

RA4L1 グループ

スマートウェイクアップソリューション

要旨

本アプリケーションノートは、RA4L1 静電容量タッチ評価システムで低消費電力動作するソフトウェアについて説明します。

動作確認デバイス

RA4L1 (R7FA4L1BD4CFP)

関連ドキュメント

1. RA4L1 グループ 静電容量タッチ評価システムユーザーズマニュアル(R12UZ0162JJ0110)

目次

1. 概要	3
2. 動作確認済環境	3
3. 機能	4
4. 静電容量タッチ設定	5
4.1 タッチインターフェース構成	5
4.2 構成(メソッド)の設定	5
5. ソフトウェア仕様	6
5.1 ソフトウェア構造	6
5.2 ファイル構成	7
5.3 定数一覧	8
5.4 関数一覧	9
5.5 全体処理	10
5.6 初期設定処理	12
5.7 スタンバイ準備	13
5.7.1 ベースライン平均回数	14
5.7.2 ベースライン設定の役割	14
5.8 スタンバイ	15
5.8.1 ソフトウェアスタンバイへ遷移	16
5.8.2 スヌーズモードへ遷移	16
5.8.3 スヌーズモードからの分岐	16
5.9 アクティブ	17
5.9.1 アクティブ→スタンバイ準備	18
5.9.2 アクティブ動作	18
6. チューニング結果	19
7. 消費電力測定	20
7.1 スタンバイの動作条件	20
7.2 計測機器、ソフトウェア	20
7.3 RA4L1 CPU ボード	21
7.4 RA4L1 CPU ボード・ジャンパ設定	21
7.5 消費電流計測環境	22
7.6 消費電流計測設定	22
7.7 消費電流計測結果	23
7.8 平均消費電流算出結果	24
改訂記録	25

1. 概要

本アプリケーションノートは、RA4L1 に搭載されている CTSU2SLa の自動判定機能および複数電極接続機能 (MEC : Multiple Electrode Connection) を使用した低消費電力でのタッチ動作をするソフトウェアについて説明します。

本ソフトウェアは「スタンバイ」と「アクティブ」の 2 つのステートを持ちます。スタンバイで任意のボタンでのタッチ検出をしたらアクティブに移行します。アクティブで非タッチが続いたらスタンバイに移行します。この動作をループするシステムです。

本アプリケーションノートでは、このシステムを「スマートウェイクアップソリューション」と呼びます。

2. 動作確認済環境

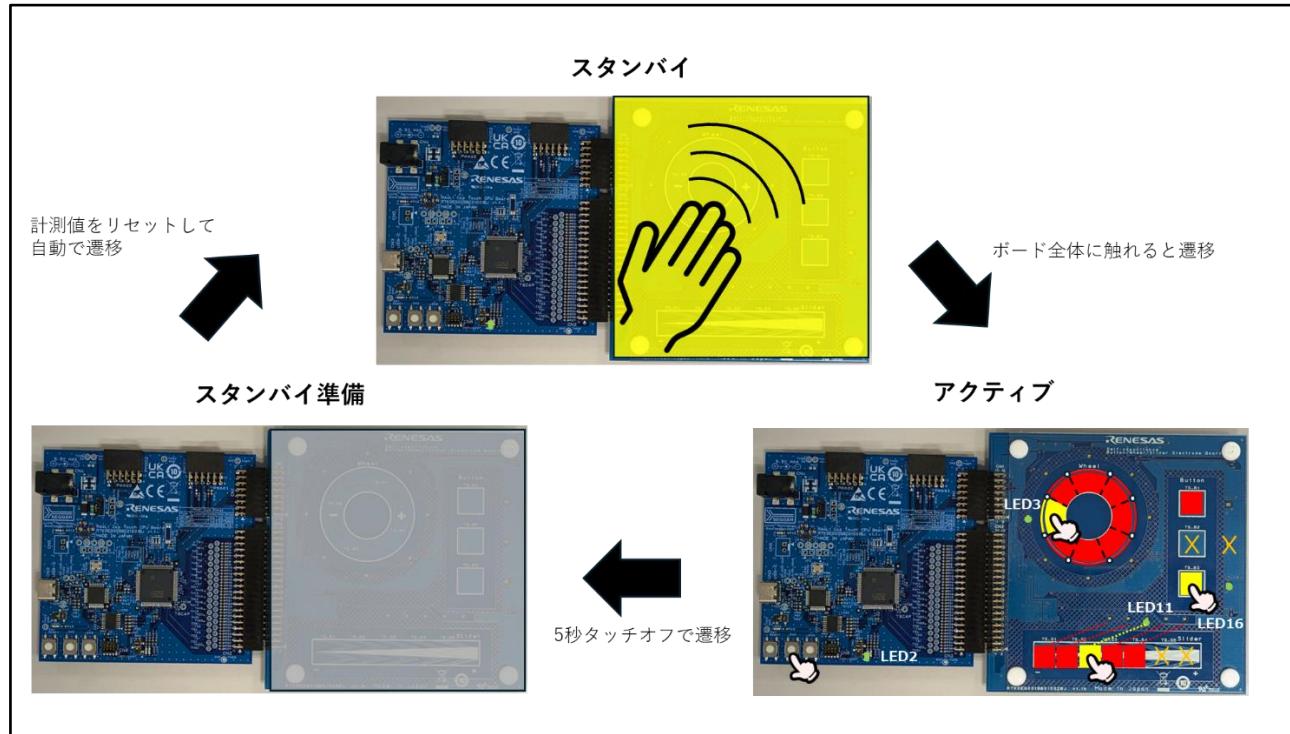
本ソフトウェアの動作確認済環境を表 2.1 に示します。

表 2.1 動作確認済環境

項目	内容
MCU (e2 studio プロジェクト共通部分)	RA4L1 搭載静電容量タッチ評価システム (製品型名 : RTK0EG0057S01001BJ) • RA4L1 CPU ボード (型名 : RTK0EG0056C01001BJ) • 静電容量タッチ評価用アプリケーションボード – Self-Capacitance Buttons / Wheels / Slider Board (型名 : RTK0EG0019B01002BJ v1.1b)
使用マイコン	R7FA4L1BD4CFP (RA4L1 MCU グループ)
動作周波数	24MHz
動作電圧	5V
統合開発環境	e ² Studio 2025-04
C コンパイラ	GCC 13.2.2.arm-13-7
OCD エミュレータ	E2 エミュレータ Lite
QE for Capacitive Touch	V4.1.0

3. 機能

スマートウェイクアップソリューションは、低消費電力状態からの復帰を最適化するために設計されています。本設計では、タッチセンサを用いた入力をトリガとして、アクティブ動作状態への移行を行います。スタンバイの消費電力を最小限に抑えつつ、必要なタイミングで迅速に応答できる設計が特徴です。動作の概要を図 3.1 に示します。



4. 静電容量タッチ設定

本ソフトウェアのタッチインターフェース構成、構成(メソッド)の設定、QE のチューニング機能を使用したチューニング結果を示します。

4.1 タッチインターフェース構成

図 4.1 にタッチインターフェース構成を示します。

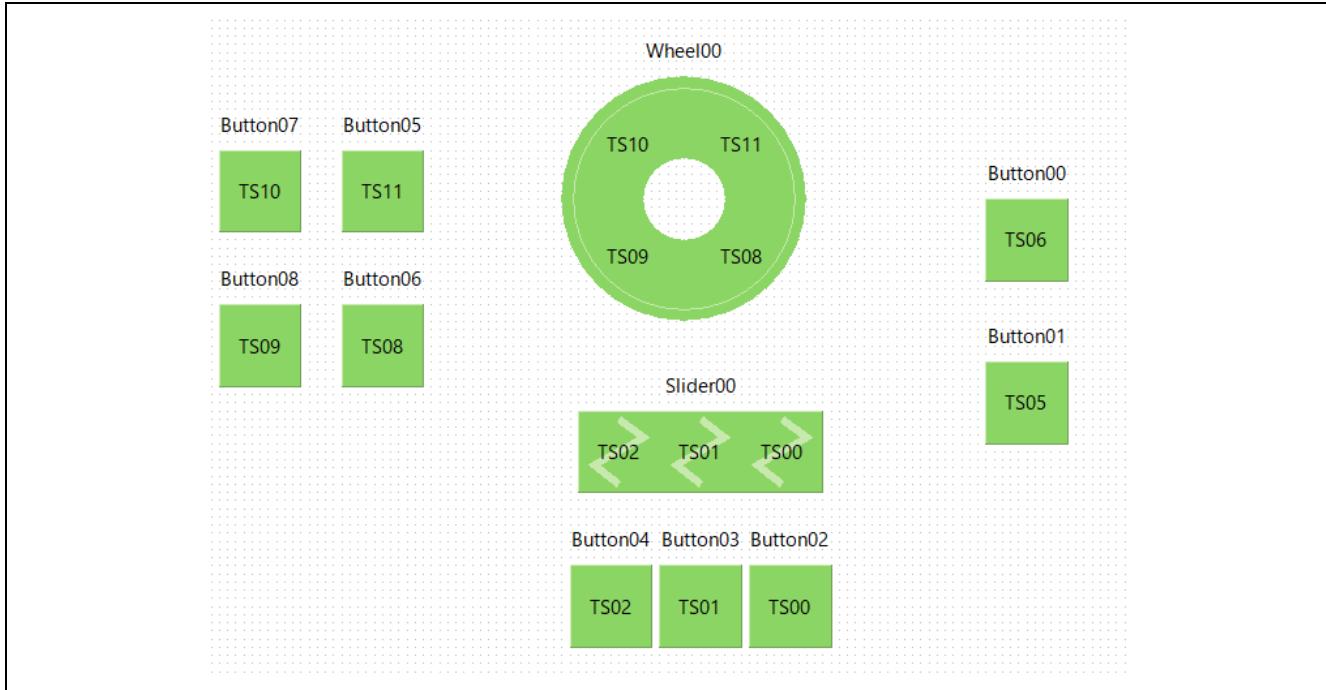


図 4.1 タッチインターフェース構成画面

4.2 構成(メソッド)の設定

図 4.2 にタッチインターフェース設定を示します。各設定の説明を以下に示します。

mec 構成・・・すべてのボタンを設定し、スタンバイの計測時（図 3.1 参照）に利用します。スタンバイ時の計測のため自動判定機能を有効、低消費電力計測のため複数電極接続を有効に設定します。

bsw 構成・・・2つのボタン、スライダ、ホイールを設定します。アクティブの計測時（図 3.1 参照）に利用します。タッチ自動判定後に、スライダやホイールを利用するユースケースを想定して、タッチインターフェース構成を設定します。

