待ちます。

低消費電力モードの移行トリガが発生すると、LED を消灯し、ソフトウェアスタンバイモードへ移行します。

ソフトウェアスタンバイモード中に低消費電力モードの移行トリガが発生すると、ソフトウェアスタンバイモードを解除し、LED を点灯します。

図 4.4 にソフトウェアスタンバイモードの動作概要を示します。

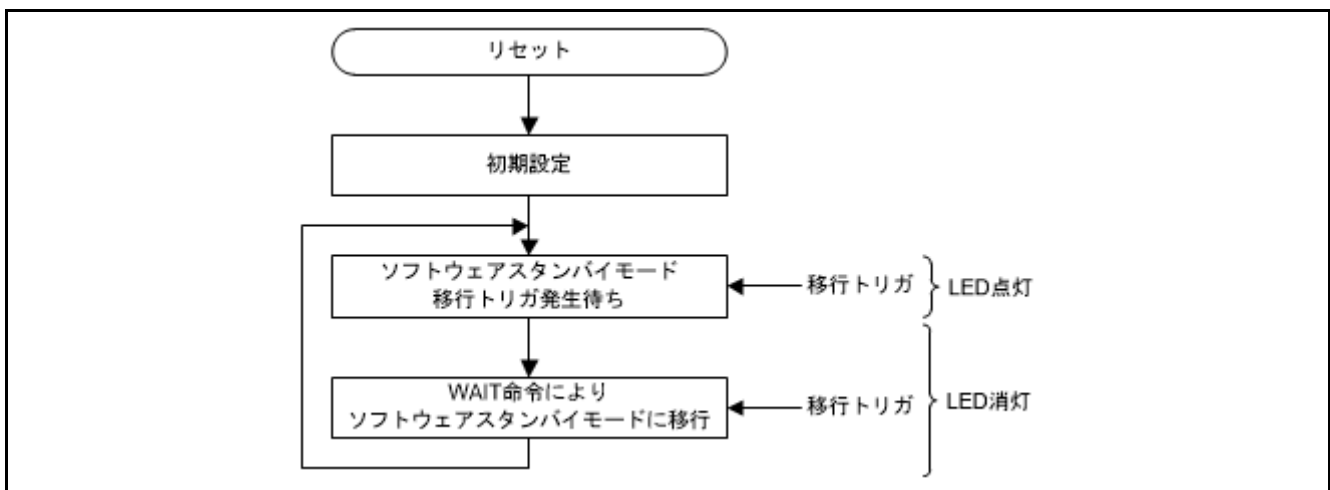


図 4.4 ソフトウェアスタンバイモードの動作概要

## 4.1.5 ディープソフトウェアスタンバイモード

初期設定後、LED を点灯し、低消費電力モードの移行トリガの発生を待ちます。

低消費電力モードの移行トリガが発生すると、LED を消灯し、ディープソフトウェアスタンバイモードへ移行します。

ディープソフトウェアスタンバイモード中に RES# 端子によるリセットや RTC のアラーム割り込みが発生すると、ディープソフトウェアスタンバイモードは解除されます。

図 4.5、図 4.6 にディープソフトウェアスタンバイモードの動作概要を示します。

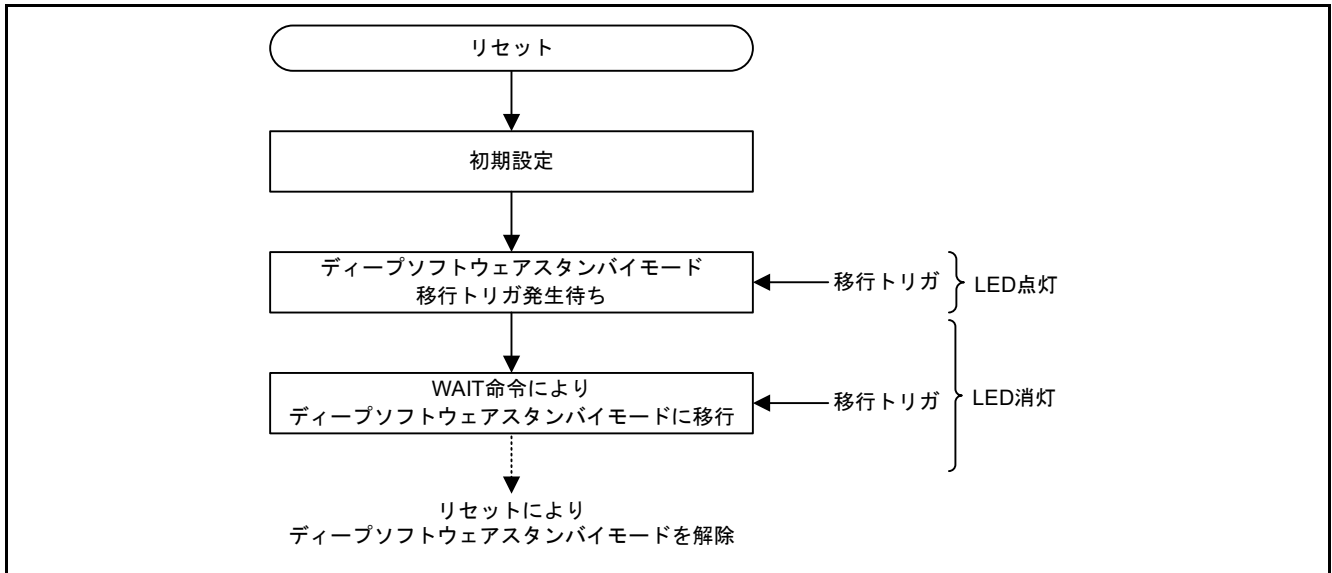


図 4.5 ディープソフトウェアスタンバイモードの動作概要(リセットによるモード解除)

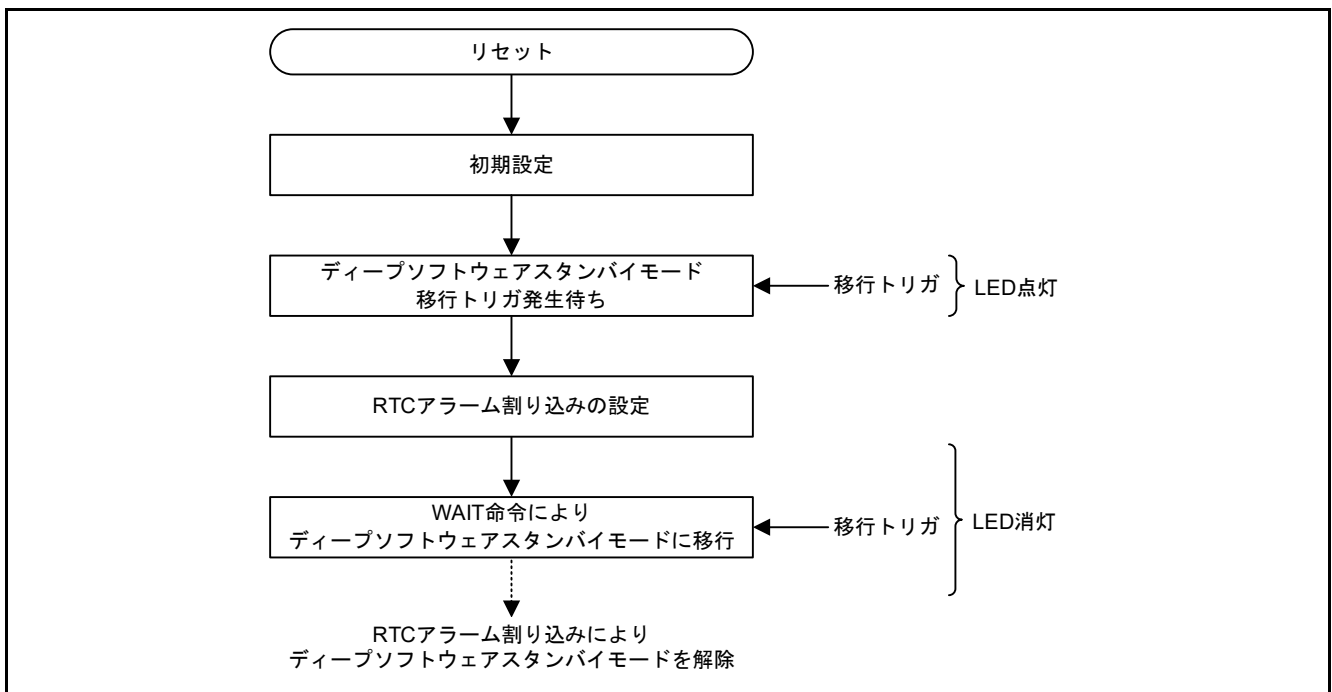


図 4.6 ディープソフトウェアスタンバイモードの動作概要(アラーム割り込みによるモード解除)

## 4.2 モジュール構成

本アプリケーションノートのサンプルプログラムでは、表 4.2 に示すコンポーネントに関するソースコードをスマート・コンフィグレータのコード生成機能によって生成したものを組み合わせて使用しています。

