

RX ファミリ

CAC を使用した HOCO キャリブレーションの方法

要旨

本アプリケーションノートでは、RX ファミリのクロック周波数精度測定回路（以下、CAC）を使用して、高速オンチップオシレータ（以下、HOCO）の周波数を調整する方法について説明します。

動作確認デバイス

- ・RX13T グループ
- ・RX140 グループ
- ・RX231 グループ
- ・RX671 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. 動作確認環境	4
2.1 RX13T	4
2.2 RX140	6
2.3 RX231	8
2.4 RX671	10
3. 使用ソフトウェアモジュール	12
4. ファイル構成	13
5. ソフトウェア説明	14
5.1 動作概要	14
5.2 ファイル構成	19
5.3 定数一覧	19
5.4 変数一覧	20
5.5 関数一覧	20
5.6 関数仕様	21
5.7 フローチャート	22
5.7.1 メイン処理	22
5.7.2 CMT のコンペアマッチ割り込み	23
5.7.3 キャリブレーションの設定処理	24
5.7.4 CAC の測定完了割り込み	25
6. プロジェクトをインポートする方法	27
6.1 e ² studio での手順	27
6.2 CS+での手順	28
7. 付録	29
7.1 MTU の端子出力を行わない場合	29
7.2 ソフトウェアモジュールの設定	29
7.2.1 RX13T	29
7.2.2 RX140	31
7.2.3 RX231	33
7.2.4 RX671	35
8. サンプルコード	37
9. 参考ドキュメント	37
改訂記録	38

1. 仕様

HOCO は、周辺温度など外的要因により、工場出荷時の発振周波数からずれることがあります。周波数の誤差を補正するためにはキャリブレーションが必要です。この HOCO の周波数を調整するために高速オンチップオシレータトリミングレジスタ n（以下、HOCOTRRn レジスタ（n=0~3））があります。一定周期で HOCOTRRn レジスタの値を調整することで、HOCO の周波数の誤差を補正できます。

本アプリケーションノートでは、コンペアマッチタイマによるコンペアマッチ割り込みで HOCO の発振周波数を CAC で測定し、得られた結果により HOCOTRRn レジスタの値を調整します。また、MTIOC0B 端子から HOCO を 4 分周したクロックを出力します。

表 1.1 に使用する周辺機能と用途を、図 1.1 にブロック図を示します。

図中の外部クロックは測定基準クロックであり、キャリブレーションの精度に影響を与えます。従ってできるだけ精度の高い信号発生器を使用してください。例えば測定基準クロックに誤差が一切ない信号発生器を使用した場合、キャリブレーション完了後の HOCO の周波数の誤差は最大 $\pm 0.1\%$ （サンプルコードはデフォルト設定）となります。

信号発生器の誤差が大きいとキャリブレーションの誤差もさらに大きくなりますので、使用状況に応じ誤差が許容できるように信号発生器を選択して下さい。

表 1.1 使用する周辺機能と用途

周辺機能	用途
CAC	CACREF 端子入力を基準に HOCO の周波数を測定
CMT	コンペアマッチ割り込みによる HOCO の測定開始
MTU	HOCO を 4 分周したクロックを出力

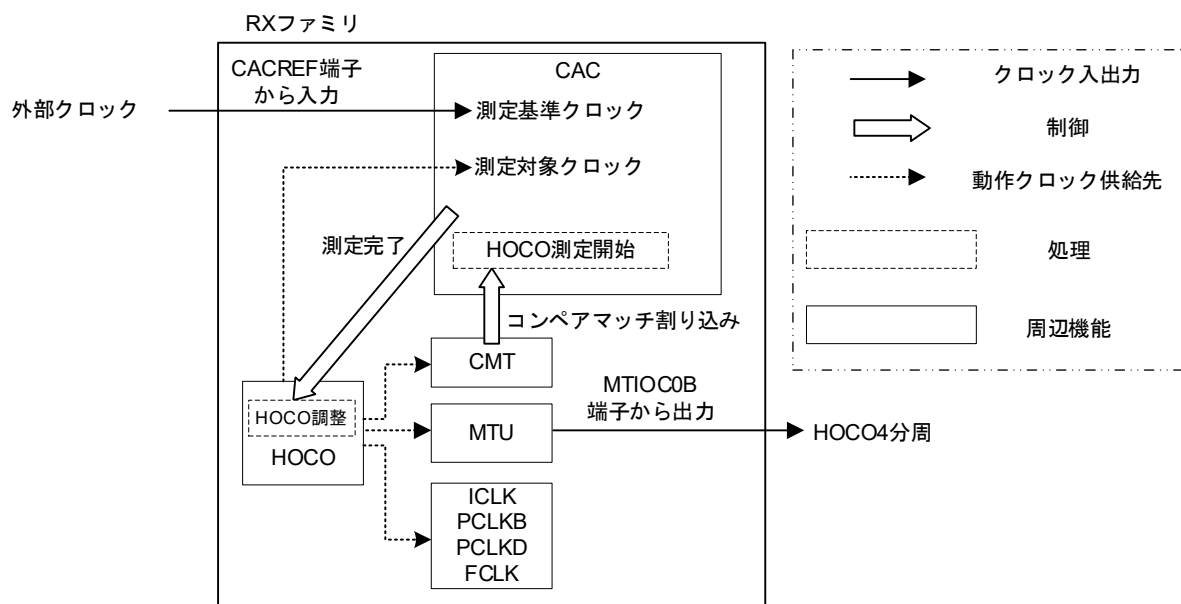


図 1.1 ブロック図

2. 動作確認環境

本アプリケーションノートのサンプルコードは、2.1 から 2.4 の条件で動作を確認しています。

2.1 RX13T

表 2.1 RX13T 動作確認条件

項目	内容
使用マイコン	R5F513T5ADFL (RX13T グループ)
動作周波数	・ HOCO : 32MHz ・ システムクロック (ICLK) : 32MHz (HOCO の 1 分周) ・ 周辺モジュールクロック B (PCLKB) : 32MHz (HOCO の 1 分周) ・ 周辺モジュールクロック D (PCLKD) : 32MHz (HOCO の 1 分周) ・ FlashIF クロック (FCLK) : 32MHz (HOCO の 1 分周)
動作電圧	5.0V
統合開発環境	ルネサスエレクトロニクス製 e ² studio 2022-04
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V.3.04.00 コンパイルオプション 統合開発環境のデフォルト設定
iodefine.h のバージョン	Version 1.00A
エンディアン	リトルエンディアン、ビッグエンディアン
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 1.10
使用ボード	RX13T CPU カード (型名 : RTK0EMXA10CxxxxxBJ)
ファンクションジェネレータ	周波数精度が 2ppm(±0.0002%、18~28℃時)の矩形波を出力できるアナログ信号出力端子を備えた信号発生器。 出力信号は GND に対しバイアス 2.5V、振幅 5.0Vpp に設定した 32Hz の矩形波を出力します。図 2.1 に出力波形を示します。

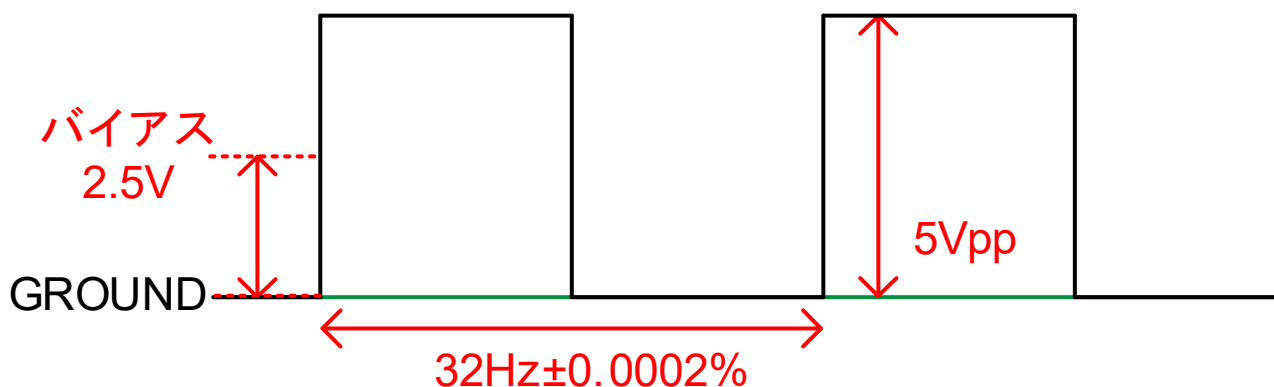
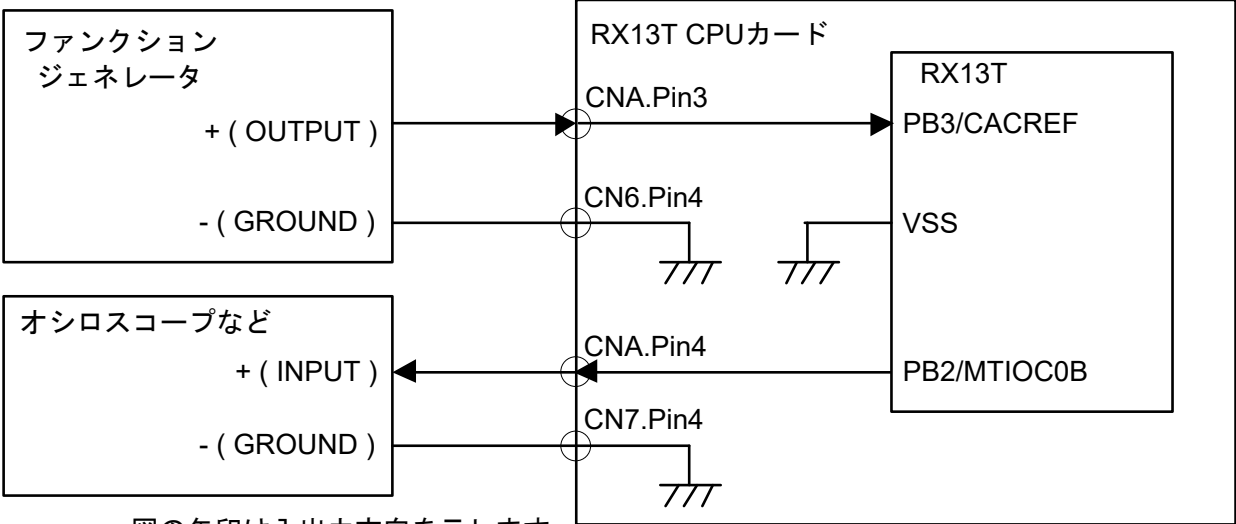