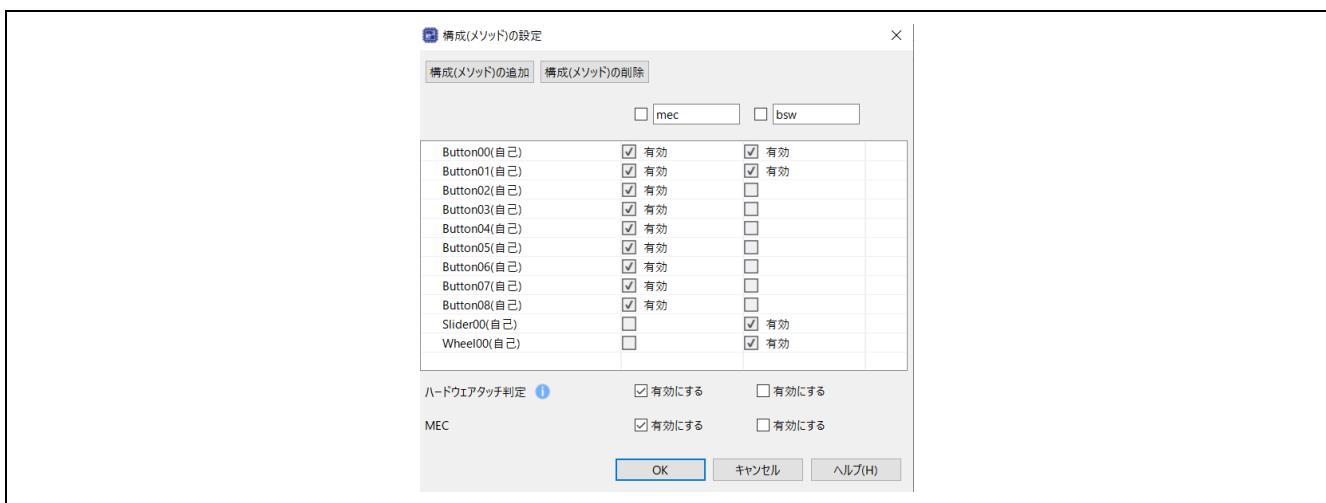


図 4.2 構成(メソッド)の設定画面

5. ソフトウェア仕様

5.1 ソフトウェア構造

静電容量式タッチセンサ開発支援ツール QE for Capacitive Touch と FSP を使用して下記モジュールを追加してアプリケーションを作成しています。ソフトウェア構造図を下記に示します。

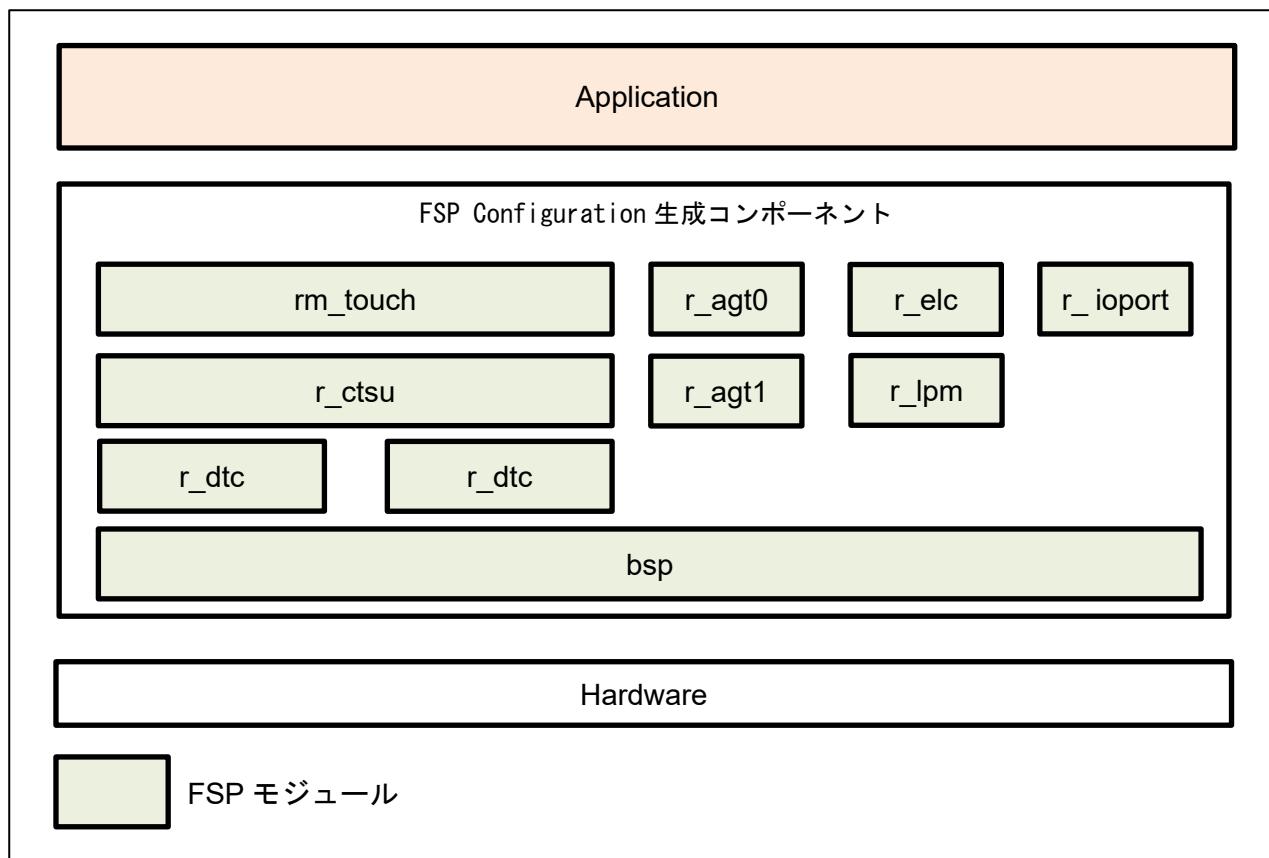


図 5.1 ソフトウェア構造図

表 5.1 にコンポーネントを示します。コンポーネントの設定は、スマート・コンフィグレータを参照してください。

表 5.1 コンポーネント一覧

Selected software components	
Board Support Package Common Files	v5.9.0
Asynchronous General Purpose Timer	v5.9.0
Capacitive Touch Sensing Unit	v5.9.0
Data Transfer Controller	v5.9.0
Event Link Controller	v5.9.0
I/O Port	v5.9.0
Low Power Modes	v5.9.0
Touch	v5.9.0
Arm CMSIS Version 6 - Core (M)	v6.1.0+fsp.5.9.0
RSSK-RA4L1 Board Support Files	v5.9.0
Board support package for R7FA4L1BD4CFP	v5.9.0
Board support package for RA4L1	v5.9.0
Board support package for RA4L1 - FSP Data	v5.9.0
Board support package for RA4L1 - Events	v5.9.0

5.2 ファイル構成

図 5.2 にソースファイルツリーを示します。

スマート・コンフィグレータのファイルは省略します。

```

└── qe_gen
    ├── qe_touch_config.c  ······ Touch QE 構成定義ソースファイル
    ├── qe_touch_config.h  ······ Touch QE 構成定義ヘッダファイル
    ├── qe_touch_define.h  ······ Touch QE 構成定義ヘッダファイル
    └── qe_touch_sample.c  ······ Touch QE アプリケーションファイル

└── src
    ├── hal_entry.c  ······ main ファイル
    ├── r_board_control.c  ······ ボード制御ソースファイル
    ├── r_board_control.h  ······ ボード制御ヘッダファイル
    ├── r_rssk_touch_led.c  ······ タッチ電極 LED 処理用ソース
    └── r_rssk_touch_led.h  ······ タッチ電極 LED 処理用ヘッダ

└── QE-Touch
    ├── smart_wakeup_ra4l1_rssk.tifcfg  ··· タッチ I/F 構成ファイル
    └── smart_wakeup_ra4l1_rssk_log_tuning20250416210442.log  ··· QE チューニングログ