図 4.7 に本アプリケーションノートのサンプルプログラムのモジュール構成を示します。

表 4.2 コンポーネント一覧

コンポーネント	バージョン	リソース
ボードサポートパッケージ (BSP)	7.20	—
リアルタイムクロック	1.8.0	RTC
消費電力低減機能	2.3.0	LPC
割り込みコントローラ	2.3.0	ICU

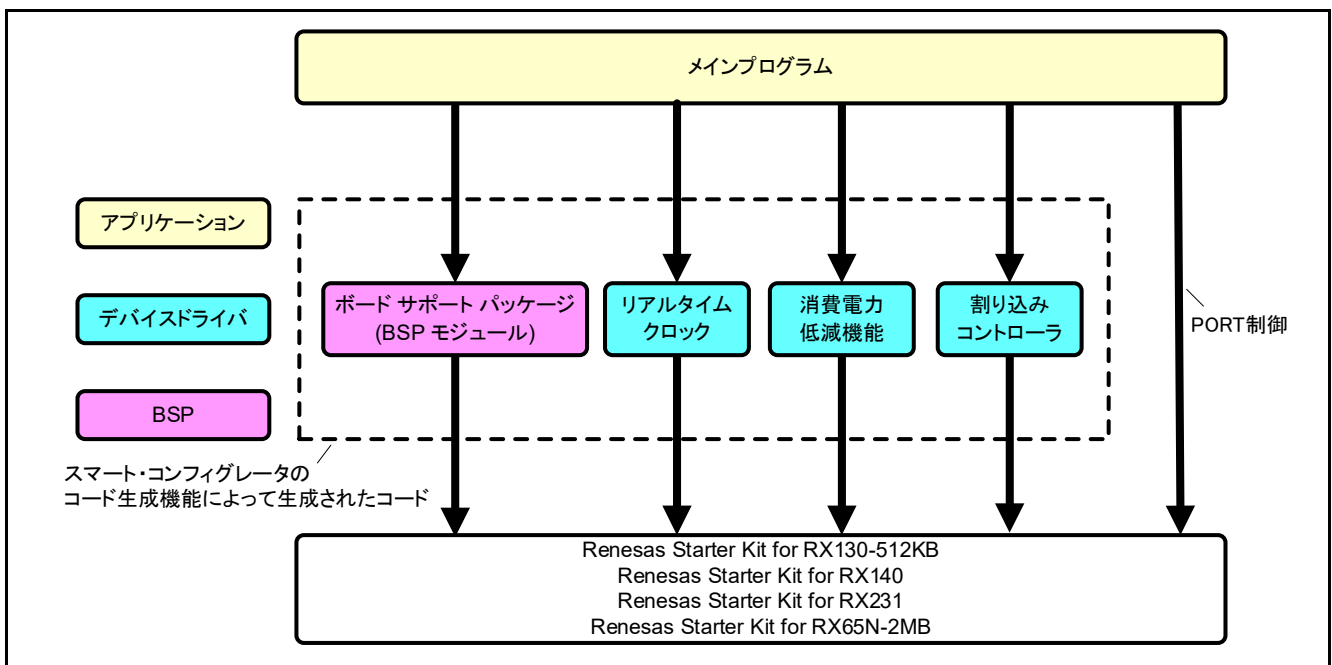


図 4.7 モジュール構成

### 4.3 プロジェクト構成

表 4.3 と表 4.4 に本アプリケーションノートでのプロジェクト構成の一覧を図 4.8 に本アプリケーションノートのプロジェクト構成を示します。

表 4.3 プロジェクト構成一覧(1/2)

対象デバイス	プロジェクト名	内容
RX130	power_save_sleep_rx130	RX130-512KB スリープモード確認用プロジェクト
	power_save_deep_sleep_rx130	RX130-512KB ディープスリープモード確認用プロジェクト
	power_save_software_standby_rx130	RX130-512KB ソフトウェアスタンバイモード確認用プロジェクト
	power_save_software_standby_rtc_operation_rx130	RX130-512KB RTC 動作時のソフトウェアスタンバイモード確認用プロジェクト
RX140	power_save_sleep_rx140	RX140 スリープモード確認用プロジェクト
	power_save_deep_sleep_rx140	RX140 ディープスリープモード確認用プロジェクト
	power_save_software_standby_rx140	RX140 ソフトウェアスタンバイモード確認用プロジェクト
	power_save_software_standby_rtc_operation_rx140	RX140 RTC 動作時のソフトウェアスタンバイモード確認用プロジェクト
RX231	power_save_sleep_rx231	RX231 スリープモード確認用プロジェクト
	power_save_deep_sleep_rx231	RX231 ディープスリープモード確認用プロジェクト
	power_save_software_standby_rx231	RX231 ソフトウェアスタンバイモード確認用プロジェクト
	power_save_software_standby_rtc_operation_rx231	RX231 RTC 動作時のソフトウェアスタンバイモード確認用プロジェクト

表 4.4 プロジェクト構成一覧(2/2)

対象デバイス	プロジェクト名	内容
RX65N	power_save_sleep_rx65n	RX65N-2MB スリープモード確認用プロジェクト
	power_save_all_module_clock_stop_rx65n	RX65N-2MB 全モジュールクロックストップモード確認用プロジェクト
	power_save_software_standby_rx65n	RX65N-2MB ソフトウェアスタンバイモード確認用プロジェクト
	power_save_deep_software_standby_rx65n	RX65N-2MB ディープソフトウェアスタンバイモード確認用プロジェクト
	power_save_deep_software_standby_rtc_operation_rx65n	RX65N-2MB RTC 動作時のディープソフトウェアスタンバイモード確認用プロジェクト(リセットによりディープソフトウェアスタンバイモードを解除)
	power_save_deep_software_standby_rtc2_operation_rx65n	RX65N-2MB RTC 動作時のディープソフトウェアスタンバイモード確認用プロジェクト(アラーム割り込みによりディープソフトウェアスタンバイモードを解除)

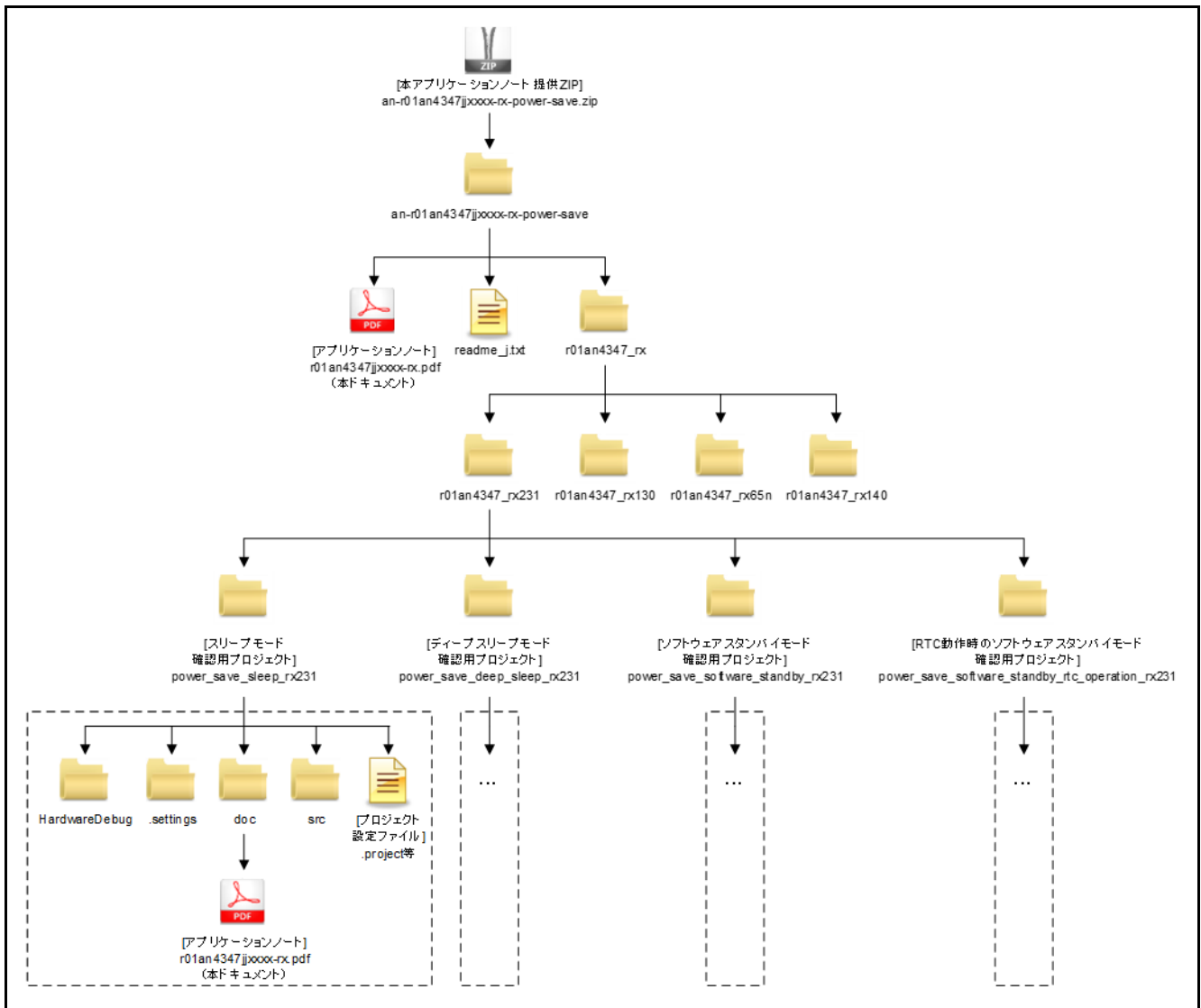


図 4.8 プロジェクト構成

本アプリケーションノートが提供する zip ファイルを解凍すると、同名のフォルダが作成され、その中に各フォルダやファイルが入っています。

「r01an4347\_rx」フォルダ下に対象デバイス毎のフォルダが展開され、その配下に低消費電力モード毎のプロジェクトが展開されます。各プロジェクトは e<sup>2</sup>studio のワークスペースにインポートすることで動作させることができます。