図 2.1 ファンクションジェネレータ 出力波形



図の矢印は入出力方向を示します。

図 2.2 RX13T 端子接続

表 2.2 RX13T 使用端子と機能

端子名	入出力	内容
PB3/CACREF	入力	測定基準クロックを入力
PB2/MTIOC0B	出力	HOCO を 4 分周したクロックを出力

2.2 RX140

表 2.3 RX140 動作確認条件

項目	内容
使用マイコン	R5F51403ADFM (RX140 グループ)
動作周波数	・ HOCO : 48MHz ・ システムクロック (ICLK) : 48MHz (HOCO の 1 分周) ・ 周辺モジュールクロック B (PCLKB) : 24MHz (HOCO の 2 分周) ・ 周辺モジュールクロック D (PCLKD) : 48MHz (HOCO の 1 分周) ・ FlashIF クロック (FCLK) : 48MHz (HOCO の 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 e ² studio 2022-04
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V.3.04.00 コンパイルオプション 統合開発環境のデフォルト設定
iodefine.h のバージョン	Version 1.10A
エンディアン	リトルエンディアン、ビッグエンディアン
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 1.10
使用ボード	Target Board for RX140 (型名 : RTK5RX1400CxxxxxBJ)
ファンクションジェネレータ	周波数精度が 2ppm(±0.0002%、18~28℃時)の矩形波を出力できるアナログ信号出力端子を備えた信号発生器。 出力信号は GND に対しバイアス 1.65V、振幅 3.3Vpp に設定した 32Hz の矩形波を出力します。図 2.3 に出力波形を示します。

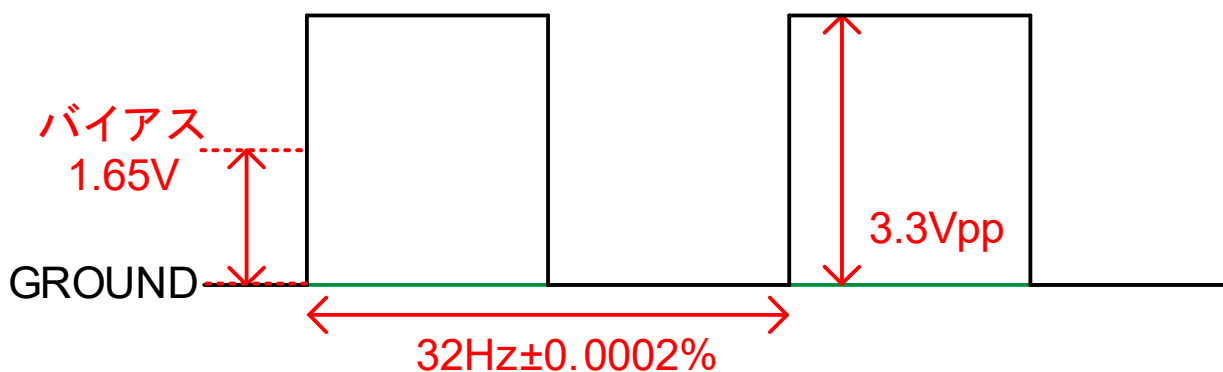
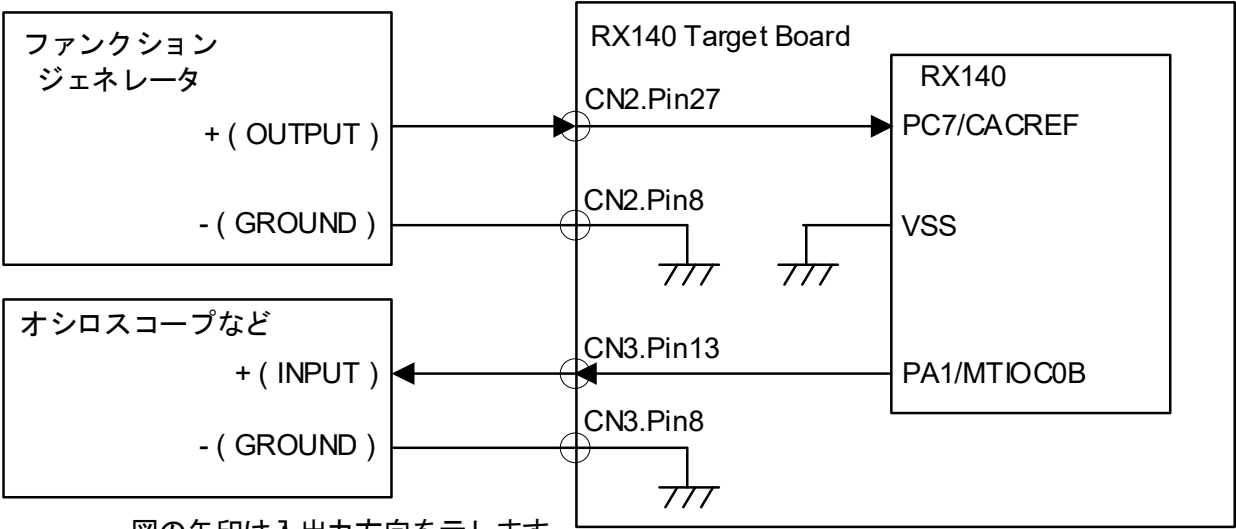


図 2.3 ファンクションジェネレータ 出力波形



図の矢印は入出力方向を示します。

図 2.4 RX140 端子接続

表 2.4 RX140 使用端子と機能

端子名	入出力	内容
PC7/CACREF	入力	測定基準クロックを入力
PA1/MTIOC0B	出力	HOCO を 4 分周したクロックを出力

2.3 RX231

表 2.5 RX231 動作確認条件

項目	内容
使用マイコン	R5F52318ADFP (RX231 グループ)
動作周波数	・ HOCO : 54MHz ・ システムクロック (ICLK) : 54MHz (HOCO の 1 分周) ・ 周辺モジュールクロック A (PCLKA) : 54MHz (HOCO の 1 分周) ・ 周辺モジュールクロック B (PCLKB) : 27MHz (HOCO の 2 分周) ・ 周辺モジュールクロック D (PCLKD) : 27MHz (HOCO の 2 分周) ・ FlashIF クロック (FCLK) : 27MHz (HOCO の 2 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 e ² studio 2022-04
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V.3.04.00 コンパイルオプション 統合開発環境のデフォルト設定
iodefine.h のバージョン	Version 1.00I
エンディアン	リトルエンディアン、ビッグエンディアン
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 1.10
使用ボード	Target Board for RX231 (型名 : RTK5RX2310CxxxxxBR)
ファンクションジェネレータ	周波数精度が 2ppm(±0.0002%、18~28°C時)の矩形波を出力できるアナログ信号出力端子を備えた信号発生器。 出力信号は GND に対しバイアス 1.65V、振幅 3.3Vpp に設定した 32Hz の矩形波を出力します。図 2.5 に出力波形を示します。