```

図 5.2 ソースファイルツリー

表 5.2 にソースファイルを示します。

表 5.2 ソースファイル

ファイル名	内容
hal_entry.c	メインソースファイル
r_board_control.c	ボード制御ソースファイル
qe_touch_config.c	Touch QE 構成定義ソースファイル
qe_touch_sample.c	Touch QE アプリケーションファイル

表 5.3 にヘッダファイルを示します。

表 5.3 ヘッダファイル

ファイル名	内容
r_board_control.h	ボード制御ヘッダファイル
qe_touch_config.h	Touch QE 構成定義ヘッダファイル
qe_touch_define.h	Touch QE 構成定義ヘッダファイル
r_board_control.h	ボード制御ヘッダファイル
r_rssk_touch_led.h	タッチ電極 LED 処理用ヘッダ

5.3 定数一覧

表 5.4 に定数一覧を示します。

表 5.4 定数一覧

定数名	設定値	内容
ファイル名 : <code>qe_touch_sample.c</code>		
BASELINE_SET_TIME	(33)	自動判定に必要なベースライン設定時間(1msec)を Hz で指定
SOFTWARE_MODE_TIME	(3326)	スタンバイ中 AGT アンダーフロー値 (100msec)を Hz で指定
NORMAL_MODE_TIME	(655)	アクティブ中 AGT アンダーフロー値 (20msec)を Hz で指定
AJINTC_BIT_INDEX	(9U)	CTSUOP の AJINTC ビット
LED_ROW0	BSP_IO_PORT_05_PIN_13	COL0 制御ポート定義
LED_ROW1	BSP_IO_PORT_00_PIN_04	COL1 制御ポート定義
LED_ROW2	BSP_IO_PORT_00_PIN_03	COL2 制御ポート定義
LED_ROW3	BSP_IO_PORT_00_PIN_02	COL3 制御ポート定義
LED_COL0	BSP_IO_PORT_04_PIN_08	ROW0 制御ポート定義
LED_COL1	BSP_IO_PORT_04_PIN_07	ROW1 制御ポート定義
LED_COL2	BSP_IO_PORT_04_PIN_10	ROW2 制御ポート定義
LED_COL3	BSP_IO_PORT_04_PIN_09	ROW3 制御ポート定義
BUTTON_OFF_STATE	(0x0000)	ボタン非タッチ時の値
SLIDER_OFF_STATE	(0xFFFF)	スライダ非タッチ時の値
WHEEL_OFF_STATE	(0xFFFF)	ホイール非タッチ時の値
TOUCH_OFF_PERIOD	(250U)	非タッチ判定カウント (20ms * 250 = 5sec)
WAKEUP_TIME_BASELINE	(32)	スタンバイ準備の計測回数
ファイル名 : <code>r_rssk_touch_led.c</code>		
LED_COL0	(BSP_IO_PORT_05_PIN_13)	COL0 制御ポート定義
LED_COL1	(BSP_IO_PORT_00_PIN_04)	COL1 制御ポート定義
LED_COL2	(BSP_IO_PORT_00_PIN_03)	COL2 制御ポート定義
LED_COL3	(BSP_IO_PORT_00_PIN_02)	COL3 制御ポート定義
LED_ROW0	(BSP_IO_PORT_04_PIN_08)	ROW0 制御ポート定義
LED_ROW1	(BSP_IO_PORT_04_PIN_07)	ROW1 制御ポート定義
LED_ROW2	(BSP_IO_PORT_04_PIN_10)	ROW2 制御ポート定義
LED_ROW3	(BSP_IO_PORT_04_PIN_09)	ROW3 制御ポート定義
LED_COL_MAX	(4)	COL 信号数
LED_ROW_MAX	(4)	ROW 信号数
LED_COL_OFF	(BSP_IO_LEVEL_LOW)	COL 信号 OFF
LED_COL_ON	(BSP_IO_LEVEL_HIGH)	COL 信号 ON
LED_ROW_OFF	(BSP_IO_LEVEL_HIGH)	ROW 信号 OFF
LED_ROW_ON	(BSP_IO_LEVEL_LOW)	ROW 信号 ON
SLIDER_LED_NUM	(5U)	スライダ LED 数
SLIDER_RESOLUTION	(100)	スライダタッチ結果最大値
WHEEL_LED_NUM	(8U)	ホイール LED 数
WHEEL_LED_MSB	(1U << (WHEEL_LED_NUM - 1))	ホイール制御ビット MSB

WHEEL_RESOLUTION_DEGREE	(360)	ホイールタッチ結果 最大値[単位：度]
WHEEL_POSITION_OFFSET_DEGREE	(112)	ホイールタッチ位置 オフセット[単位：度]
ALL_LED_NUM	(16U)	タッチボード LED 数
LED_TEST_INTERVAL	(100U)	LED 点灯インターバル 時間
DUMMY_BUTTON02	(2)	LED 15 点灯用のダミー 判定ボタン
ファイル名 : r_board_control.c		
LED1	BSP_IO_PORT_06_PIN_01	CPU LED2 制御ポート定義

5.4 関数一覧

表 5.5 に関数の一覧を示します。

表 5.5 関数一覧

関数名	処理概要
ファイル名 : qe_touch_sample.c	
qe_touch_main	Main function
init_peripheral_function	周辺機能の初期化
agt_led_control_callback	LED 制御用のコールバック関数
r_rssk_touch_led_control	LED 制御関数
ファイル名 : r_rssk_touch_led.c	
r_rssk_touch_led_test	タッチボード LED テスト処理
r_rssk_touch_led_control	タッチボード LED 制御処理
ファイル名 : r_board_control.c	
r_rssk_touch_led_control	タッチボード LED 制御処理

5.5 全体処理

図 5.3 に全体処理のフローチャートを示します。

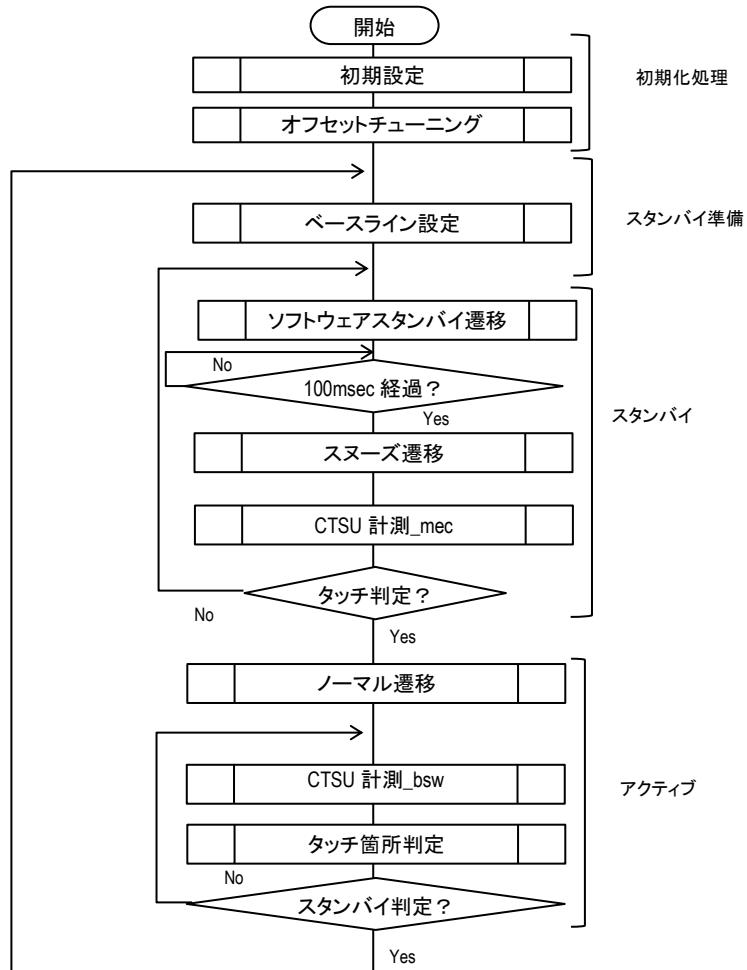


図 5.3 全体フロー

※LED 制御は、割り込み関数で制御しています。詳しくは、5.9 アクティブをご確認ください。

以下にスタンバイからアクティブになり、再度スタンバイになる流れを示します。この流れは図 3.1 の動作説明を基に、二つの構成(mec, bsw)のステートに着目して説明しています。

状態	スタンバイ 準備	計測値をリセットして自動で遷移	スタンバイ	ボード全体に触れると遷移	アクティブ	5秒タッチOFFで遷移	スタンバイ 準備	スタンバイ 開始	スタンバイ	...
mec	計測中	ベースライン 平均回数分 の計測完了	計測中	タッチON	計測停止	計測停止	計測中	ベースライン 平均回数分 の計測完了	計測中	...
bsw	計測停止	計測停止	計測停止	計測停止	計測中	5秒タッチOFF	計測停止	計測停止	計測停止	...