#### 4.4 ファイル構成

本サンプルプログラムで使用するファイルの構成を以下に示します。

メインプログラムは“プロジェクトフォルダ¥src¥”フォルダ配下に生成されます。

例) RX231 用のスリープモード確認用プロジェクトの場合  
 “power\_save\_sleep\_rx231¥src¥main.c”

表 4.5 に消費電力モード確認用プロジェクトのメインプログラムに関連するファイルを示します。

**表 4.5 メインプログラムに関連するファイル**

ファイル名	概要
main.c	消費電力モード確認用メイン処理

スマート・コンフィグレータのコード生成機能によって、  
 “プロジェクトフォルダ¥src¥smc\_gen”フォルダ配下にソースコードが生成されます。

例) RX231 用のスリープモード確認用プロジェクトの場合  
 “power\_save\_sleep\_rx231¥src¥smc\_gen”

表 4.6 にスマート・コンフィグレータのコード生成機能によって生成されたソースコードに変更を加えているものを示します。フォルダ名は“プロジェクトフォルダ¥src¥smc\_gen”までを省略しています。

**表 4.6 スマート・コンフィグレータのコード生成機能によって生成されたソースコードに変更を加えているファイル**

フォルダ名	ファイル名	概要	備考
Config_ICU	Config_ICU_user.c	ユーザ実装用の割り込み処理	r_Config_ICU_irq4_interrupt()関数に処理を追加、詳細は 4.10.5 章を参照

スマート・コンフィグレータのコード生成機能によって生成されたソースコードをそのまま使用している部分については省略します。



#### 4.5 オプション設定メモリ

表 4.7、表 4.8 にサンプルプログラム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 4.7 RX130-512KB、RX140、RX231 用のサンプルプログラム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 と WDT(注 1)は停止
OFS1	FFFF FF8Bh~FFFF FF88h	FFFF FFFFh	起動時間短縮無効 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
MDE	FFFF FF83h~FFFF FF80h	FFFF FFFFh	リトルエンディアン

注.1 WDT は RX231 グループの場合のみ。

**表 4.8 RX65N-2MB 用のサンプルプログラム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	FE7F 5D04h~FE7F 5D07h	FFFF FFFFh	リセット後、IWDT と WDT は停止
OFS1	FE7F 5D08h~FE7F 5D0Bh	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
MDE	FE7F 5D00h~FE7F 5D03h	FFFF FFFFh	リトルエンディアン

#### 4.6 定数一覧

表 4.9 にサンプルプログラムで使用する定数を示します。

**表 4.9 サンプルプログラムで使用する定数**

定数名	設定値	内容
LED_ON	(0)	LED の点灯
LED_OFF	(1)	LED の消灯
LED0	(PORTD.PODR.BIT.B3)	LED0(PD3)の PODR レジスタのビット (RX130-512KB 用のサンプルプログラムで使用)
	(PORT2.PODR.BIT.B1)	LED0(P21)の PODR レジスタのビット (RX140 用のサンプルプログラムで使用)
	(PORT1.PODR.BIT.B7)	LED0(P17)の PODR レジスタのビット (RX231 用のサンプルプログラムで使用)
	(PORT7.PODR.BIT.B3)	LED0(P73)の PODR レジスタのビット (RX65N-2MB 用のサンプルプログラムで使用)
ALARMSET	(30)	ディープソフトウェアスタンバイモードに遷移してから RTC のアラーム割り込みによってモード解除するまでの秒数 (RX65N-2MB 用のサンプルプログラムで使用)

#### 4.7 変数一覧

表 4.10 にグローバル変数を示します。

表 4.10 グローバル変数

型	変数名	内容
extern volatile bool	g_sw_pressed	SW が押下されているかの状態 false : 押下されていない true : 押下されている

#### 4.8 関数一覧

表 4.11 にサンプルプログラムで使用する関数を示します。

表 4.11 サンプルプログラムで使用する関数

関数名	概要
main	メイン処理
save_power_port_init	ポートの初期設定
save_power_peripheral_init	周辺機能の初期設定
save_power_wait_trigger	低消費モード移行トリガ待ち
alarm_setting	RTC アラーム割り込みの設定 (RX65N-2MB 用のサンプルプログラムで使用)
convert_to_dec	10 進数への変換処理 (RX65N-2MB 用のサンプルプログラムで使用)
r_Config_ICU_irq2_interrupt(注 1)	IRQ2 割り込み処理 (RX130-512KB 用、RX140 用のサンプルプログラムで使用)
r_Config_ICU_irq4_interrupt(注 1)	IRQ4 割り込み処理 (RX231 用のサンプルプログラムで使用)
r_Config_ICU_irq13_interrupt(注 1)	IRQ13 割り込み処理 (RX65N-2MB 用のサンプルプログラムで使用)

注.1 スマート・コンフィグレータのコード生成機能によって生成される関数のうち、ソースコードに変更を加えているものを示しています。スマート・コンフィグレータのコード生成機能によって生成されたソースコードをそのまま使用している部分については省略しています。

## 4.9 関数仕様

サンプルプログラムの関数仕様を以下に示します。

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、低消費電力モードへの移行、解除を行います。
引数	なし
リターン値	なし

save_power_port_init	
概要	ポートの初期設定
ヘッダ	なし
宣言	void save_power_port_init (void)
説明	ポートの初期設定を行います。
引数	なし
リターン値	なし

save_power_peripheral_init	
概要	周辺機能の初期設定
ヘッダ	なし
宣言	void save_power_peripheral_init (void)
説明	周辺機能の初期設定を行います。
引数	なし
リターン値	なし

save_power_wait_trigger	
概要	低消費モード移行トリガ待ち
ヘッダ	なし
宣言	void save_power_wait_trigger (void)
説明	低消費モード移行トリガ(IRQ)が発生するのを待ちます。
引数	なし
リターン値	なし

alarm_setting	
概要	RTC アラーム割り込みの設定
ヘッダ	なし
宣言	void alarm_setting(void)
説明	RTC のアラーム割り込みによってディープソフトウェアスタンバイモードを解除する時間を設定します。 サンプルプログラムでは分と秒のみ対応しています。 (RX65N-2MB 用のサンプルプログラムで使用)
引数	なし
リターン値	なし

convert_to_dec	
概要	10 進数への変換処理
ヘッダ	なし
宣言	uint8_t convert_to_dec(uint8_t time)
説明	分と秒カウンタビットから得た値を 10 進数に変換します。 (RX65N-2MB 用のサンプルプログラムで使用)
引数	uint8_t time
リターン値	uint8_t time_dec
r_Config_ICU_irq2_interrupt	
概要	IRQ2 割り込み処理
ヘッダ	Config_ICU.h
宣言	static void r_Config_ICU_irq2_interrupt (void)
説明	IRQ2 の割り込み処理を行います。 IRQ2 の割り込み処理では SW 押下状態(g_sw_pressed)を true にセットします。 (RX130-512KB 用、RX140 用のサンプルプログラムで使用)
引数	なし
リターン値	なし
備考	本関数はスマート・コンフィグレータのコード生成機能によって生成されます。
r_Config_ICU_irq4_interrupt	
概要	IRQ4 割り込み処理
ヘッダ	Config_ICU.h
宣言	static void r_Config_ICU_irq4_interrupt (void)
説明	IRQ4 の割り込み処理を行います。 IRQ4 の割り込み処理では SW 押下状態(g_sw_pressed)を true にセットします。 (RX231 用のサンプルプログラムで使用)
引数	なし
リターン値	なし
備考	本関数はスマート・コンフィグレータのコード生成機能によって生成されます。
r_Config_ICU_irq13_interrupt	
概要	IRQ13 割り込み処理
ヘッダ	Config_ICU.h
宣言	static void r_Config_ICU_irq13_interrupt (void)
説明	IRQ13 の割り込み処理を行います。 IRQ13 の割り込み処理では SW 押下状態(g_sw_pressed)を true にセットします。 (RX65N-2MB 用のサンプルプログラムで使用)
引数	なし
リターン値	なし
備考	本関数はスマート・コンフィグレータのコード生成機能によって生成されます。