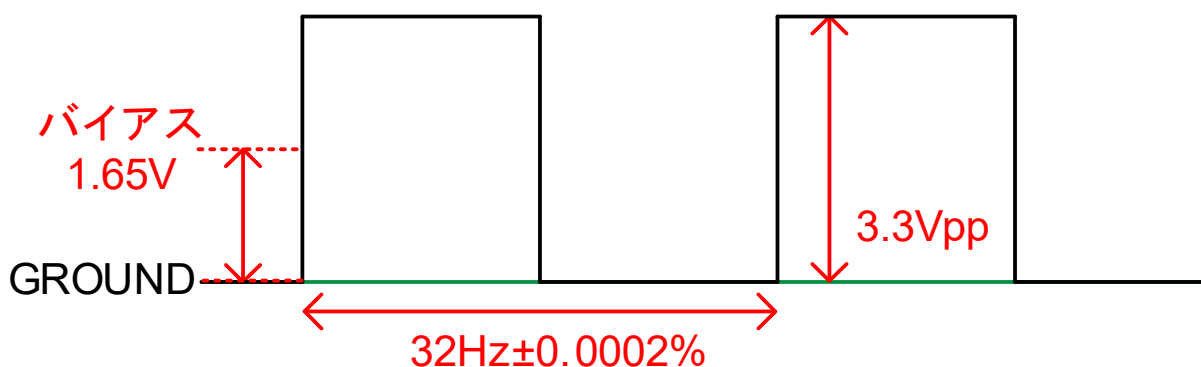


図 2.5 ファンクションジェネレータ 出力波形

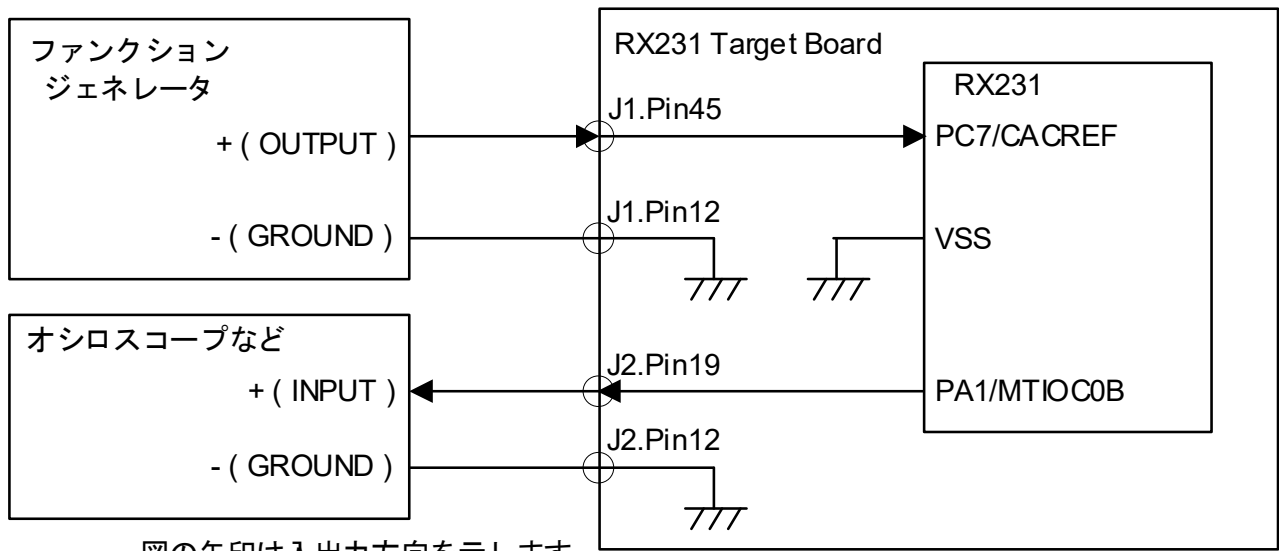


図 2.6 RX231 端子接続

表 2.6 RX231 使用端子と機能

端子名	入出力	内容
PC7/CACREF	入力	測定基準クロックを入力
PA1/MTIOC0B	出力	HOCO を 4 分周したクロックを出力

2.4 RX671

表 2.7 RX671 動作確認条件

項目	内容
使用マイコン	R5F5671EHAFP (RX671 グループ)
動作周波数	・ HOCO : 16MHz ・ PLL : 240MHz (HOCO 1 分周 15 通倍) ・ システムクロック (ICLK) : 120MHz (PLL の 2 分周) ・ 周辺モジュールクロック A (PCLKA) : 120MHz (PLL の 2 分周) ・ 周辺モジュールクロック B (PCLKB) : 60MHz (PLL の 4 分周) ・ 周辺モジュールクロック C (PCLKC) : 60MHz (PLL の 4 分周) ・ 周辺モジュールクロック D (PCLKD) : 60MHz (PLL の 4 分周) ・ FlashIF クロック (FCLK) : 60MHz (PLL の 4 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 e ² studio 2022-04
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V.3.04.00 コンパイルオプション 統合開発環境のデフォルト設定
iodefine.h のバージョン	Version 1.00
エンディアン	リトルエンディアン、ビッグエンディアン
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 1.10
使用ボード	Target Board for RX671 (型名 : RTK5RX6710CxxxxxBJ)
ファンクションジェネレータ	周波数精度が 2ppm(±0.0002%、18~28℃時)の矩形波を出力できるアナログ信号出力端子を備えた信号発生器。 出力信号は GND に対しバイアス 1.65V、振幅 3.3Vpp に設定した 32Hz の矩形波を出力します。図 2.7 に出力波形を示します。

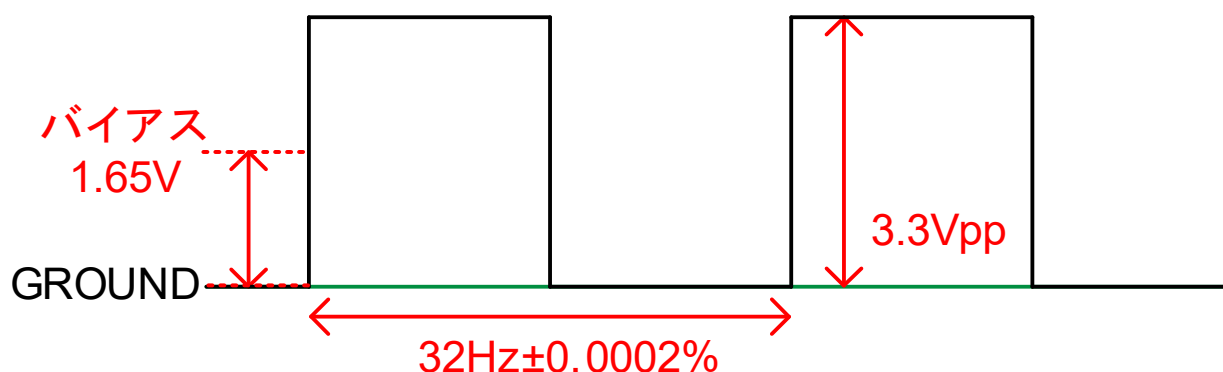


図 2.7 ファンクションジェネレータ 出力波形

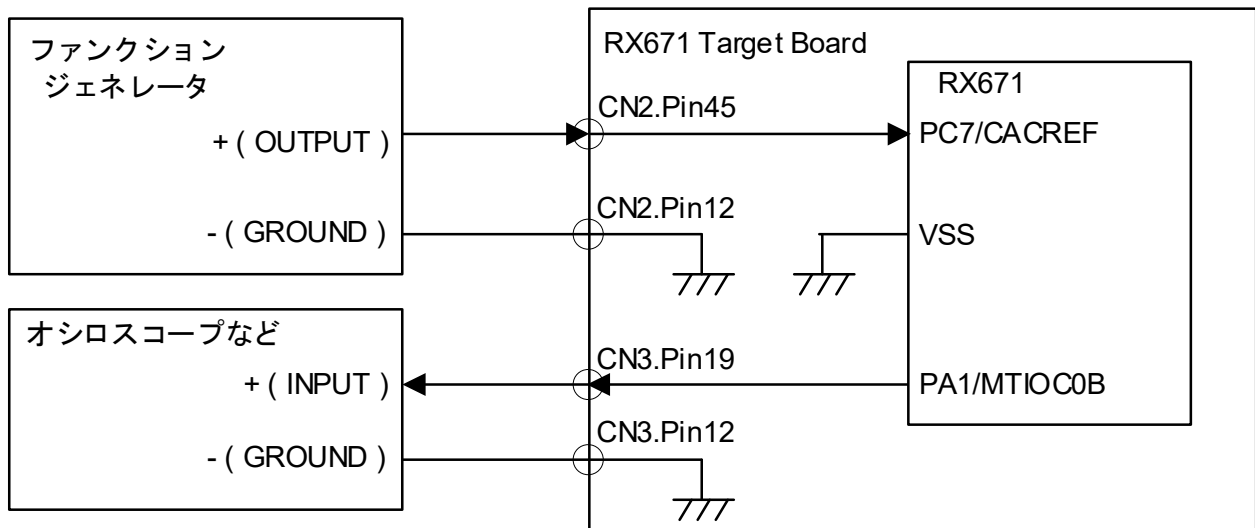


図 2.8 RX671 端子接続

表 2.8 RX671 使用端子と機能

端子名	入出力	内容
PC7/CACREF	入力	測定基準クロックを入力
PA1/MTIOC0B	出力	HOCO を 4 分周したクロックを出力

3. 使用ソフトウェアモジュール

図 3.1 にサンプルコードの構成、表 3.1 に使用ソフトウェアモジュールを示します。なお、サンプルコードで使用しているソフトウェアモジュールの設定については 7.2 ソフトウェアモジュールの設定を確認してください。

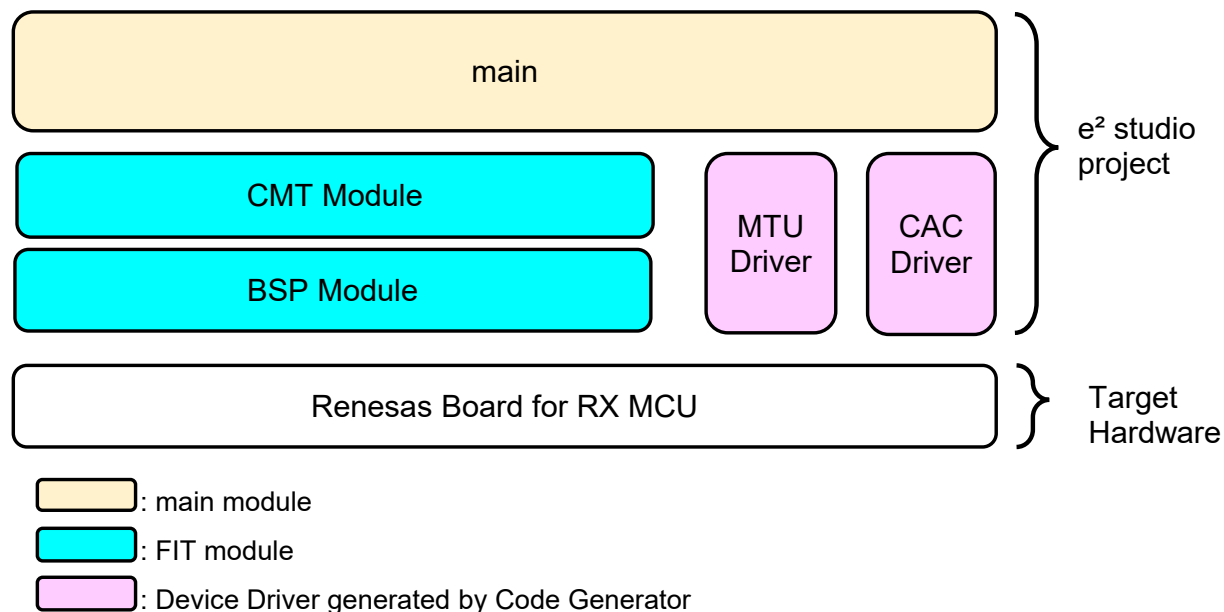


図 3.1 サンプルコードの構成

表 3.1 使用ソフトウェアモジュール

モジュール	ドキュメントタイトル	ドキュメント番号	種類
main	-	-	本アプリケーションノートで開発した main 関数を含むモジュール
r_bsp	RX ファミリ ボードサポート パッケージモジュール Firmware Integration Technology	R01AN1685	FIT モジュール
Config_CAC	スマート・コンフィグレータ ユーザーズマニュアル RX API リファレンス編	R20UT4360	コード生成
Config_MTU	スマート・コンフィグレータ ユーザーズマニュアル RX API リファレンス編	R20UT4360	コード生成
r_cmt_rx	RX ファミリ CMT モジュール Firmware Integration Technology	R01AN1856	FIT モジュール