図 5.4 状態遷移イメージ

※各ステートの詳細は、下記の章をご参照ください。

- ・ 5.7 スタンバイ準備
- ・ 5.8 スタンバイ
- ・ 5.9 アクティブ

スマートウェイクアップソリューションは、スタンバイとアクティブをループします。各ステート中に環境変化などで寄生容量が大きく変わってしまうと、動作させていないタッチインターフェース構成ではベースライン更新処理が動作できません。従って、非タッチでしきい値を超える恐れがありタッチ判定ができなくなる可能性があります。その場合はシステムリセットが必要となってしまうため、環境変化が大きい場所で使用する場合は注意してください。

5.6 初期設定処理

図 5.5 に初期設定処理のフローチャートを示します。

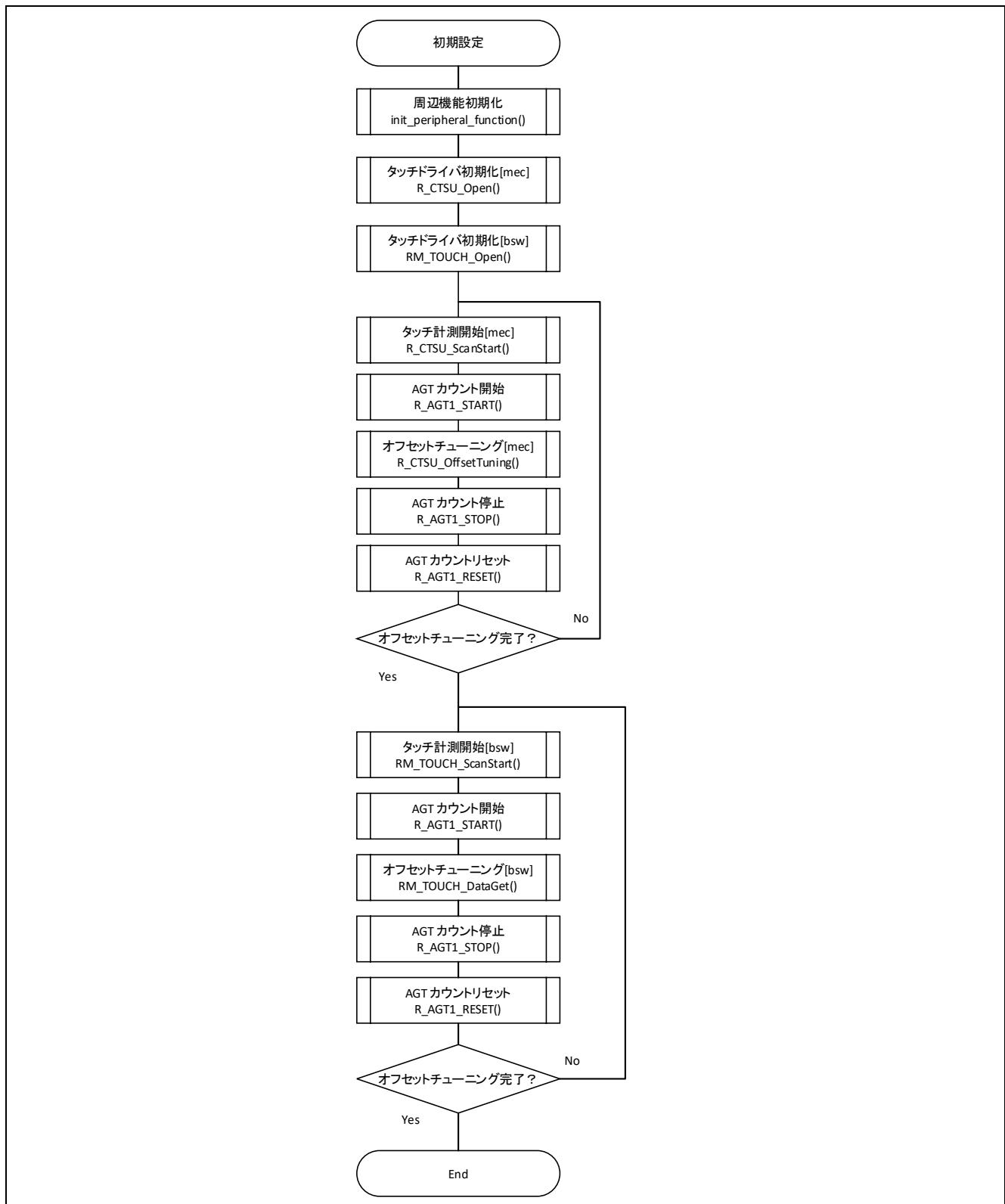


図 5.5 初期設定処理のフローチャート

5.7 スタンバイ準備

アクティブからスタンバイに遷移するときにタッチの誤判定を防ぐためのステートです。

非タッチの状態でスタンバイに遷移するために、この処理ではベースライン平均回数分の計測をして、全て非タッチであればスタンバイに遷移します。スタンバイの詳細は、5.7 をご参照ください。

以下にスタンバイ準備のフローチャートを示します。

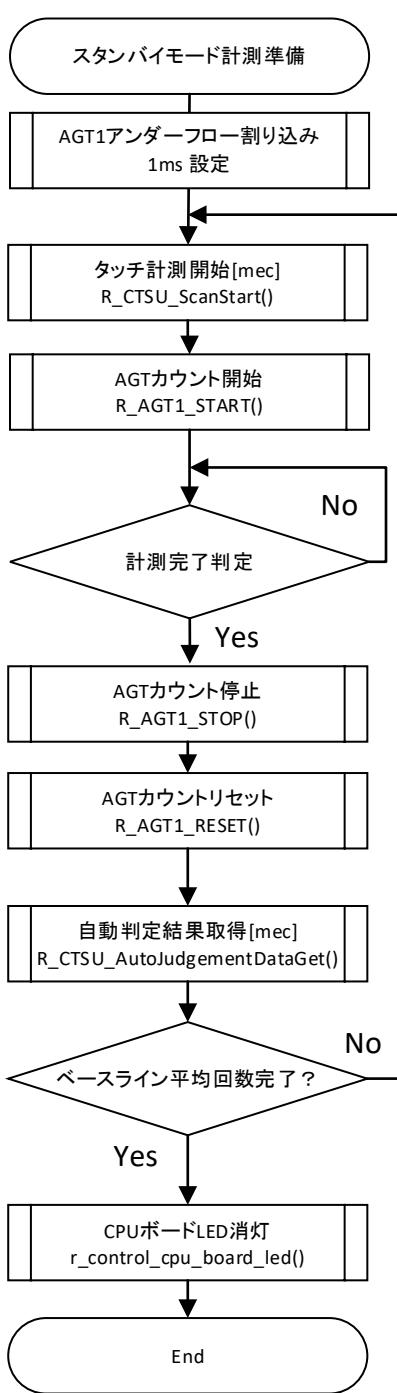


図 5.6 スタンバイ準備のフローチャート

5.7.1 ベースライン平均回数

ベースライン平均回数は、下記の式によって定まっています。

$$\text{ベースライン平均回数} = 2^{\text{ajbmat}+1}$$

今回は ajbmat の設定値が 4 であるため、ベースライン平均回数は 32 です。

5.7.2 ベースライン設定の役割

本アプリケーションでは、タッチ自動判定を行う前に、スタンバイ準備にてベースラインを設定しています。

この遷移を計測値で表すと以下のようになります。

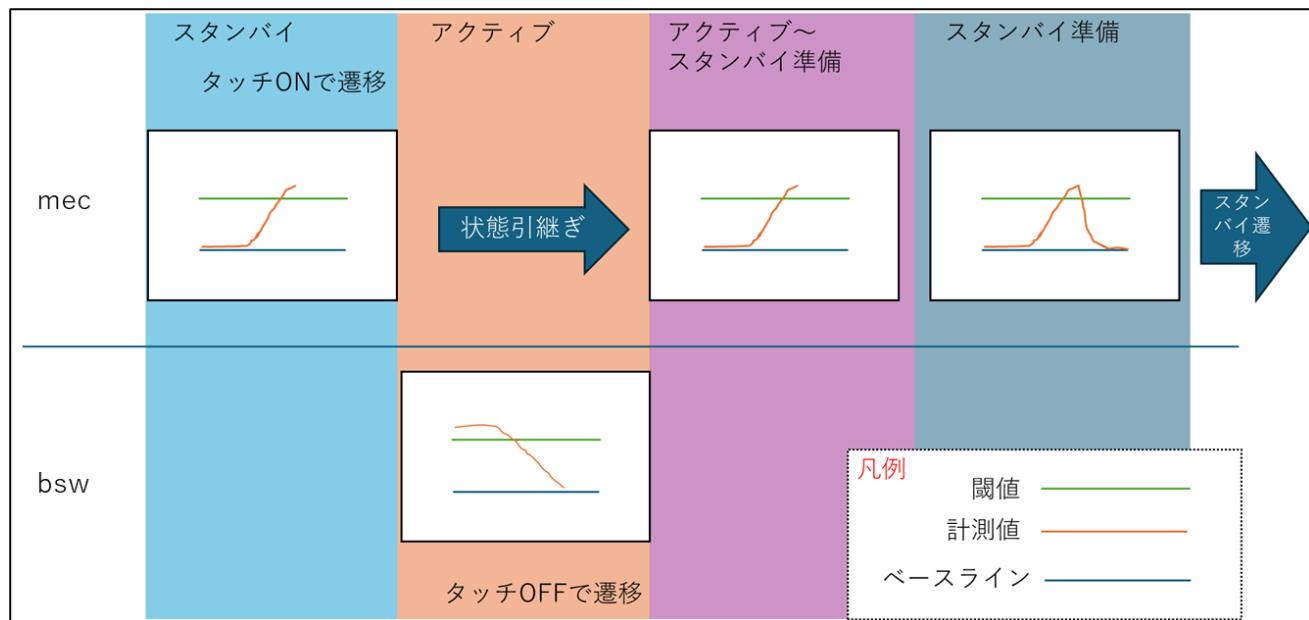


図 5.7 計測値遷移イメージ

この設定には、以下の役割があります。

- タッチ判定誤動作防止のための空回し処理

本処理では、MEC のタッチ判定が OFF になるまで計測を繰り返すことで、ベースライン取得回数の計測をスタンバイ準備のステータスで完了させます。

※ ドライバ仕様に基づき、最初のスタンバイへ遷移する前に、アクティブ状態でベースラインの更新を行う必要があります。本処理は、そのベースライン更新の役割も兼ねています。