## 4.10 フローチャート

RX231 用のサンプルプログラムのフローチャートを一例として以下に示します。

また、ディープソフトウェアスタンバイモードに遷移するサンプルプログラムのメイン処理については RX65N を例として以下に示します。

本アプリケーションノートのサンプルプログラムはスマート・コンフィグレータのコード生成機能によって生成されたコードが含まれます。スマート・コンフィグレータのコード生成機能によって生成されたソースコードに変更を加えている部分については 4.10.5 章に示します。スマート・コンフィグレータのコード生成機能によって生成されたソースコードをそのまま使用している部分のフローチャートについては省略します。

### 4.10.1 メイン処理

図 4.9 にメイン処理のフローチャートを示します。

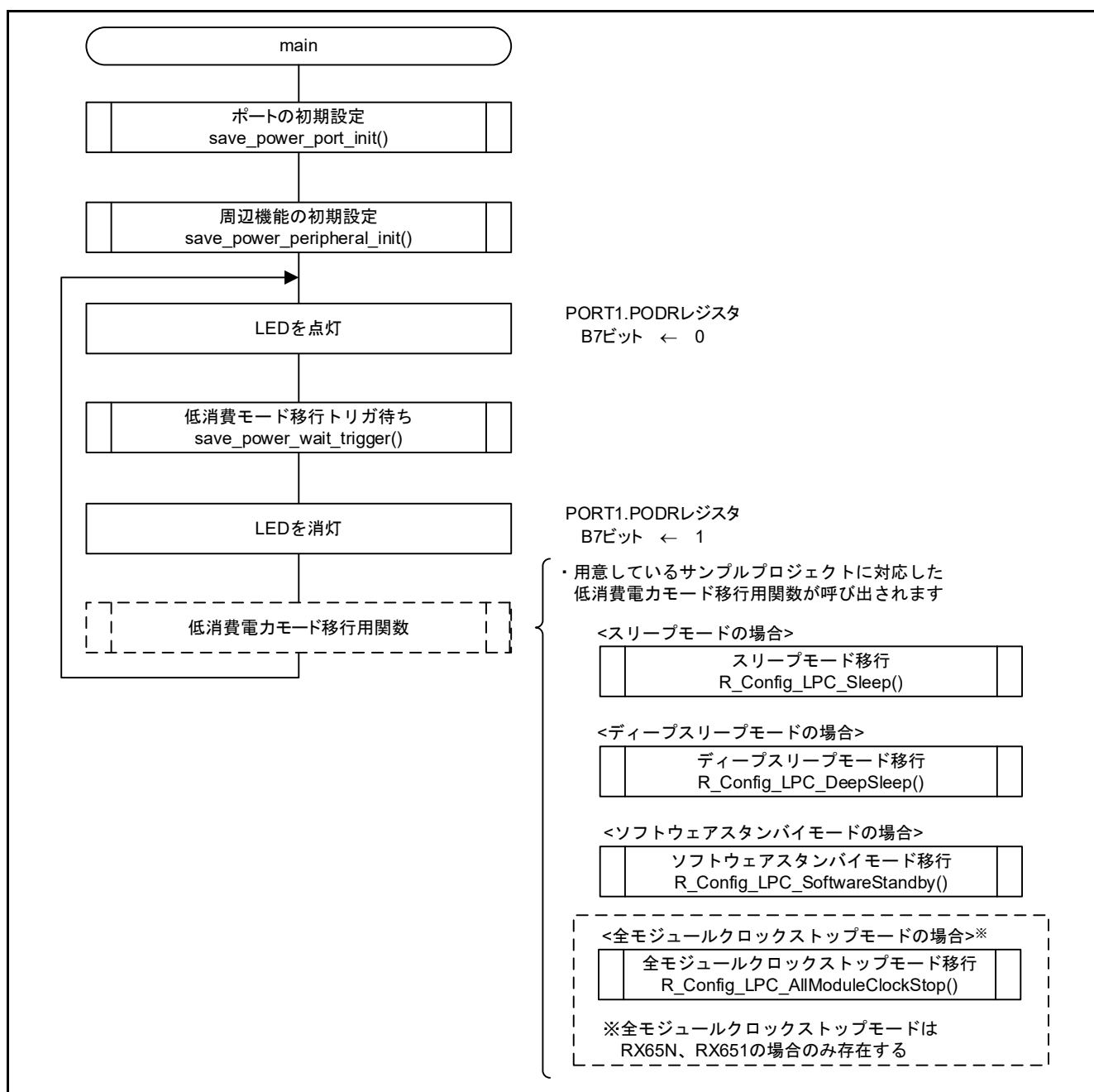


図 4.9 メイン処理

図 4.10 にディープソフトウェアスタンバイモードに遷移するサンプルプログラムのメイン処理のフローチャートを示します。