4. ファイル構成

図 4.1 に本アプリケーションノートのファイル構成を示します。

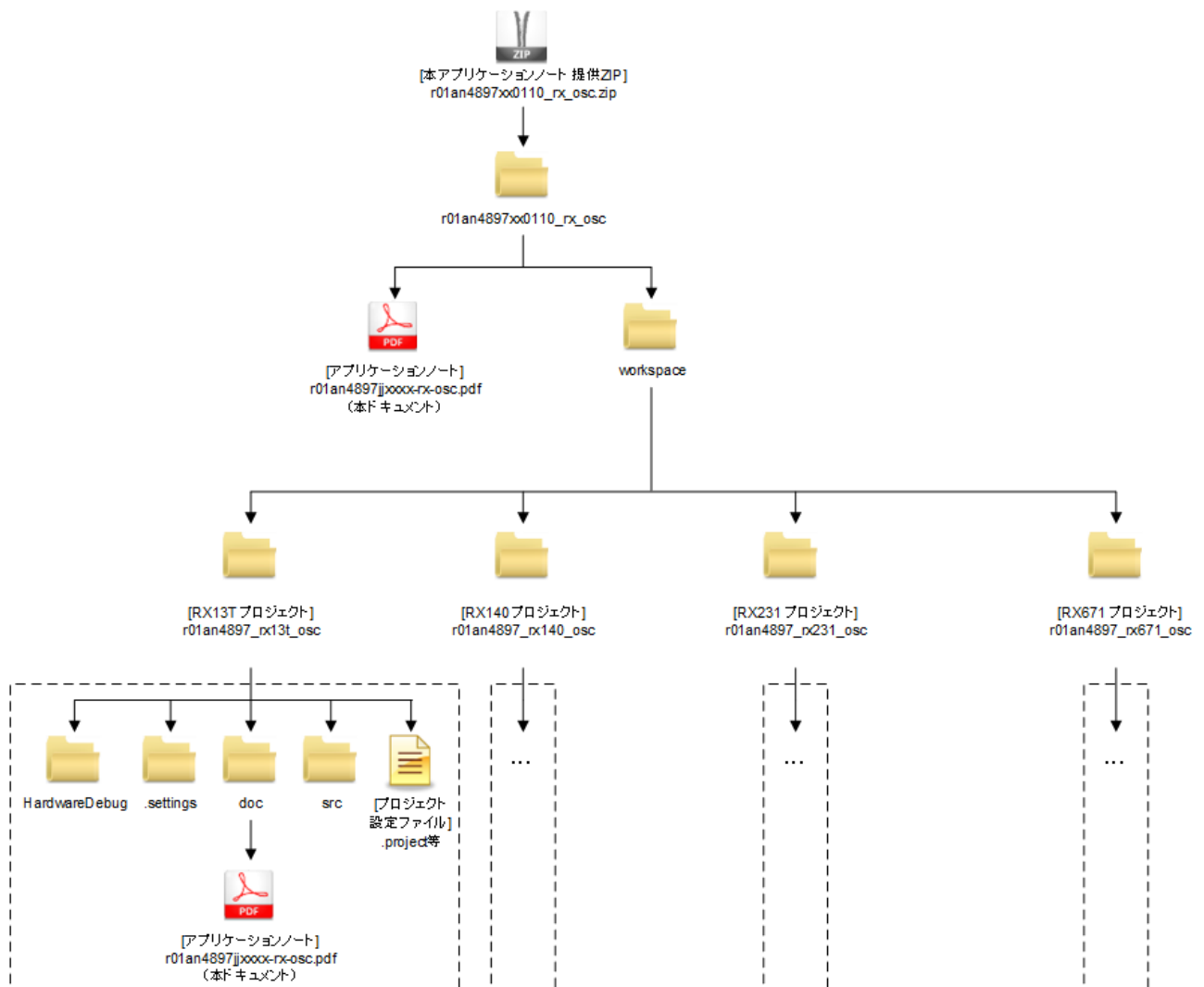


図 4.1 ファイル構成

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。

「r01an4897_rx_osc」フォルダ下に対象デバイス毎のサンプルプロジェクトが展開されます。各プロジェクトは e²studio 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。

5. ソフトウェア説明

5.1 動作概要

本章では RX13T を例に説明します。

本サンプルプログラムは、MCU 外部から与える測定基準クロックに対し、MCU が生成したクロックの周波数を測定し、周波数のずれを小さくするように動作します。（図 1.1 参照）

- ・ 測定基準クロック
ファンクションジェネレータを使用、32Hz の矩形波を出力するよう設定（図 2.1 を参照）
- ・ 測定対象クロック
HOCO（32MHz）とし、分周比を 1/32 に設定
- ・ CAC
測定基準クロックの立ち上がりエッジから次の立ち上がりエッジまでの期間に、測定対象クロックの有効エッジをカウント

測定は、コンペアマッチタイマ（以下、CMT）のコンペアマッチ割り込みで開始し、5 回連続で測定します。測定した 5 回のうち 2 回目から 5 回目の平均値を測定結果とし、キャリブレーション状態を遷移させます。表 5.1 にキャリブレーションの状態、図 5.1 にキャリブレーション状態遷移図を示します。

表 5.1 キャリブレーションの状態

キャリブレーションの状態名	説明
CALIBRATION_STANDBY	キャリブレーション待機中
CALIBRATION_START	キャリブレーション開始
CALIBRATION_RESULT_ABOVE	周波数を小さく調整し続ける
CALIBRATION_RESULT_BELOW	周波数を大きく調整し続ける
CALIBRATION_WITHIN_RANGE	キャリブレーション安定

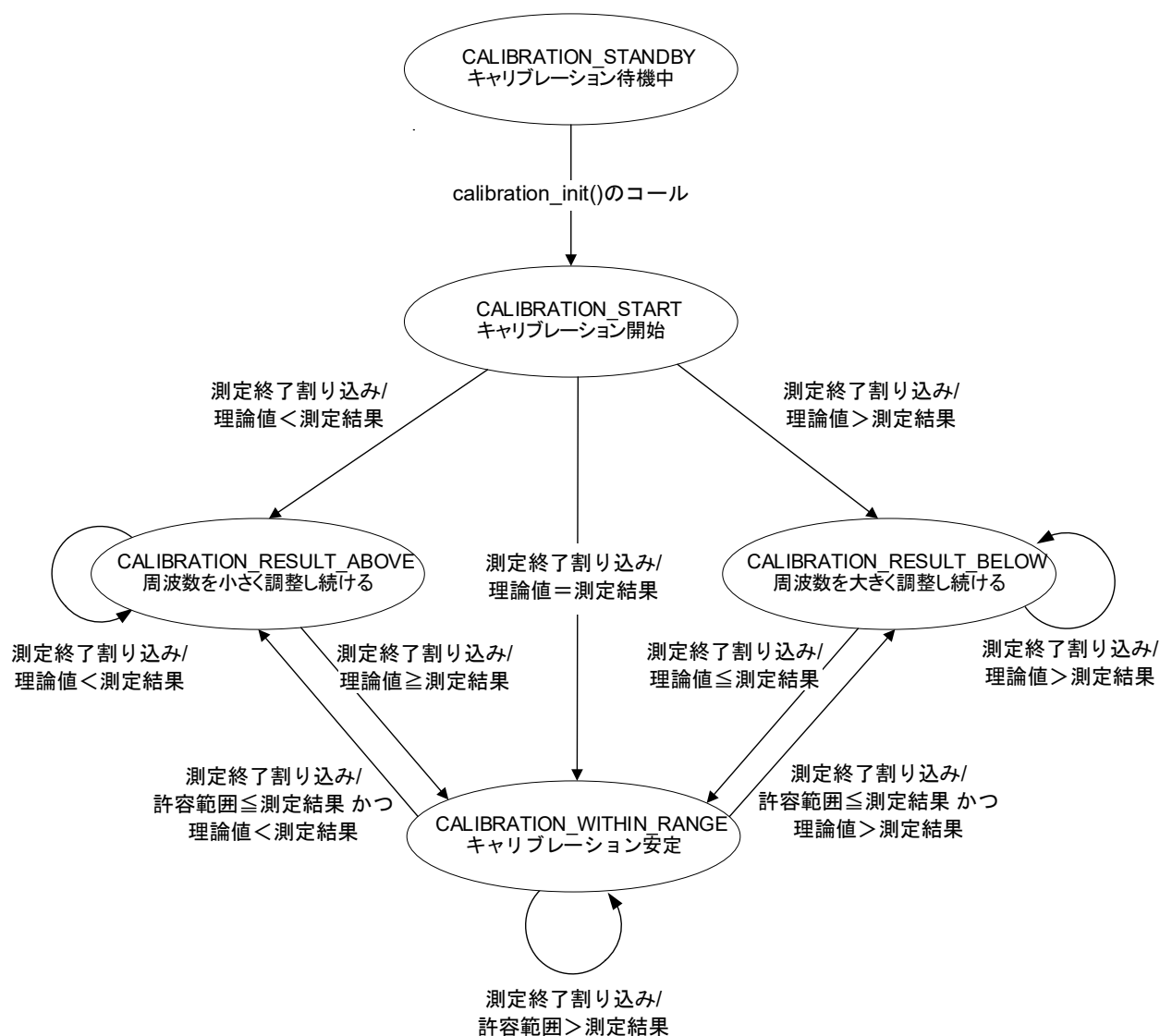


図 5.1 キャリブレーション状態遷移図

キャリブレーション状態の遷移後に理論値との差分及びキャリブレーション状態に応じて HOCOTRRn レジスタの値を加算または減算します。表 5.2 にレジスタ値の調整パターン、図 5.2 及び図 5.3 にキャリブレーション例、図 5.4 にキャリブレーションのタイミング図を示します。

表 5.2 レジスタ値の調整パターン

パターン	キャリブレーションの状態	条件	レジスタ値の変更
1	CALIBRATION_RESULT_ABOVE	$X < (Z+M)$	-1
2		$X \geq (Z+M)$	-5
3	CALIBRATION_RESULT_BELOW	$X > (Z-M)$	+1
4		$X \leq (Z-M)$	+5
5	CALIBRATION_WITHIN_RANGE	$(Z+M) > X > (Z+L)$	-1
6		$X \geq (Z+M)$	-5
7		$(Z-M) \leq X < (Z-L)$	+1
8		$X < (Z-M)$	+5
9		$(Z-L) \leq X \leq (Z+L)$	0

X: 測定結果、Z: 理論値 (注 1)、L: 許容範囲 (注 2)、M: 変更量を決定する基準値 (注 3)

注 3 理論値は以下の式で決定されます。

理論値 $Z = \text{測定対象クロックの周波数} / \text{測定基準クロックの周波数}$ (小数点以下切り捨て)