5.8 スタンバイ

CPU を低消費電力モードにして、自動判定機能および複数電極接続機能を使用したタッチ計測をします。複数電極接続機能(MEC)を使用することで、9ch を 1 回で計測して消費電力を削減します。

以下に CPU 動作モードと CTSU 動作状態のイメージを示します。

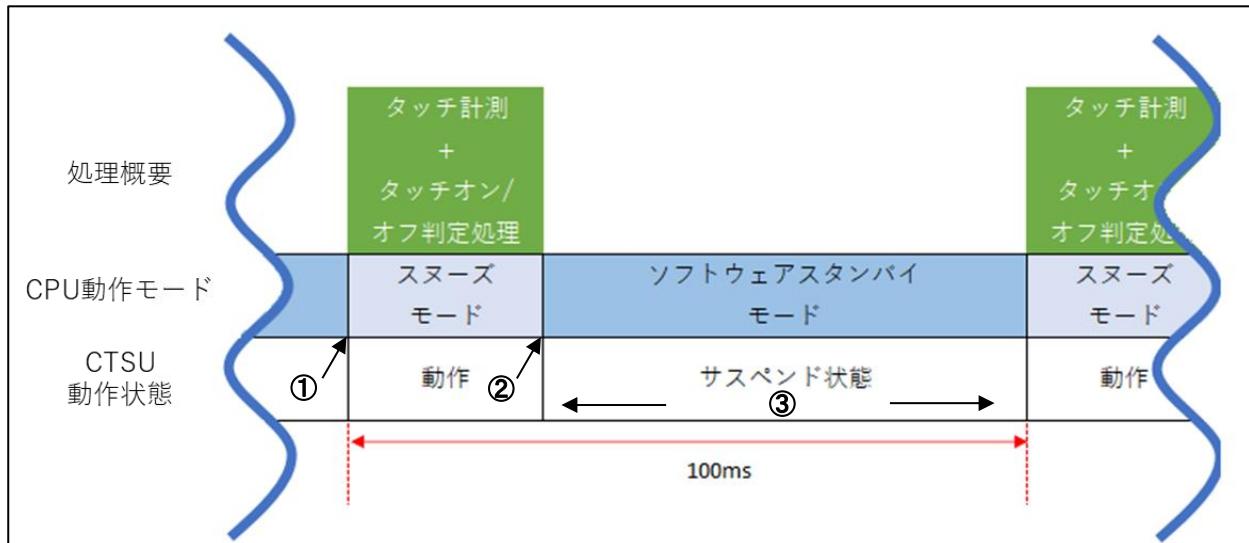


図 5.8 CPU 動作モードと CTSU 動作状態のイメージ

上記の図は、AGT のアンダーフロー割り込みを LPM のスヌーズ要求として設定し、CTSU がそのスヌーズエントリを受け取って計測を開始する流れを示しています。図中の番号と以下の説明が対応しています。

- ① 外部トリガ検出で CPU はスヌーズモードへ遷移します。
- ② スヌーズモードの CTSU 計測は DTC を使用した自動判定機能によりタッチ判定します。タッチ ON 判定が検出されなければ、再度ソフトウェアスタンバイに遷移します。タッチ ON 判定を検出したら、スタンバイモードからアクティブモードに遷移します。
- ③ LPM を使用して CPU をソフトウェアスタンバイモードに遷移します。CTSU はサスペンド状態になります。

以下にスタンバイのフローチャートを示します。

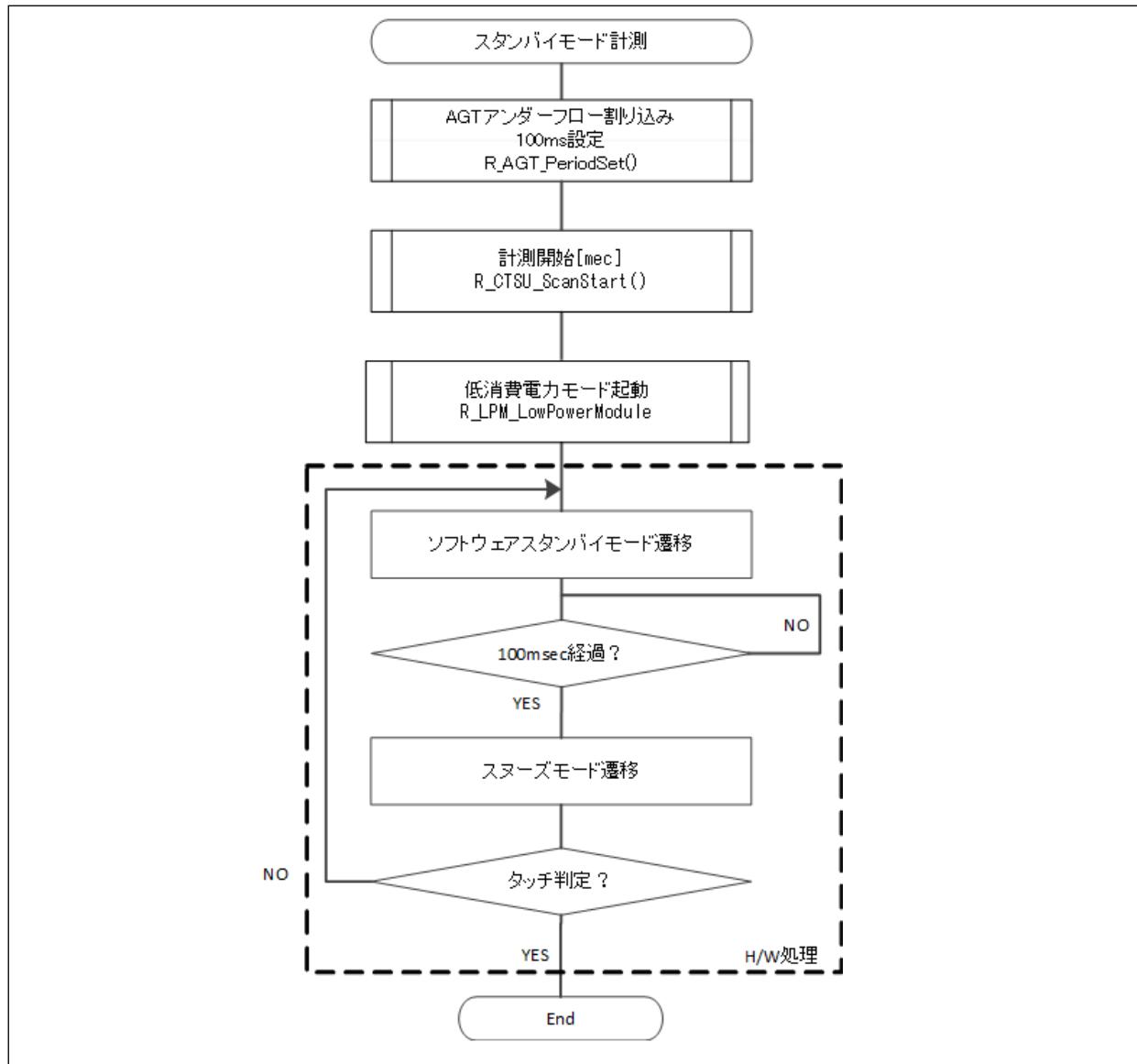


図 5.9 スタンバイのフローチャート

5.8.1 ソフトウェアスタンバイへ遷移

LPM モジュールの API 関数, R_LPM_LowPowerModeEnter を実行しソフトウェアスタンバイモードへ遷移します。

5.8.2 スヌーズモードへ遷移

外部トリガ (AGT アンダーフロー割り込み) をトリガにしてソフトウェアスタンバイモードからスヌーズモードへ遷移します。

5.8.3 スヌーズモードからの分岐

スヌーズモード状態で、自動判定機能+MEC を使用して CTSU 計測を行います。

非タッチ判定の場合：ソフトウェアスタンバイモードへ遷移

タッチ判定の場合：アクティブモードへ遷移

5.9 アクティブ

タッチボードのボタン 2 個、スライダの電極 3 極、ホイールを動作します。

- タッチボードをタッチすると対応した LED を点灯します。非タッチで消灯します。どの LED がタッチに対応するかは、図 3.1 をご参照ください。
- 非タッチ状態で 5 秒経過すると、アクティブからスタンバイ準備へ遷移します。