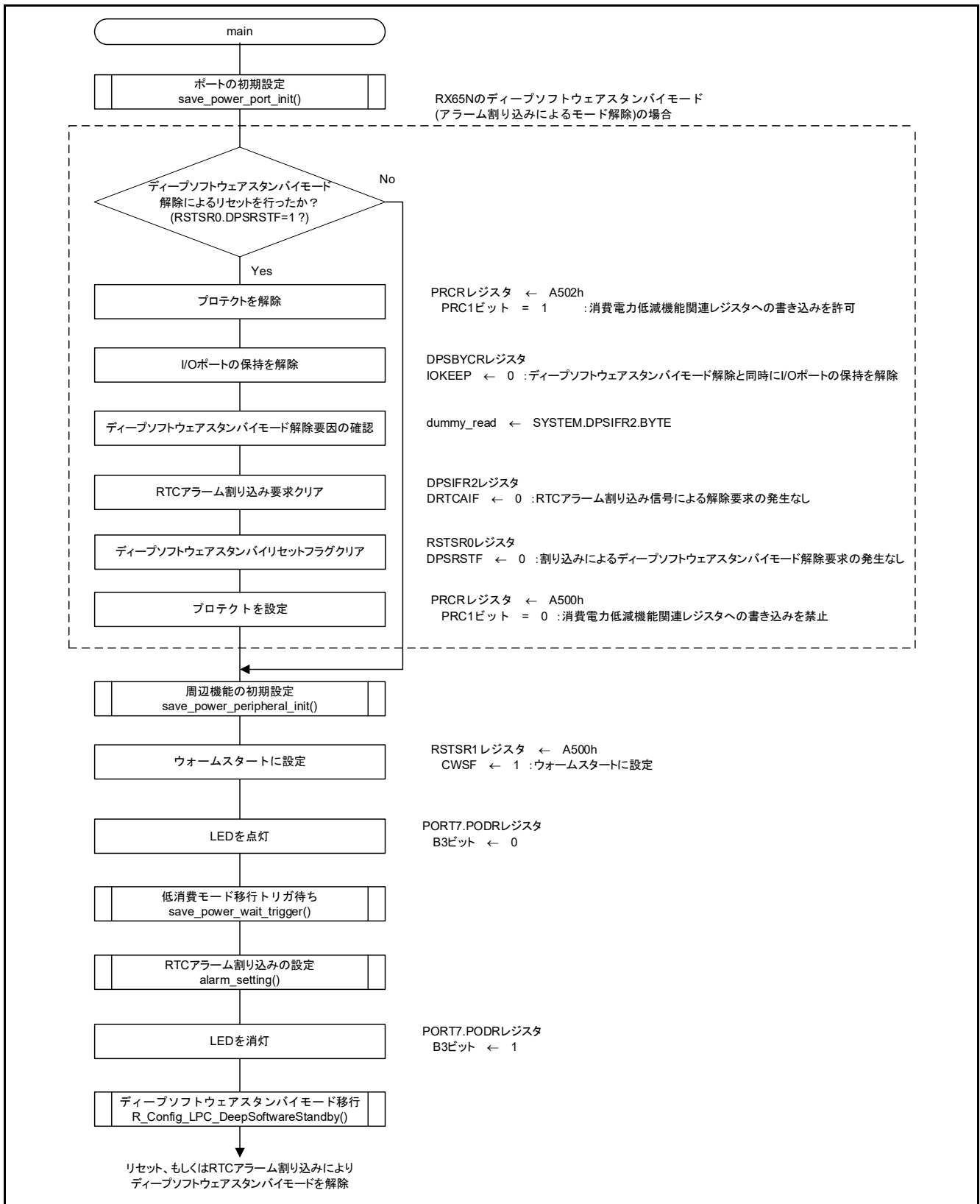


図 4.10 ディープソフトウェアスタンバイモードに遷移するサンプルプログラムのメイン処理

## 4.10.2 ポートの初期設定

図 4.11 にポートの初期設定のフローチャートを示します。

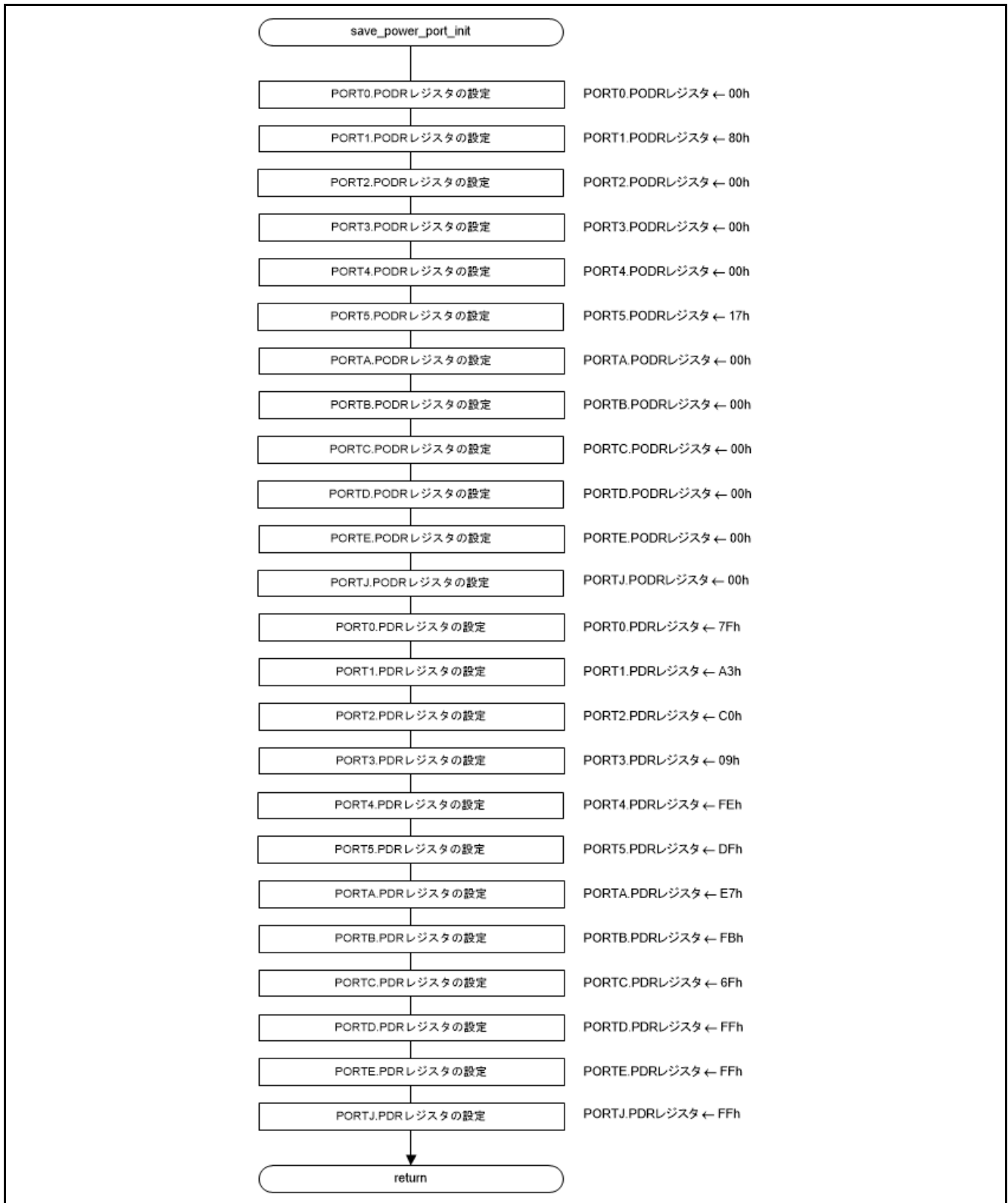


図 4.11 ポートの初期設定

## 4.10.3 周辺機能の初期設定

図 4.12 に周辺機能の初期設定のフローチャートを示します。

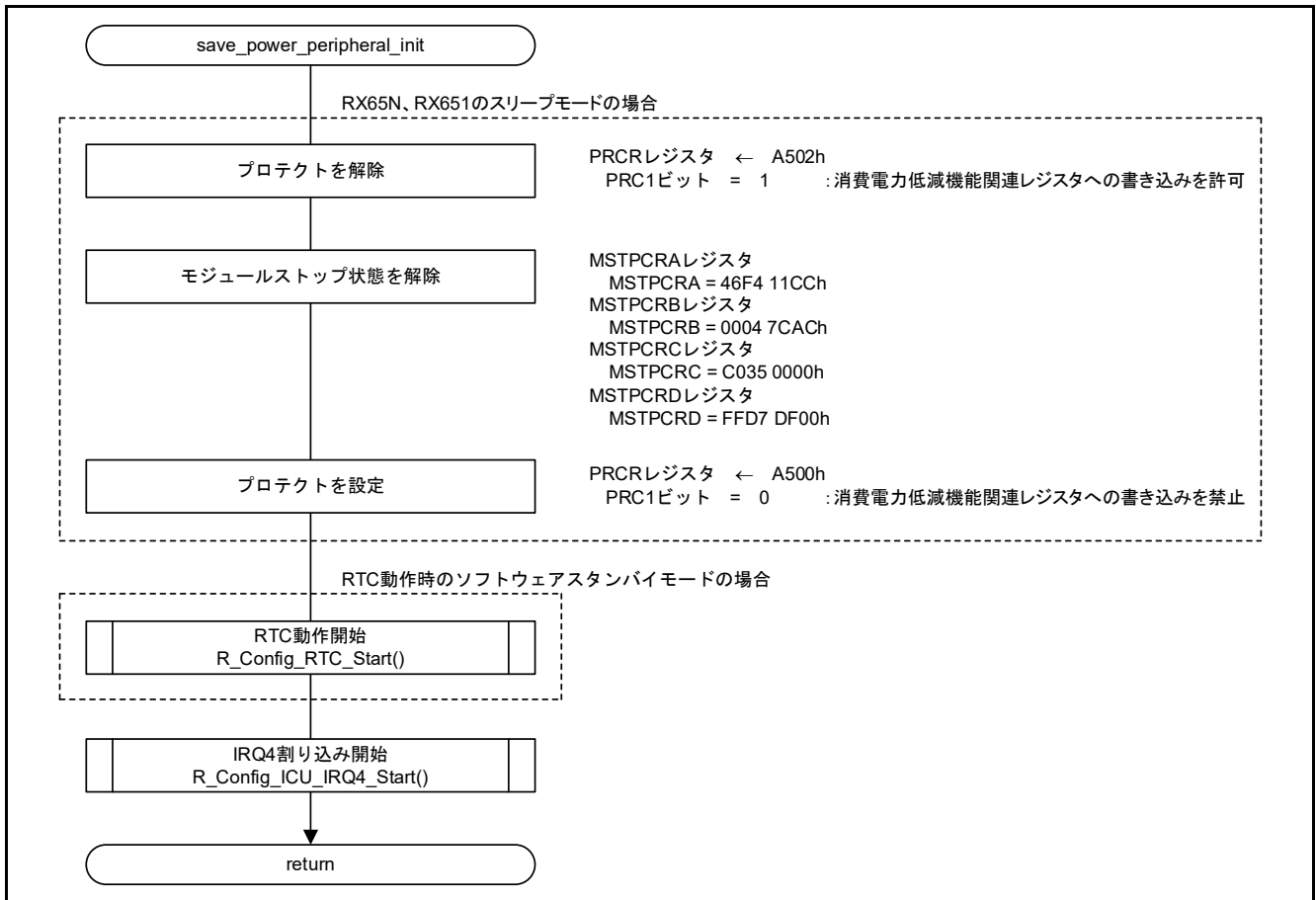


図 4.12 周辺機能の初期設定



## 4.10.4 低消費モード移行トリガ待ち

図 4.13 に低消費モード移行トリガ待ちのフローチャートを示します。

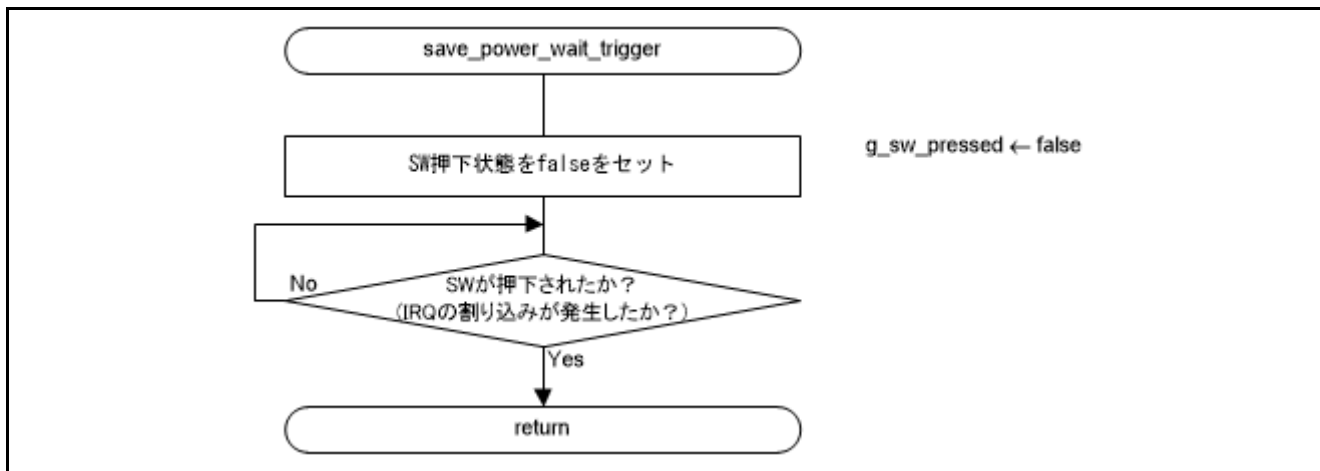


図 4.13 低消費モード移行トリガ待ち