注 2 許容範囲は以下の式で決定されます。

許容範囲 $L = Z * \text{ACCEPTABLE_PERCENT} / 10000$ (小数点以下切り捨て)

注 3 変更量を決定する基準値は以下の式で決定されます。

変更量を決定する基準値 $M = Z * 0.003$ (小数点以下切り捨て)

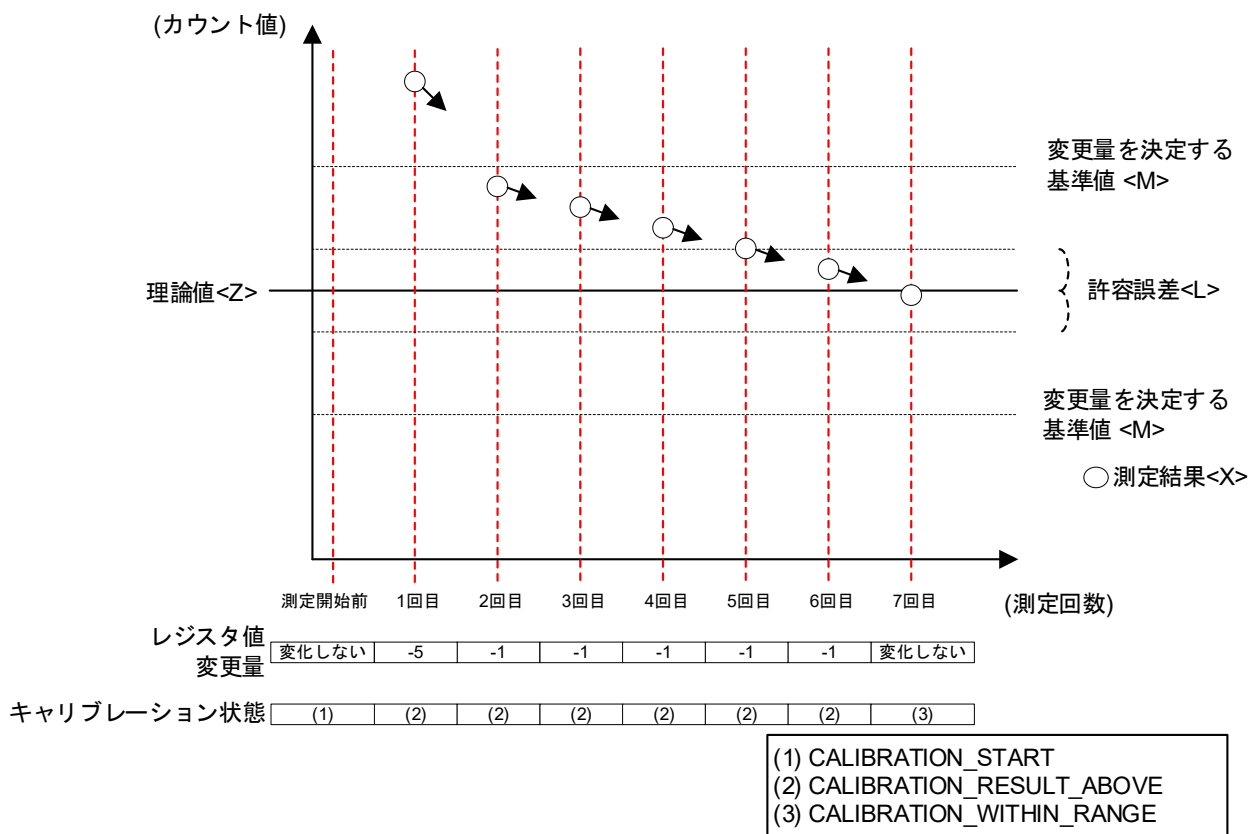


図 5.2 測定した HOCO 周波数が理論値より大きい状態からのキャリブレーション例

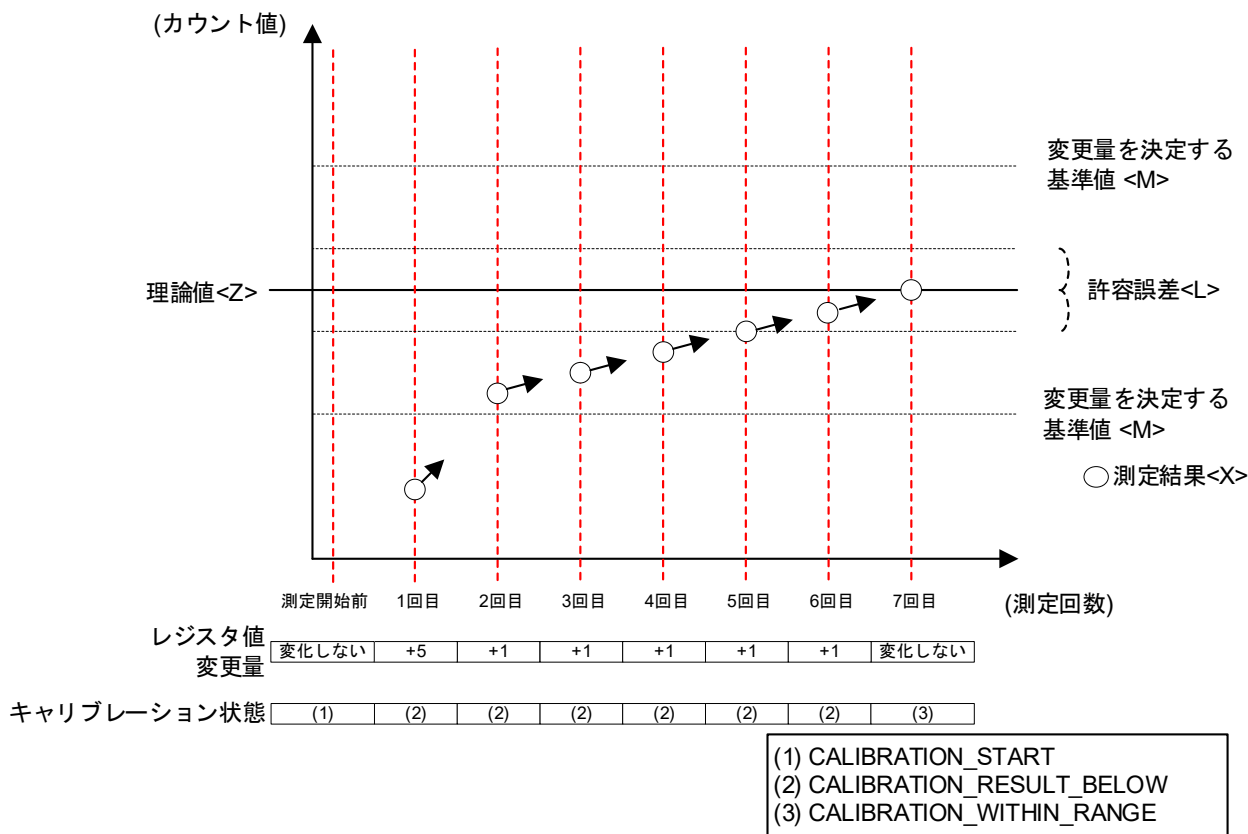


図 5.3 測定した HOCO 周波数が理論値より小さい状態からのキャリブレーション例

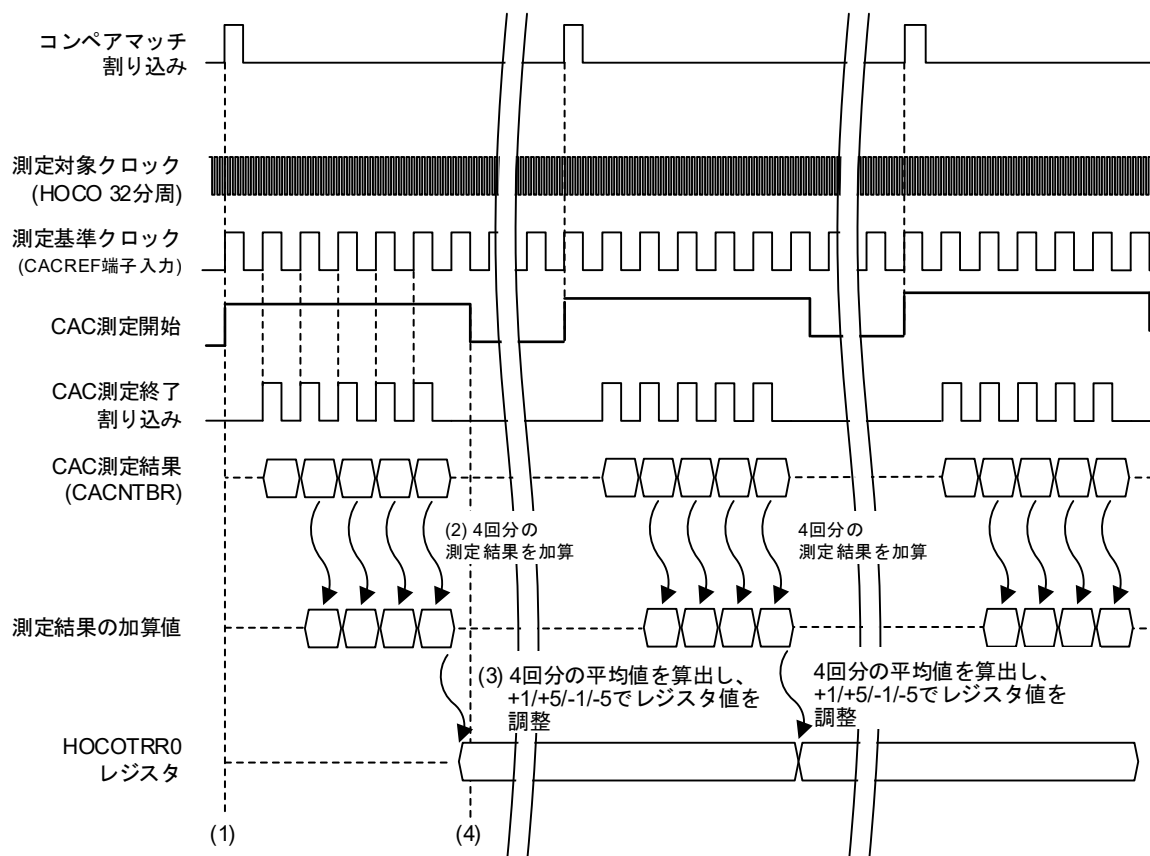


図 5.4 キャリブレーションのタイミング図

- (1) CMT のコンペアマッチ割り込みで CAC の起動設定と測定開始を行います。
- (2) 測定基準クロックの立ち上がりエッジから立ち上がりエッジまでの期間に、測定対象クロックの有効エッジをカウントします。測定完了時の割り込みで、測定結果を取得します。測定結果は 5 回 1 セットとします。
- (3) 1 セット分の測定結果の 2 回目から 5 回目より平均値を算出し、その値とキャリブレーション状態によって HOCOTRRn レジスタの値を調整します。
- (4) HOCO の発振周波数の調整が終わると、CAC の測定を停止させます。次の CMT のコンペアマッチ割り込みが発生するまで、ICLK など CPU や周辺機能に HOCO のクロックを供給します。