以下にアクティブのフローチャートを示します。

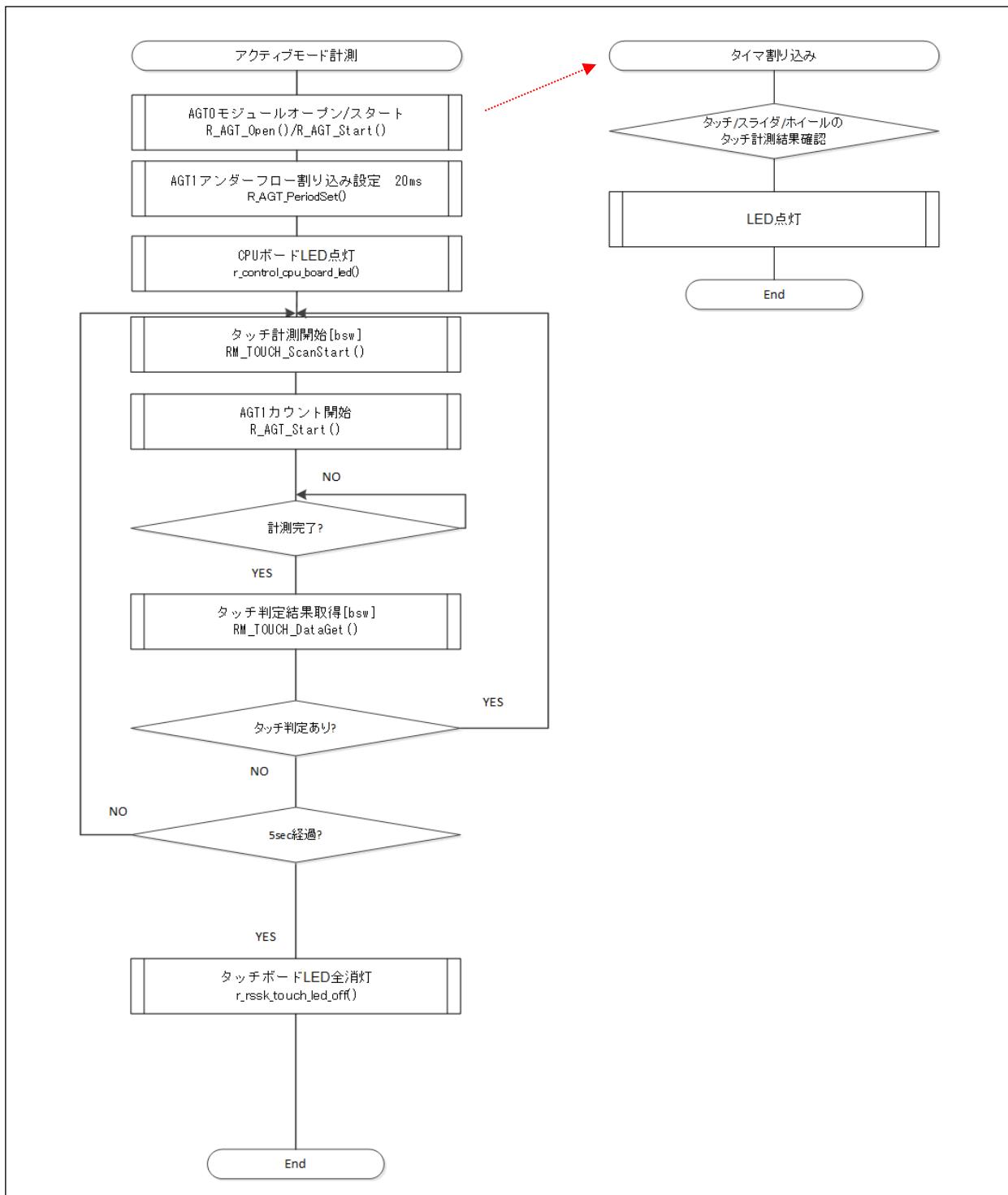


図 5.10 アクティブモードのフローチャート

5.9.1 アクティブ→スタンバイ準備

アクティブで CTSU 計測を行い、スタンバイ判定になればスタンバイ準備へ遷移します。

- ・ CTSU 計測は、周期は AGT タイマの周期=20msec、端子構成は bsw を使用します。
- ・ スタンバイ判定へは、下記の動作条件で遷移します。

非タッチ時 : 非タッチ状態が 5 秒継続

5.9.2 アクティブ動作

アクティブは、下記の動作を行います。

1. bsw で CTSU 計測。ボタン/スライダ/ホイールのタッチ計測。
2. タッチ箇所（ボタン/スライダ/ホイール）に該当する LED をタッチしている間に点灯させる。
※各電極がどの LED に対応しているのかは、図 5.11 をご確認ください。
3. CPU ボードの LED 2 が点灯する。

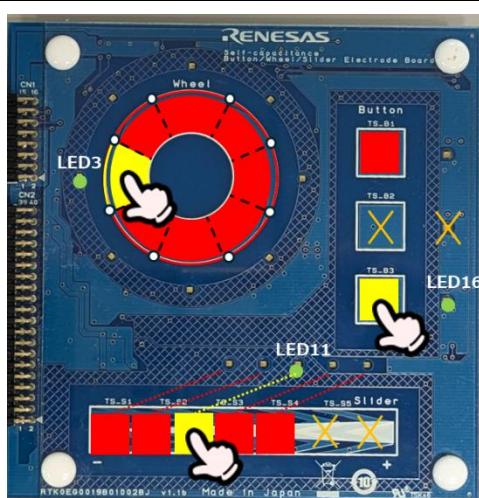


図 5.11 タッチボタン電極の動作範囲例

6. チューニング結果

QE チューニングでのチューニング結果を示します。本ソフトウェアは結果一覧に示される設定値で動作しています。

本ソフトウェアは QE チューニング時に高度な設定で `mec` の `snum` を `0x03` に変更しています。

表 6.1 チューニング結果一覧

メソッド	名前	タッチセンサ	寄生容量 [pF]	ドライブ パルス 周波数[MHz]	閾値	計測 時間 [ms]	so	snum	sdpa
mec	mec00	TS00	167.09	0.5	92;79;104	0.384	0x2D3	0x03	0x1F
bsw	Button00	TS06	20.618	2	1062	0.576	0x11C	0x07	0x07
bsw	Button01	TS05	19.049	2	1036	0.576	0x12C	0x07	0x07
bsw	Slider00	TS02	17.201	2	720	0.576	0x0F6	0x07	0x07
bsw	Slider00	TS01	19.868	2	720	0.576	0x11A	0x07	0x07
bsw	Slider00	TS00	19.111	2	720	0.576	0x13A	0x07	0x07
bsw	Wheel00	TS10	22.757	1	402	0.576	0x09F	0x07	0x0F
bsw	Wheel00	TS11	25.424	1	402	0.576	0x079	0x07	0x0F
bsw	Wheel00	TS08	25.722	1	402	0.576	0x081	0x07	0x0F
bsw	Wheel00	TS09	21.986	1	402	0.576	0x09C	0x07	0x0F

so : センサオフセット設定の変数

snum : 計測期間設定の変数

sdpa : クロック分周設定の変数

注 1 : 結果一覧の値は QE チューニング時の動作環境に依存するため、再度 QE チューニングするとこれらの値が変化する可能性があります。

注 2 : 消費電力を減らすため、`qe_touch_config.c` 内、`g_qe_ctsu_cfg_mec` の設定の一部を手動で変更しています。

`tlot(非タッチ判定基準) = 2 → 1`

`thot(タッチ判定基準) = 2 → 1`

`ajbmat(ベースライン平均回数) = 7 → 4`

7. 消費電力測定

7.1 スタンバイの動作条件

表 7.1 にスタンバイの動作条件を示します。

表 7.1 スタンバイの動作条件

項目	内容
CPU 動作周波数	24MHz 高速オンチップオシレータ (HOCO) 3.2768KHz 低速オンチップオシレータ (LOCO)
システムクロック (ICLK)	6 MHz
周辺クロックモジュール A (PCLKA)	6 MHz
周辺クロックモジュール B (PCLKB)	6 MHz
周辺クロックモジュール C (PCLKC)	6 MHz
周辺クロックモジュール D (PCLKD)	6 MHz
FlashIF クロック (FCLK)	6 MHz
タッチ計測周期	100ms
センサドライブパルス周波数	0.5MHz
CTSU 計測モード	自己容量方式(MD1 = 0)
CTSU スキャンモード	マルチスキャンモード(MD0 = 1)
CTSU 計測動作開始トリガ選択	外部トリガ(CAP = 1)
CTSU 待機時省電力有効	待機時省電力機能有効 (SNZ = 1)
CTSU 電源動作モード	通常電圧動作モード(ATUNE0 = 0)
CTSU 電流範囲調整	40µA (ATUNE1 = 1, ATUNE2 = 0)
CTSU 非計測チャネル出力 (POSEL)	GPIO LOW 出力 (POSEL = 0)
CTSU センサドライブパルス選択(SDPSEL)	高分解パルスモード (SDPSEL = 1)
CTSU センサ安定待ち時間 (SST)	64µs (推奨値) (SST = 0x1F)
CTSU マルチクロック制御	3 周波数 (MCA0,MCA1,MCA2:有効)
CTSU 計測時間	64µs (SNUM= 3)