## 4.10.5 RTC アラーム割り込みの設定

図 4.14 に RTC アラーム割り込みの設定のフローチャートを示します。

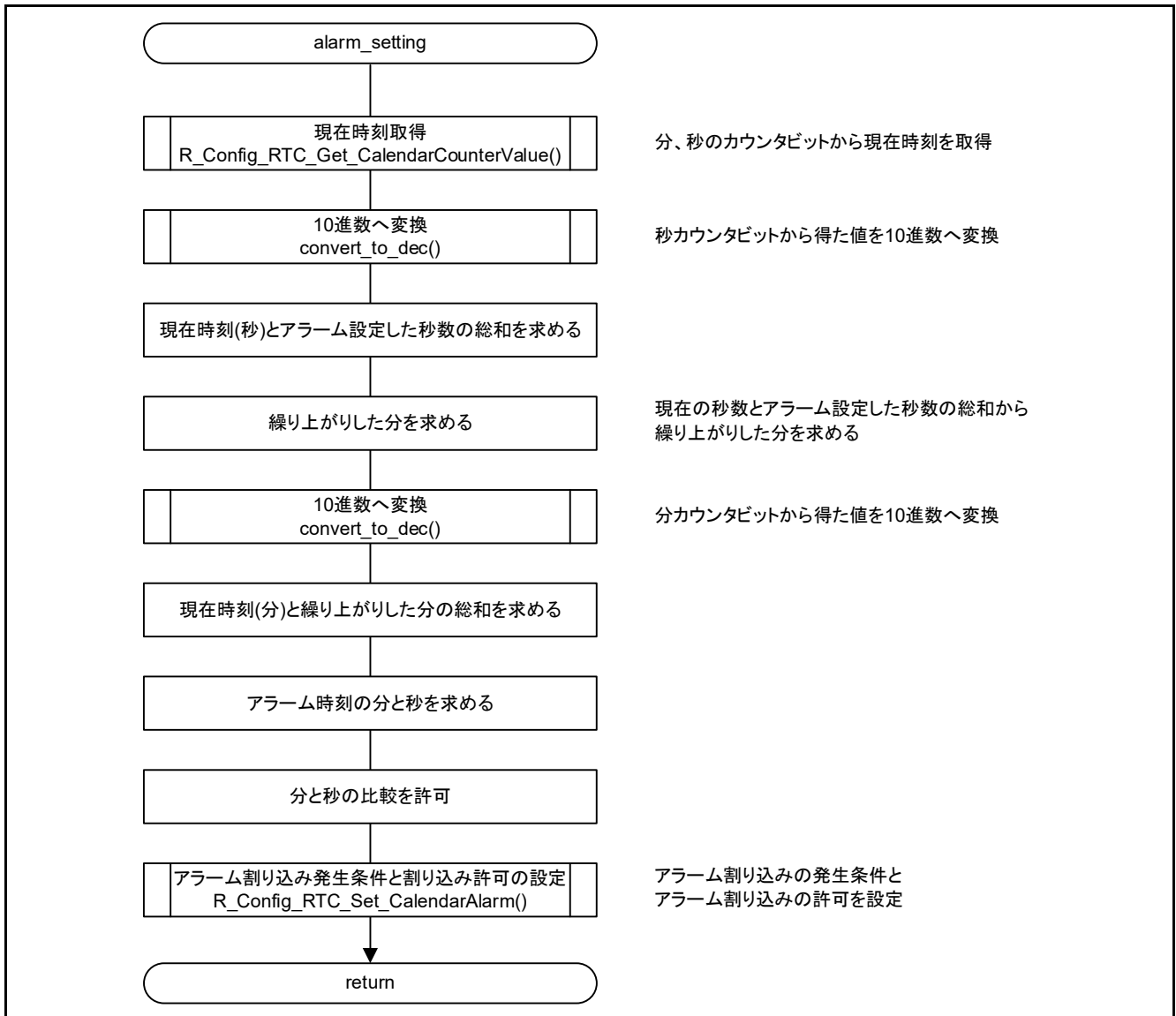


図 4.14 RTC アラーム割り込みの設定

## 4.10.6 10進数への変換処理

図 4.15 に 10 進数への変換処理のフローチャートを示します。

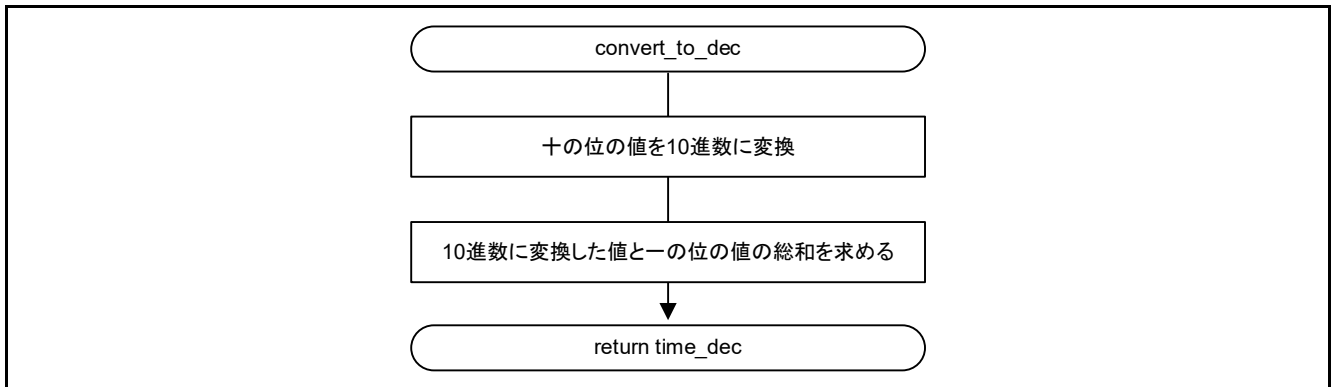


図 4.15 10 進数への変換処理

## 4.10.7 IRQ4 割り込み処理

図 4.16 に IRQ4 割り込み処理のフローチャートを示します。

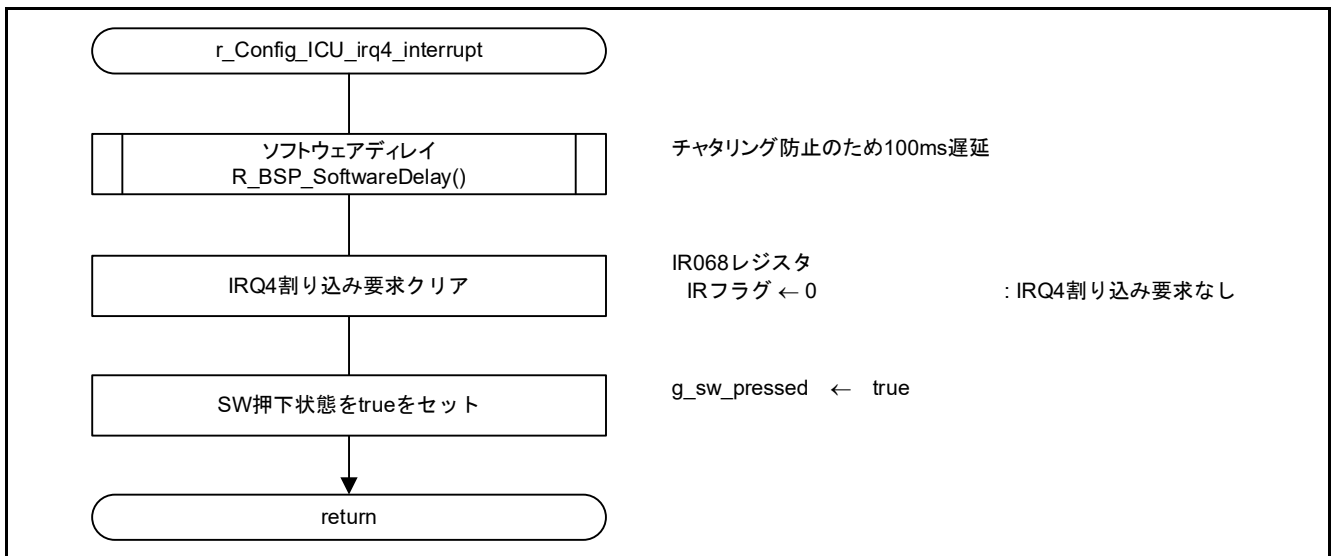


図 4.16 IRQ4 割り込み処理

## 5. スマート・コンフィグレータのコンフィグレーションについて

本サンプルプロジェクトのスマート・コンフィグレータのコンフィグレーションについては、各サンプルプロジェクトのスマート・コンフィグレータのコンフィグレーション画面に設定が保持されています。