5.2 ファイル構成

表 5.3 に本アプリケーションノートで作成したソースファイルを表に示します。

表 5.3 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_calibration_func.c	キャリブレーション設定と CAC 測定完了割り込み処理	
r_calibration_func.h	r_calibration_func.c のヘッダファイル	

5.3 定数一覧

表 5.4、表 5.5 に本アプリケーションノートで作成したソースファイルで使用する定数を示します。

表 5.4 サンプルコードで使用する定数(ユーザ変更可)

定数名	設定値	内容
REG_HOCOTRR	SYSTEM.HOCOTRR0.BIT.HOCOTRD ^(注1) SYSTEM.HOCOTRR3.BIT.HOCOTRD ^(注2)	動作させている HOCO に対応した HOCOTRRn レジスタ
ACCEPTABLE_PERCENT	10	測定結果に対する許容範囲パーセンテージ (0.01%刻み) "10"と設定した場合、許容範囲は"0.1%"になります。 変更する場合は、"5" 以上 (0.05%)に設定してください。
CHECK_CNT	4	測定結果加算回数

注 1. RX13T、RX140、RX671 のサンプルコードでの設定値です。

注 2. RX231 のサンプルコードでの設定値です。

表 5.5 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
REG_HOCOTRR_MIN	0x00000000	HOCOTRRn レジスタに設定可能な最小値
REG_HOCOTRR_MAX	0x000001FF ^(注1) 0x0000003F ^(注2)	HOCOTRRn レジスタに設定可能な最大値
HOCOTRR_CHANGE_AFTER_WAIT_US	65	HOCOTRRn レジスタ変更後の待ち時間 ^(注3)
CALIBRATION_STANDBY	0	キャリブレーション待機中
CALIBRATION_START	1	キャリブレーション開始
CALIBRATION_RESULT_ABOVE	2	周波数を小さく調整し続ける
CALIBRATION_RESULT_BELOW	3	周波数を大きく調整し続ける
CALIBRATION_WITHIN_RANGE	4	キャリブレーション安定
ACCEPTABLE_OVER_PERCENT	30	測定結果に対する変更量を決定する基準値のパーセンテージ (0.01%刻み)
CMT_INTERRUPT_PERIOD_HZ	2 ^(注1) 1 ^(注2)	CMT のコンペアマッチ割り込み周期
MEND_FINISH	0	周波数測定は停止
MEND_START	1	周波数測定は開始
MEND_CHECK_FINISH	(CHECK_CNT+1)	周波数測定は完了

注 1.RX671 のサンプルコードでの設定値です。

注 2.RX13T、RX140、RX231 のサンプルコードでの設定値です。

注 3.RX671 のプロジェクトのみ定義されます。

5.4 変数一覧

表 5.6 に本アプリケーションノートで作成したソースファイルで使用するグローバル変数を示します。

表 5.6 グローバル変数

型	変数名	内容	使用関数
volatile static uint8_t	s_buffer_counter	周波数測定回数カウンタ	calibration_init cac_mendf_cb
volatile static uint32_t	s_result_buffer	周波数測定結果格納バッファ	calibration_init cac_mendf_cb
volatile static int32_t	s_result_diff	結果の格納バッファ	calibration_init cac_mendf_cb
volatile static uint32_t	s_acceptable_range	測定結果に対する許容範囲	calibration_init cac_mendf_cb
volatile static uint32_t	s_reference_value	HOCOTRRn レジスタの変更量を決定する基準値	calibration_init cac_mendf_cb
volatile static uint8_t	s_freq_calibration_status	キャリブレーション状態	calibration_init cac_mendf_cb

5.5 関数一覧

表 5.7 に本アプリケーションノートで作成したソースファイルで使用する関数一覧を示します。

表 5.7 関数一覧

関数名	概要
main	メイン処理
cmt_event_cb	CMT のコンペアマッチ割り込み処理
calibration_init	キャリブレーションの設定処理
cac_mendf_cb	CAC の測定完了割り込み処理

5.6 関数仕様

サンプルコードの関数仕様を示します。

main	
概 要	メイン処理
ヘッダ	なし
宣 言	void main (void)
説 明	CMT の設定および MTU の設定を呼び出します。
引 数	なし
リターン値	なし
cmt_event_cb	
概 要	CMT のコンペアマッチ割り込み処理
ヘッダ	なし
宣 言	static void cmt_event_cb(void)
説 明	CMT のコンペアマッチ割り込みで、周期的に calibration_init 関数を呼び出します。
引 数	なし
リターン値	なし
calibration_init	
概 要	キャリブレーションの初期設定
ヘッダ	r_calibration_func.h
宣 言	void calibration_init(void)
説 明	CAC の設定とキャリブレーションの設定を行います。この関数は cmt_event_cb に よって周期的に呼び出されます。
引 数	なし
リターン値	なし
cac_mendf_cb	
概 要	CAC の測定完了割り込み処理
ヘッダ	r_calibration_func.h
宣 言	void cac_mendf_cb(void)
説 明	測定結果をもとに HOCO のキャリブレーションを行います。
引 数	なし
リターン値	なし