7.2 計測機器、ソフトウェア

表 7.2 に消費電流を計測したときに使用した機器とソフトウェアを示します。

表 7.2 機器、ソフトウェア一覧

種別	名称	用途
デジタルマルチメータ	KEITHLEY/DMM7510	消費電流を計測
安定化電源	KENWOOD/PA18-1.2A	RA4L1 CPU ボードに電源を供給
ソフトウェア	KEITHLEY/KickStart ソフトウェア	KEITHLEY/DMM7510 から消費電流の計測結果を取得し、ログファイルに出力する

7.3 RA4L1 CPU ボード

RA4L1 CPU ボードの前面を以下に示します。

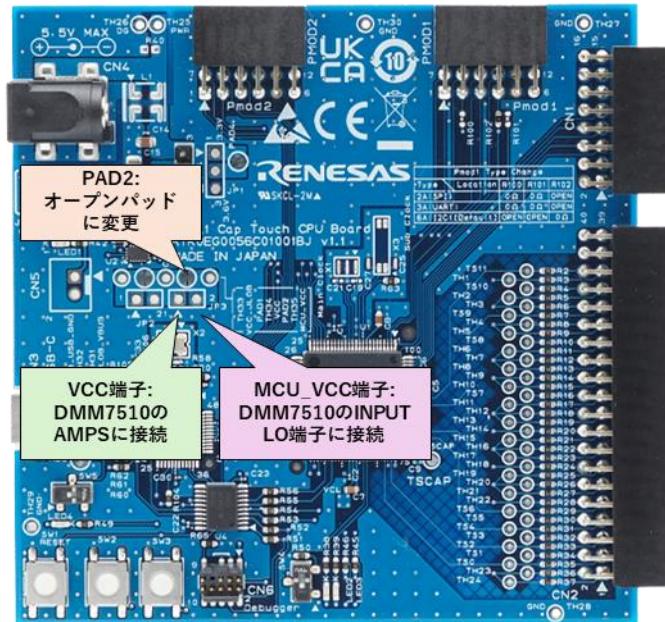


図 7.1 RA4L1 CPU ボード・前面

デフォルトショートの PAD3 について、パッド間のブリッジパターンをカットしてください。ジャンパパッドの形状を以下に示します。

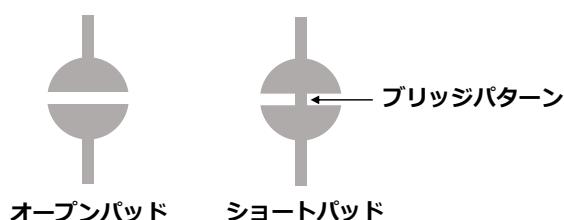


図 7.2 ジャンパパッド形状

7.4 RA4L1 CPU ボード・ジャンパ設定

表 7.3 に消費電流計測向けの RA4L1 CPU ボードのジャンパ設定を示します。

表 7.3 ジャンパ設定

位置	ジャンパ設定	用途
JP3	オープン	消費電流計測
JP4	2-3 ピン クローズ	DC ジャックから電源を供給

他のジャンパ設定、スイッチ設定は出荷時の状態となっています。

7.5 消費電流計測環境

消費電流計測を行った計測環境を以下に示します。

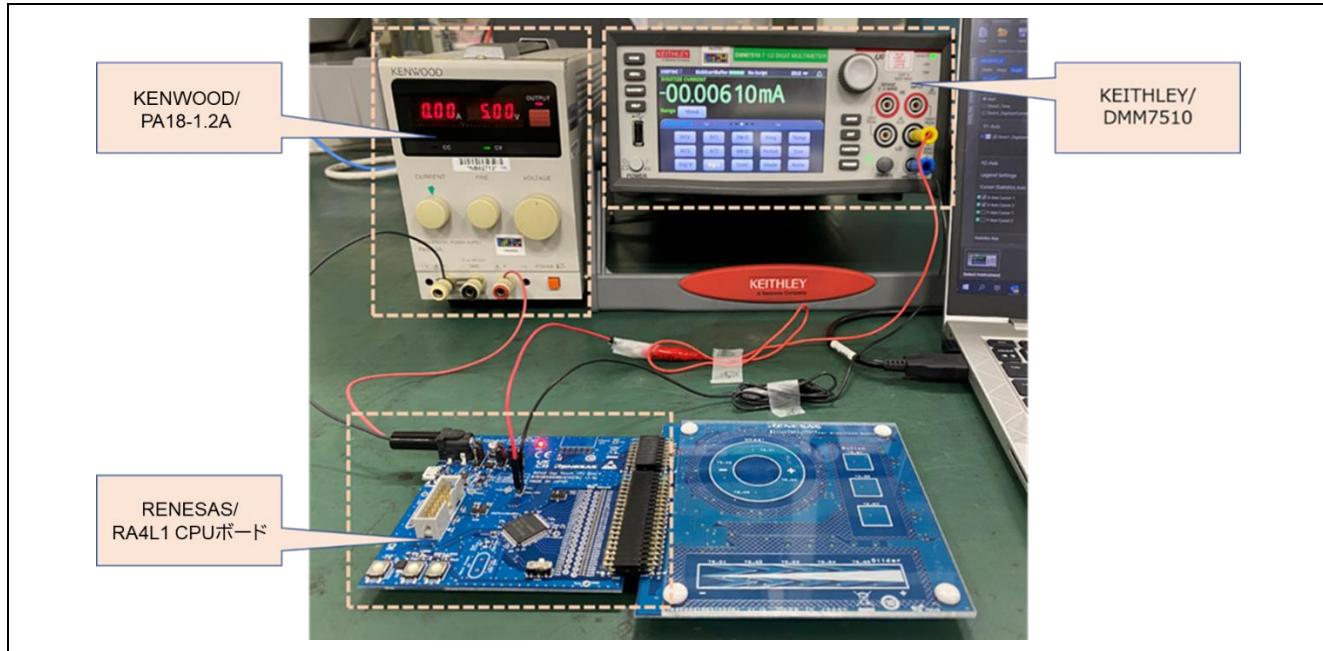


図 7.3 消費電流計測環境

7.6 消費電流計測設定

KEITHLEY/KickStart ソフトウェアの消費電流計測の設定を以下に示します。

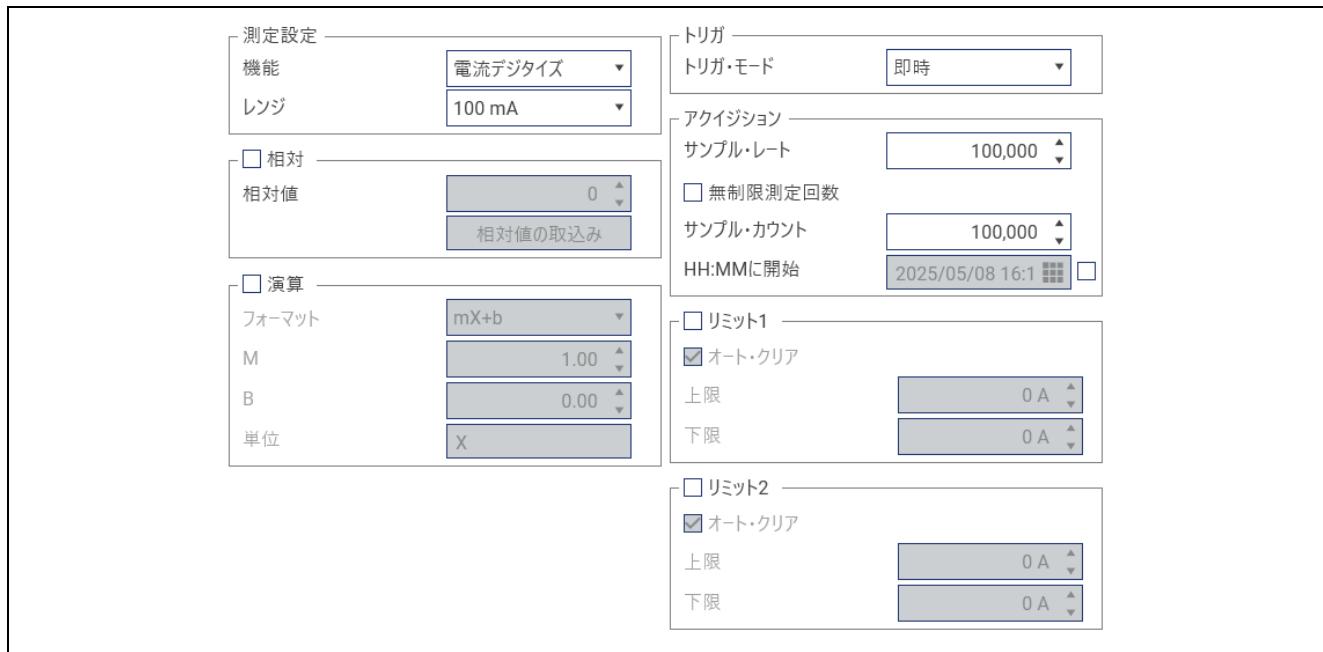


図 7.4 KEITHLEY/KickStart・消費電流計測設定

7.7 消費電流計測結果

図 7.5 および図 7.6 に、CPU 動作モードがソフトウェアスタンバイモード、スヌーズモード（タッチ計測処理、タッチオン/オフ判定処理）に遷移する一連の動作の消費電流波形を示します。