スマート・コンフィグレータの使用方法に関しては関連ドキュメントの「Renesas e<sup>2</sup> studio スマート・コンフィグレータ ユーザーガイド(R20AN0451)」をご確認ください。

## 6. 低消費電力モードへ移行する際のノウハウ

- (1) スリープモードなど、遷移する低消費電力モードによってはクロックが動作しています。消費電流を抑えるために不要なクロックは停止、または分周比を変更して周波数を下げてください。
- (2) 各周辺機能ではモジュールストップ状態や低消費電力モードへ移行する際に、注意事項がある場合があります。詳細は使用される各周辺機能の注意事項をご確認ください。
- (3) ソフトウェアスタンバイモードなど、遷移する低消費電力モードによっては I/O ポートの動作が保持された状態であるため、遷移する前に適切な入出力設定を行ってください。  
表 6.1 に I/O ポートの設定例を示します。

表 6.1 I/O ポートの設定例

I/O ポートの状態	消費電流を低減させるためのポート設定
入力設定	入力レベルを固定(High または Low)してください。 なお、入力に設定することで、スタンバイモード中はハイインピーダンスになる端子が多く存在します。
出力設定	外部のプルアップ/プルダウン抵抗に電位差を生じさせないレベルを出力してください。 接続相手が入力、且つ外部/内部ともにプルアップ/プルダウン抵抗がない場合は、Low を出力してください。

- (4) 電流計測時にエミュレータが接続された状態で計測した場合、消費電流が増えます。電流測定対象のボードにエミュレータが接続されていないことを確認してください。
- (5) e<sup>2</sup> studio のデバッグ・モードで消費電流を測定するためのプログラムを書き込んだ場合、消費電流を正しく計測することができません。  
Renesas Flash Programmer などのフラッシュ書き込みソフトウェアツールを使用して、消費電流を測定するためのプログラムを書き込んでください。

## 7. 付録

## 7.1 プロジェクト毎の動作周波数

表 7.1 ~ 表 7.11 に本アプリケーションノートの動作確認時の動作周波数を示します。

**表 7.1 RX130-512KB スリープモード、ディープスリープモード確認用プロジェクトの動作確認周波数**

名称	動作周波数	クロックソース	分周比	通倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	0.5 MHz	PLL	64	—
周辺モジュールクロック D(PCLKD)	0.5 MHz	PLL	64	—
FlashIF クロック(FCLK)	0.5 MHz	PLL	64	—

**表 7.2 RX130-512KB ソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数**

名称	動作周波数	クロックソース	分周比	通倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	32 MHz	PLL	1	—
周辺モジュールクロック D(PCLKD)	32 MHz	PLL	1	—
FlashIF クロック(FCLK)	32 MHz	PLL	1	—

表 7.3 RX130-512KB RTC 動作時のソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	8 MHz	—	—	—
サブクロック	32.768 kHz	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	32 MHz	PLL	1	—
周辺モジュールクロック D(PCLKD)	32 MHz	PLL	1	—
FlashIF クロック(FCLK)	32 MHz	PLL	1	—

表 7.4 RX140 スリープモード、ディープスリープモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	0.5 MHz	PLL	64	—
周辺モジュールクロック D(PCLKD)	0.5 MHz	PLL	64	—
FlashIF クロック(FCLK)	0.5 MHz	PLL	64	—

表 7.5 RX140 ソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	32 MHz	PLL	1	—
周辺モジュールクロック D(PCLKD)	32 MHz	PLL	1	—
FlashIF クロック(FCLK)	32 MHz	PLL	1	—

表 7.6 RX140 RTC 動作時のソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	8 MHz	—	—	—
サブクロック	32.768 kHz	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	32 MHz	メインクロック	2	8
システムクロック(ICLK)	32 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	32 MHz	PLL	1	—
周辺モジュールクロック D(PCLKD)	32 MHz	PLL	1	—
FlashIF クロック(FCLK)	32 MHz	PLL	1	—

表 7.7 RX231 スリープモード、ディープスリープモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	通倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	54 MHz	メインクロック	2	13.5
USB 専用 PLL	停止	—	—	—
システムクロック(ICLK)	54 MHz	PLL	1	—
周辺モジュールクロック A(PCLKA)	0.84375 MHz	PLL	64	—
周辺モジュールクロック B(PCLKB)	0.84375 MHz	PLL	64	—
周辺モジュールクロック D(PCLKD)	0.84375 MHz	PLL	64	—
FlashIF クロック(FCLK)	0.84375 MHz	PLL	64	—
外部バスクロック(BCK)	0.84375 MHz	PLL	64	—

表 7.8 RX231 ソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	通倍率
メインクロック	8 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	54 MHz	メインクロック	2	13.5
USB 専用 PLL	停止	—	—	—
システムクロック(ICLK)	54 MHz	PLL	1	—
周辺モジュールクロック A(PCLKA)	54 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	27 MHz	PLL	2	—
周辺モジュールクロック D(PCLKD)	27 MHz	PLL	2	—
FlashIF クロック(FCLK)	27 MHz	PLL	2	—
外部バスクロック(BCK)	27 MHz	PLL	2	—



表 7.9 RX231 RTC 動作時のソフトウェアスタンバイモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	8 MHz	—	—	—
サブクロック	32.768 kHz	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	54 MHz	メインクロック	2	13.5
USB 専用 PLL	停止	—	—	—
システムクロック(ICLK)	54 MHz	PLL	1	—
周辺モジュールクロック A(PCLKA)	54 MHz	PLL	1	—
周辺モジュールクロック B(PCLKB)	27 MHz	PLL	2	—
周辺モジュールクロック D(PCLKD)	27 MHz	PLL	2	—
FlashIF クロック(FCLK)	27 MHz	PLL	2	—
外部バスクロック(BCK)	27 MHz	PLL	2	—

表 7.10 RX65N-2MB スリープモード、ソフトウェアスタンバイモード、ディープソフトウェアスタンバイモード、全モジュールクロックストップモード確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	24 MHz	—	—	—
サブクロック	停止	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	240 MHz	メインクロック	1	10
システムクロック(ICLK)	120 MHz	PLL	2	—
周辺モジュールクロック A(PCLKA)	120 MHz	PLL	2	—
周辺モジュールクロック B(PCLKB)	60 MHz	PLL	4	—
周辺モジュールクロック C(PCLKC)	60 MHz	PLL	4	—
周辺モジュールクロック D(PCLKD)	60 MHz	PLL	4	—
FlashIF クロック(FCLK)	60 MHz	PLL	4	—
外部バスクロック(BCK)	120 MHz	PLL	2	—
USB クロック(UCK)	48 MHz	PLL	5	—

表 7.11 RX65N-2MB RTC 動作時のディープソフトウェアスタンバイモード  
確認用プロジェクトの動作確認周波数

名称	動作周波数	クロックソース	分周比	逡倍率
メインクロック	24 MHz	—	—	—
サブクロック	32.768 kHz	—	—	—
高速オンチップオシレータ(HOCO)	停止	—	—	—
低速オンチップオシレータ(LOCO)	停止	—	—	—
IWDT 専用オンチップオシレータ	停止	—	—	—
PLL	240 MHz	メインクロック	1	10
システムクロック(ICLK)	120 MHz	PLL	2	—
周辺モジュールクロック A(PCLKA)	120 MHz	PLL	2	—
周辺モジュールクロック B(PCLKB)	60 MHz	PLL	4	—
周辺モジュールクロック C(PCLKC)	60 MHz	PLL	4	—
周辺モジュールクロック D(PCLKD)	60 MHz	PLL	4	—
FlashIF クロック(FCLK)	60 MHz	PLL	4	—
外部バスクロック(BCK)	120 MHz	PLL	2	—
USB クロック(UCK)	48 MHz	PLL	5	—

## 8. プロジェクトをインポートする方法

サンプルプログラムは e<sup>2</sup> studio のプロジェクト形式で提供しています。本章では、e<sup>2</sup> studio へプロジェクトをインポートする方法を示します。インポート完了後、ビルドおよびデバッグの設定を確認してください。

### 8.1 e<sup>2</sup> studio での手順

e<sup>2</sup> studio でご使用になる際は、下記の手順で e<sup>2</sup> studio にインポートしてください。  
(使用する e<sup>2</sup> studio のバージョンによっては画面が異なる場合があります。)