5.7 フローチャート

5.7.1 メイン処理

図 5.5 にメイン処理のフローチャートを示します。

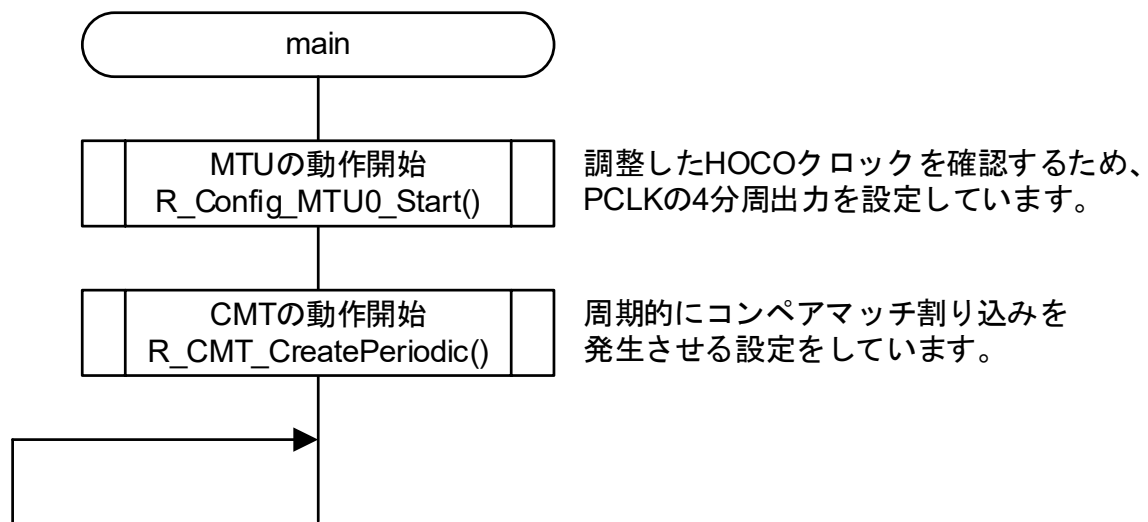


図 5.5 メイン処理

5.7.2 CMT のコンペアマッチ割り込み

図 5.6 に CMT のコンペアマッチ割り込み処理のフローチャートを示します。

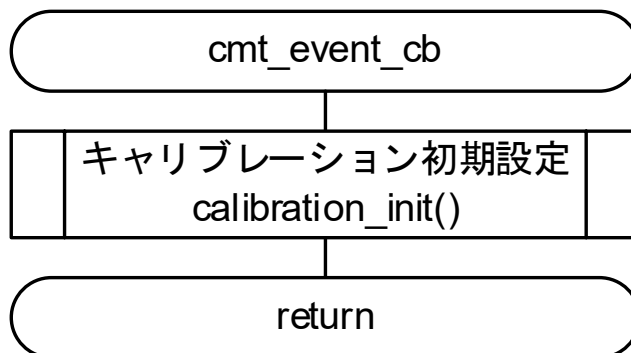


図 5.6 CMT のコンペアマッチ割り込み処理

5.7.3 キャリブレーションの設定処理

図 5.7 にキャリブレーションの設定処理のフローチャートを示します。

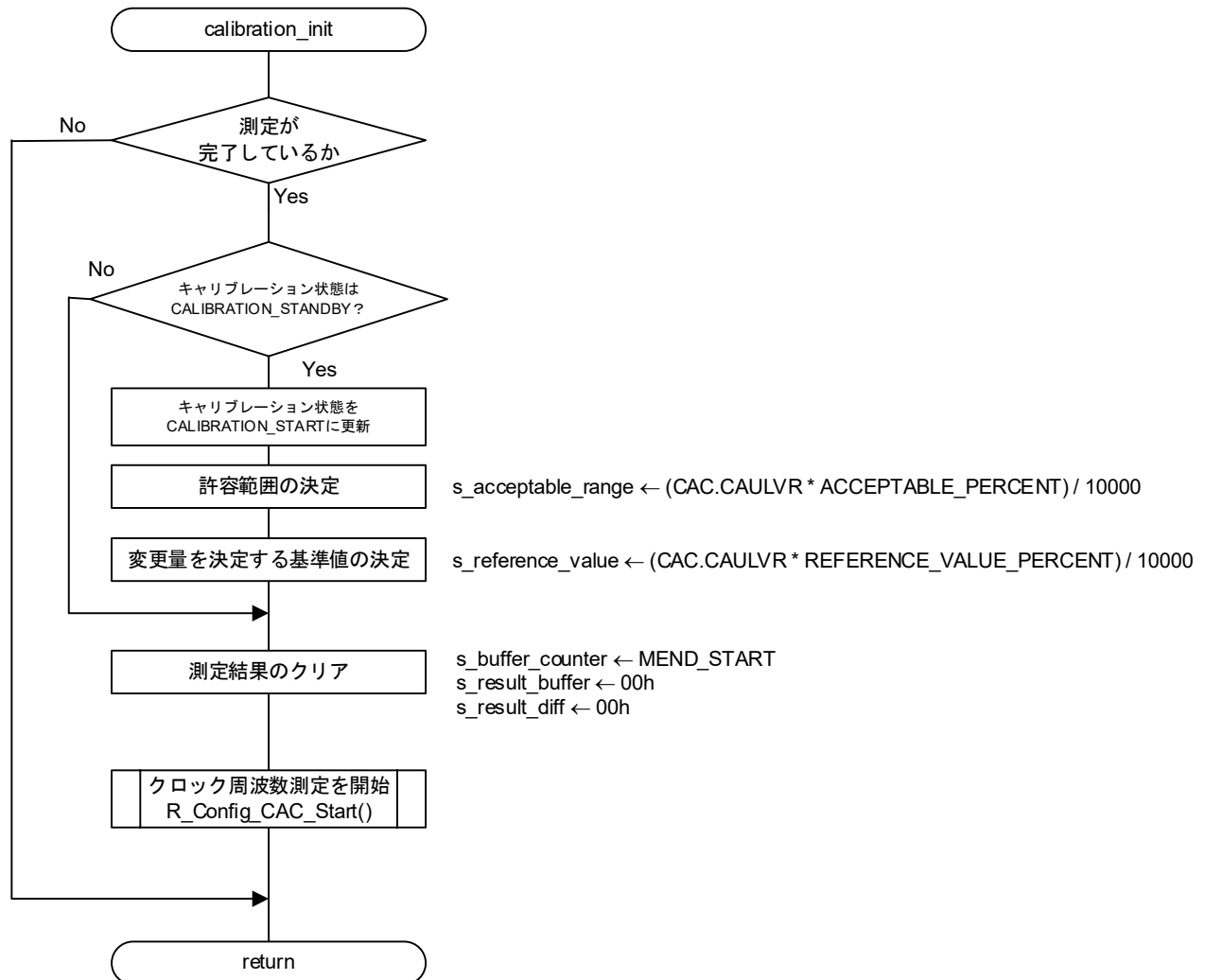


図 5.7 キャリブレーションの設定処理

5.7.4 CAC の測定完了割り込み

図 5.8 に CAC の測定完了割り込み処理のフローチャートを示します。

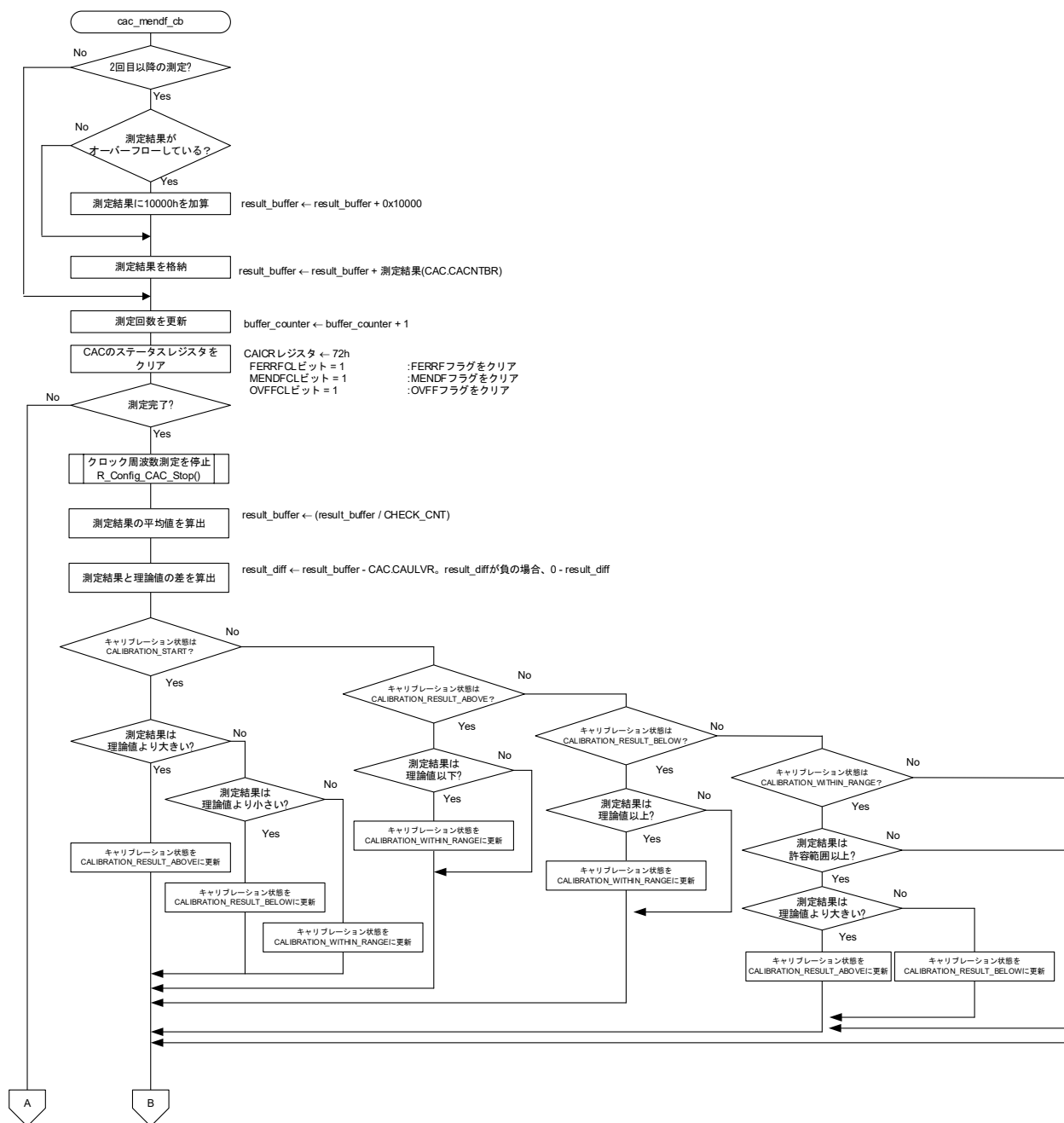


図 5.8 CAC の測定完了割り込み処理 (1/2)

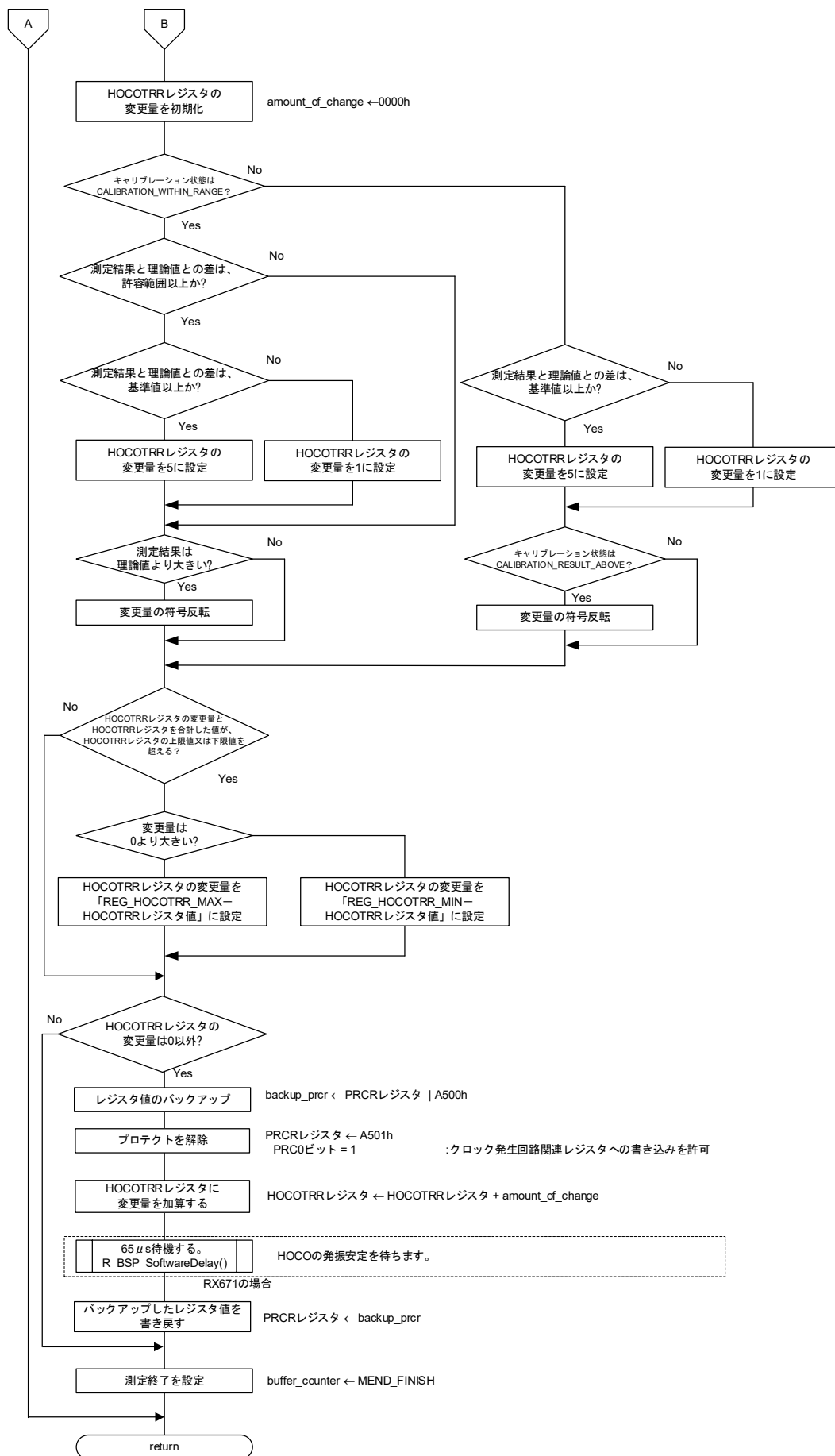


図 5.9 CAC の測定完了割り込み処理 (2/2)