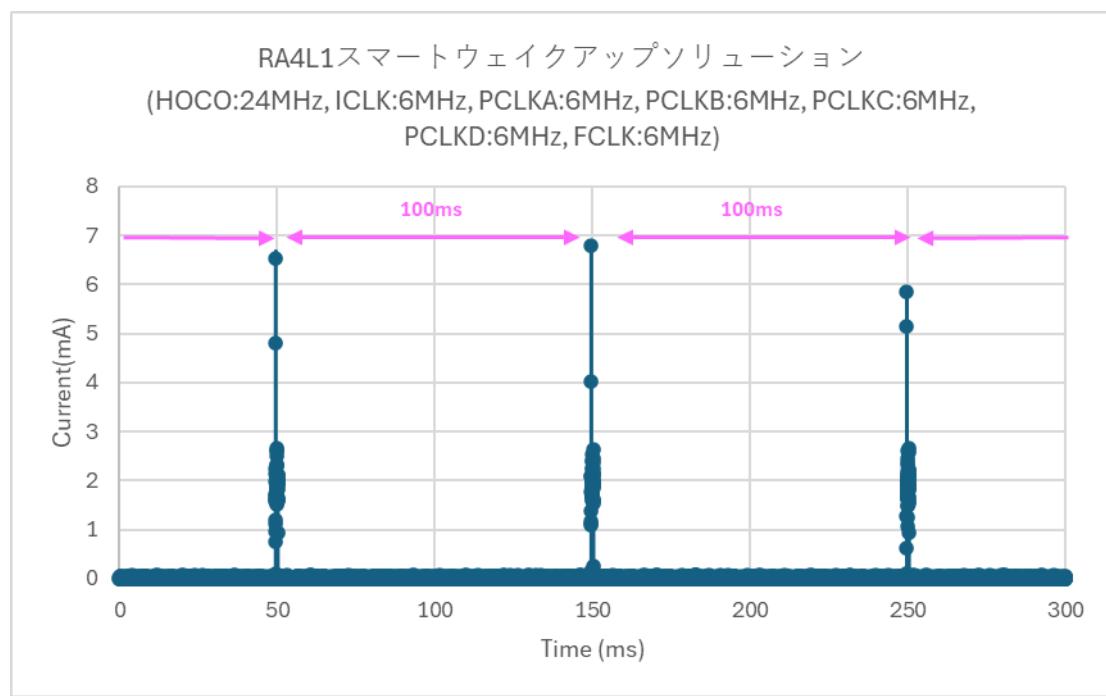


図 7.5 スタンバイモードの消費電流波形 (1/2)

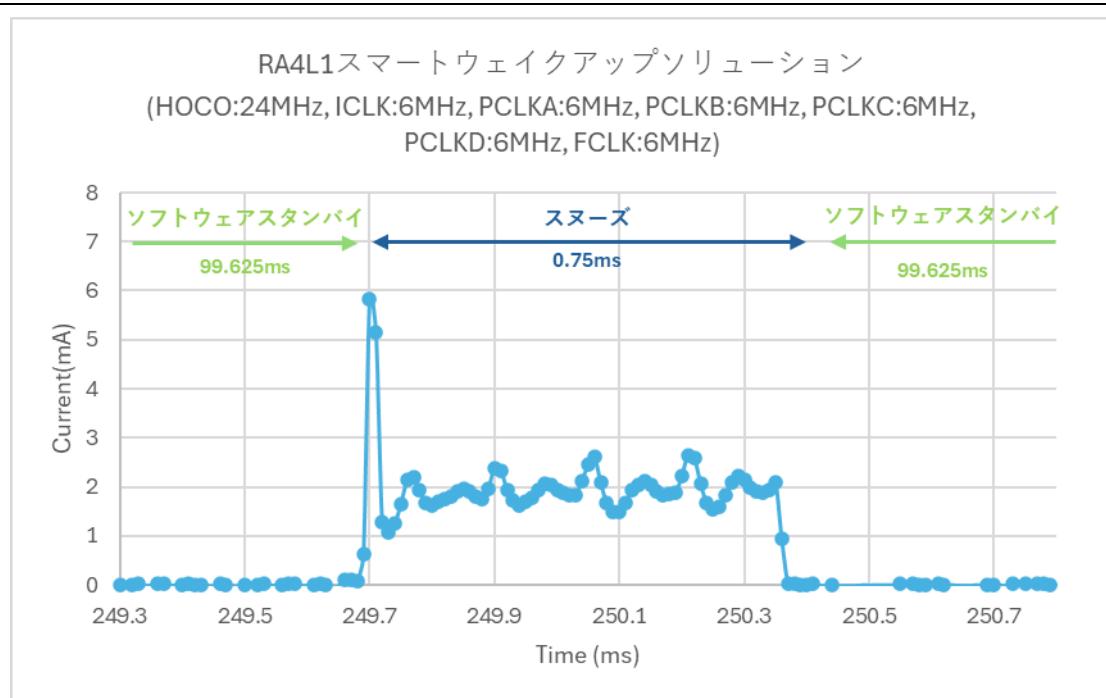


図 7.6 スタンバイモードの消費電流波形 (2/2)

7.8 平均消費電流算出結果

自動判定機能および複数電極接続機能を使用したスタンバイモードでタッチ計測周期 100 ms で測定した平均消費電流を以下に示します。

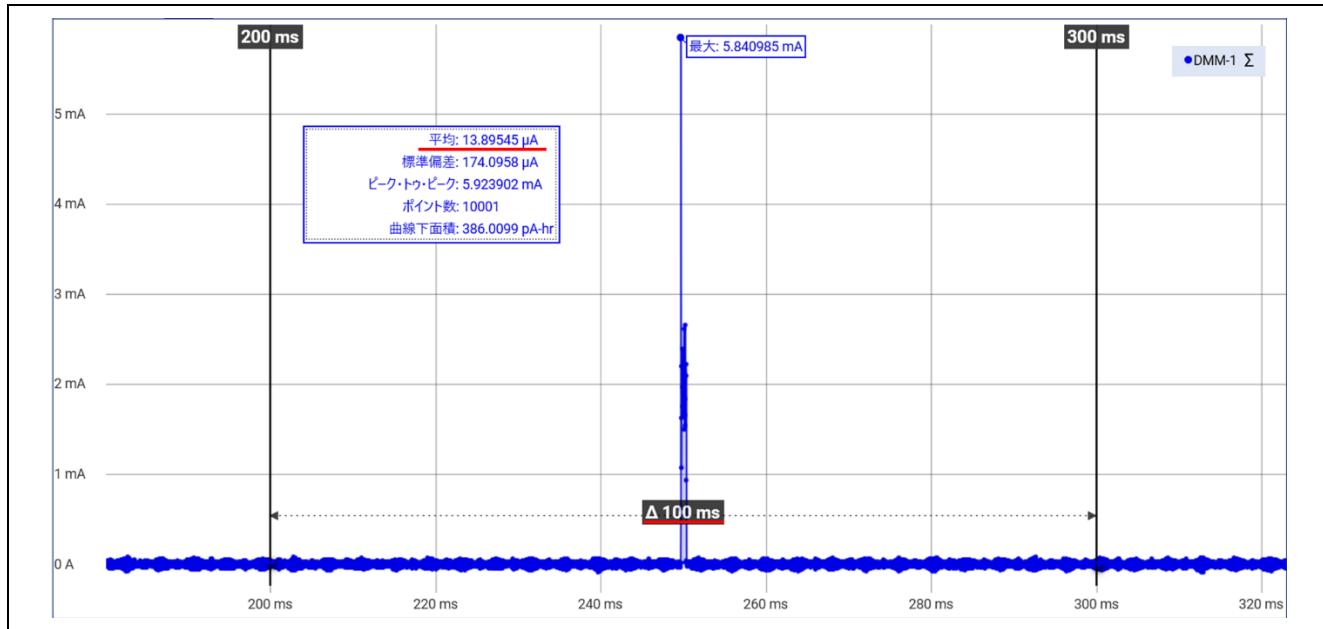


図 7.7 スタンバイモードの消費電流結果

タッチ計測周期 100 ms で測定した平均消費電流 = 13.89545 μ A

※クロック設定を下記の通りに変更した際の、タッチ計測周期 100 ms で測定した平均消費電流は 52.89462 μ A です。

表 7.4 クロックの変更条件

項目	内容
CPU 動作周波数	80MHz 高速オンチップオシレータ (HOCO) 3.2768KHz 低速オンチップオシレータ (LOCO)
システムクロック (ICLK)	80 MHz
周辺クロックモジュール A (PCLKA)	80 MHz
周辺クロックモジュール B (PCLKB)	40 MHz
周辺クロックモジュール C (PCLKC)	40 MHz
周辺クロックモジュール D (PCLKD)	80 MHz
FlashIF クロック (FCLK)	40 MHz
タッチ計測周期	100ms
センサドライブパルス周波数	0.5MHz

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	May.23.25	-	初版発行
1.01	Aug.20.25	-	表 6.1 としおりの誤記改訂

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。

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うと、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、ノイズ幅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、改変、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、改変、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、これに限りません。）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いたします。
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 (豊洲フォレシア)

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/

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。