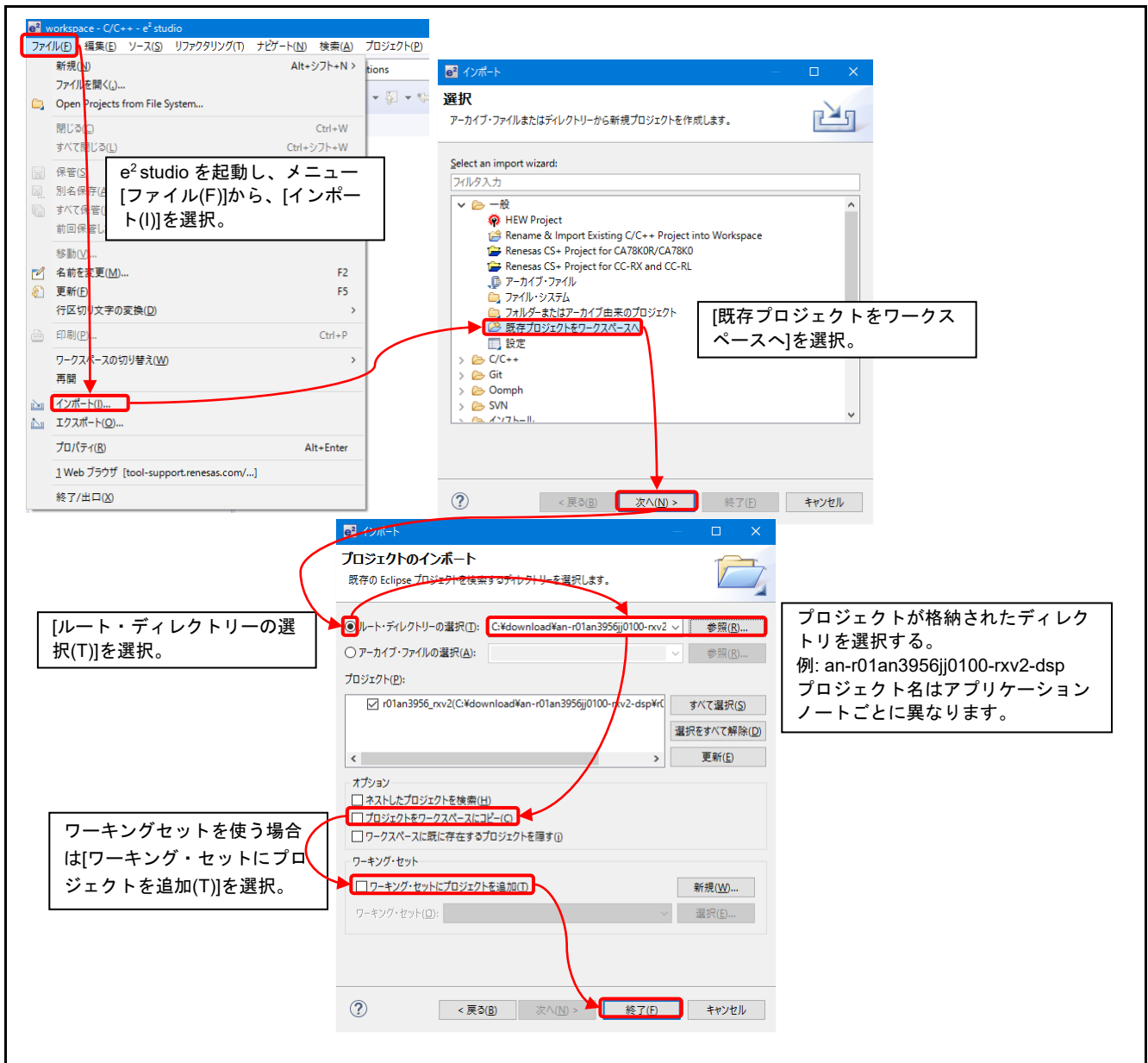


図 8.1 プロジェクトを e<sup>2</sup> studio にインポートする方法

## 8.2 CS+での手順

CS+でご使用になる際は、下記の手順でCS+にインポートしてください。  
(使用するCS+のバージョンによっては画面が異なる場合があります。)

**CS+を起動し、スタート画面から、[e<sup>2</sup> studio/CubeSuite/High-performance Embedded Workshop/PM+のプロジェクトを開く]を選択。**

**プロジェクトを選択する。  
例: r01an3956\_rvx2  
プロジェクト名はアプリケーションノートごとに異なります。**

**拡張子[.rcpc]のファイルを選択して[開く]ボタンを押す。**

**[e<sup>2</sup> studio用プロジェクト・ファイル(\*.rcpc)]を選択。**

**使用するマイクロコントローラを選択してください。**

**プロジェクト種類: 「空のアプリケーション(CC-RX)」を選択し、プロジェクト名と作成場所を指定してください。**

図 8.2 プロジェクトを e<sup>2</sup> studio にインポートする方法

## 9. 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。

## 10. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX130 グループ ユーザーズマニュアル ハードウェア編 (R01UH0560)

RX140 グループ ユーザーズマニュアル ハードウェア編 (R01UH0905)

RX230 グループ、RX231 グループ ユーザーズマニュアル ハードウェア編 (R01UH0496)

RX65N グループ、RX651 グループ ユーザーズマニュアル ハードウェア編 (R01UH0590)

(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)

ユーザーズマニュアル：Firmware Integration Technology

Firmware Integration Technology ユーザーズマニュアル(R01AN1833)

ボードサポートパッケージモジュール Firmware Integration Technology (R01AN1685)

e2 studio に組み込む方法 Firmware Integration Technology (R01AN1723)

(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 CC-RX コンパイラユーザーズマニュアル (R20UT3248)

Renesas e2 studio スマート・コンフィグレータ ユーザーガイド(R20AN0451)

(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)

ユーザーズマニュアル：Renesas Starter Kit

Renesas Starter Kit for RX231 ユーザーズマニュアル (R20UT3027)

Renesas Starter Kit for RX231 CPU ボード回路図 (R20UT3026)

(最新版をルネサス エレクトロニクスホームページから入手してください)

アプリケーションノート

RX210、RX21A、RX220 グループ 消費電力低減機能を使用した各低消費電力モードへの移行例 (R01AN1482)

RX63N グループ、RX631 グループ 消費電力低減機能を使用した各低消費電力モードへの移行例 (R01AN1920)

RX140 グループ スヌーズモードの使用例(R01AN5914)

(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)

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Jun. 22, 2018	-	初版発行
1.10	Sep. 14, 2018	-	対象デバイス追加 (RX65N、RX651 グループ)
		3	サンプルプログラムの条件に RX65N を追加
		4	動作確認条件に RX65N を追加
		9	全モジュールクロックストップモードの章を追加
		10	ディープソフトウェアスタンバイモードの章を追加
		19	RX65N 用のサンプルプログラムの追加に伴う更新
		20	サブクロックの停止設定の章を削除
		プログラム	save_power_disable_subclk 関数を削除 Rev.1.10 のプロジェクトに含まれる BSP ではサブクロックの初期状態が発振停止に変更された。このため、本関数を削除
		プログラム	USE_RTC_OPERATION マクロを削除 見直しにより当該マクロを削除
		1.20	Jun. 06, 2022
-	RX230 用のサンプルプログラム削除		
3	各デバイスがサポートしているモードの一覧表を追加		
3	サンプルプログラムの条件に RX140 を追加		
4	動作確認条件を更新		
6	使用端子一覧に RX140 を追加		
12	コンポーネントのバージョン更新		
13	プロジェクト構成一覧に RX140 を追加		
14	プロジェクト構成一覧に RX65N 用の新規プロジェクト 1 件追加		
17	サンプルプログラムで使用する定数に RX140 分を追加		
17	サンプルプログラムで使用する定数に「ALARMSET」を追加		
18	サンプルプログラムで使用する関数に「alarm_setting」、「convert_to_dec」を追加		
19	関数仕様に「alarm_setting」、「convert_to_dec」を追加		
22	main 関数のフローチャート更新		
26	alarm_setting 関数のフローチャート追加		
27	convert_to_dec 関数のフローチャート追加		
28	「低消費電力モードへ移行する際のノウハウ」項目追加		
28	「トラブルシューティング」項目削除		
30	動作確認周波数一覧に RX140 を追加		

## 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. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

### 2. 電源投入時の処置

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. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

### 4. 未使用端子の処理

未使用端子は、「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。

### 5. クロックについて

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、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。

### 6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 $V_{IL}$  (Max.) から  $V_{IH}$  (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 $V_{IL}$  (Max.) から  $V_{IH}$  (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

### 7. リザーブアドレス（予約領域）のアクセス禁止

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、将来の拡張機能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。

### 8. 製品間の相違について

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、フラッシュメモリ、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。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、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
  2. 当社製品または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またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
  3. 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. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
  5. 当社製品を、全部または一部を問わず、改造、変更、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、変更、複製、リバースエンジニアリング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。
  6. 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）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等  
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、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（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
  7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
  8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他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。
  10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。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。
  11. 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。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
  12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
  13. 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。
  14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 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が直接的、間接的に支配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。

(Rev.5.0-1 2020.10)

## 本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレストシア）

[www.renesas.com](http://www.renesas.com)

## お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

[www.renesas.com/contact/](http://www.renesas.com/contact/)

## 商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。