6. 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 e² studio のプロジェクト形式で提供しています。本章では、e² 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。

本アプリケーションノートは、下記の開発ツールに対応しています。

- e² studio Version:2022-04 および RX Compiler CC-RX V3.04.00
- CS+ V8.07.00 および RX Compiler CC-RX V3.04.00

6.1 e² studio での手順

e² studio でご使用になる際は、下記の手順で e² studio にインポートしてください。
(使用する e² 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。)

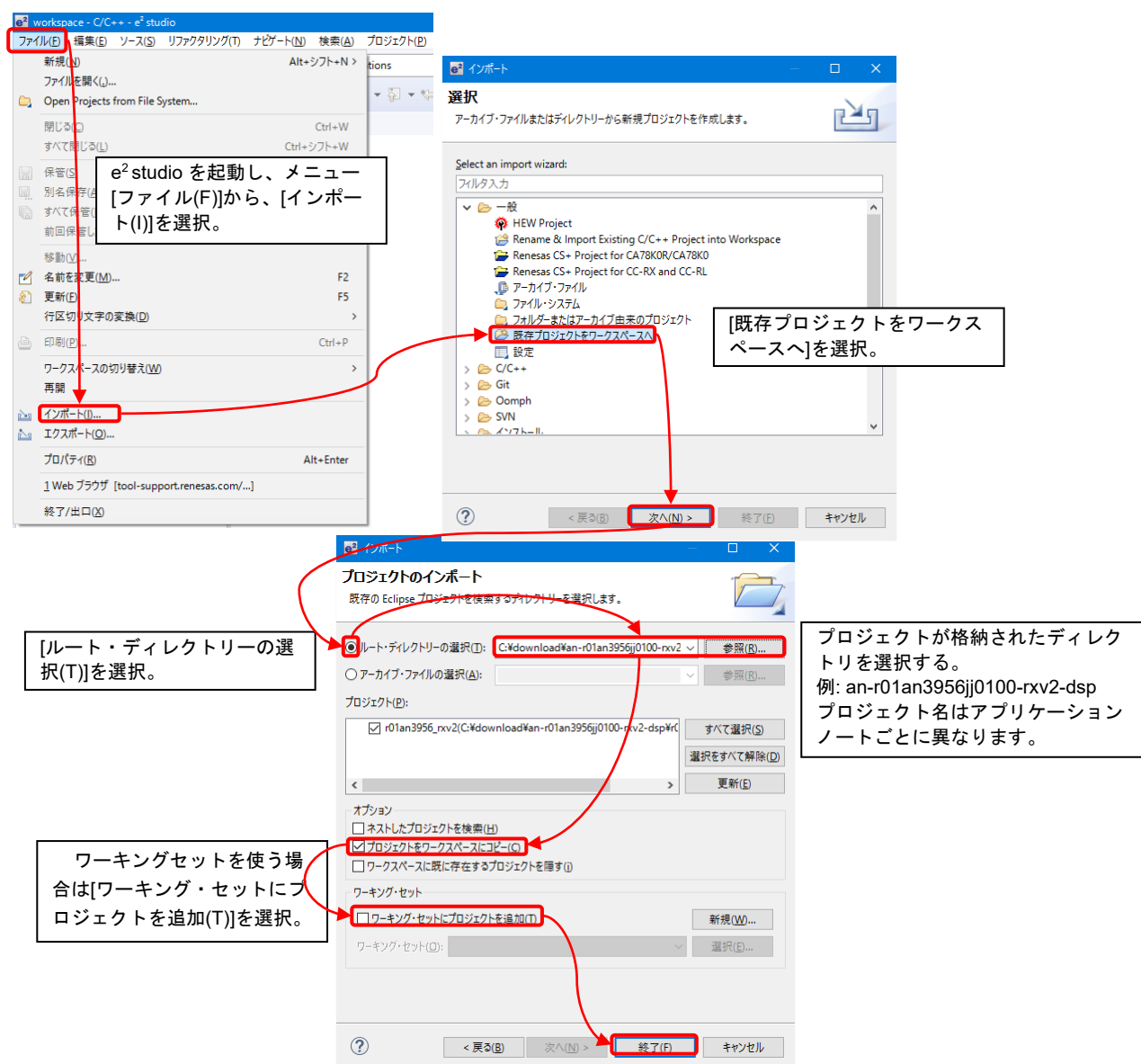


図 6.1 プロジェクトを e² studio にインポートする方法

6.2 CS+での手順

本アプリケーションノートには、e² studio 用のプロジェクトのみが同梱されています。

そのプロジェクトを CS+と組み合わせてご使用になる場合、下記の手順にて読み込んでください。

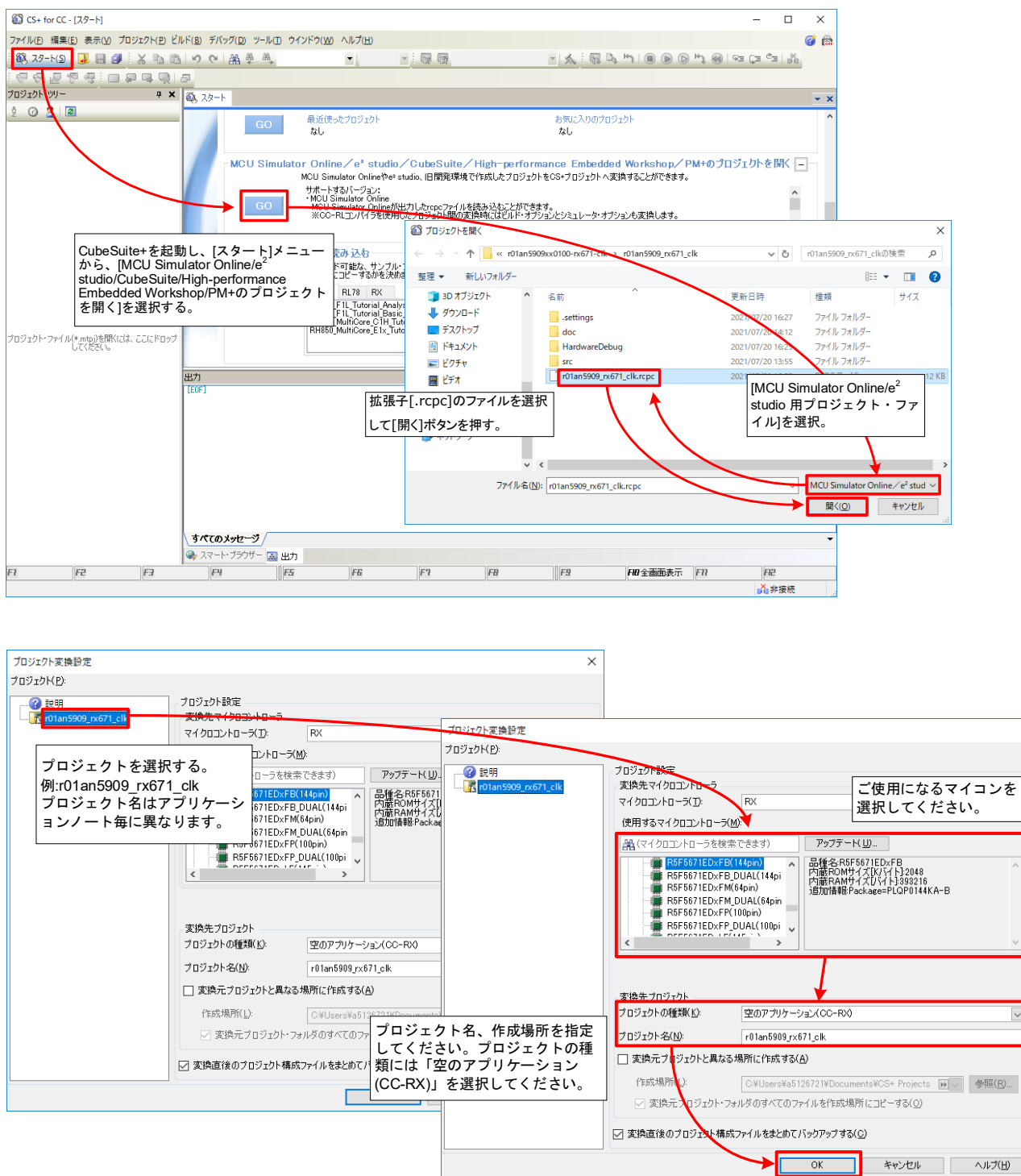


図 6.2 プロジェクトを CS+にインポートする方法

7. 付録

7.1 MTU の端子出力を行わない場合

R_Config_MTU0_Start 関数をコメントアウトしてください。

7.2 ソフトウェアモジュールの設定

サンプルプログラムで使用している FIT モジュールおよび e²studio のスマート・コンフィグレータの設定を下記に示します。スマート・コンフィグレータの設定における各表の項目、設定内容は設定画面の表記で記載しています。各 FIT モジュールの詳細は、各 FIT モジュールのドキュメントを参照してください。

7.2.1 RX13T

表 7.1 BSP モジュールの設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_bsp		デフォルトの設定とする。(変更なし)
スマート・コンフィグレータ >> クロック		「クロック」タブを下記の設定とし r_bsp_config.h に反映させる。
	HOCO クロック >> リセット後、HOCO 発振が有効	リセット後、HOCO 発振が無効：チェックを外す

表 7.2 クロック周波数精度測定回路の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_CAC		下記の変更以外はデフォルトの設定とする。
	基準信号設定	
	クロック選択	CACREF 端子に変更
	周波数	0.000032 (MHz)に変更
	周波数測定設定	
	クロック選択	高速オンチップオシレータに変更
	周波数	1/32 に変更
	カウンタ値設定	
	上限値	0%に変更
	下限値	0%に変更
	割り込み設定	
	周波数エラー割り込みを許可(FERRI)	使用しない：チェックを外す
	測定終了割り込みを許可(MENDI) >> 優先順位(MENDF)	レベル 2 に変更
	オーバーフロー割り込みを許可(OVFI)	使用しない：チェックを外す
スマート・コンフィグレータ >> 端子 >> クロック周波数精度計測回路		下記の変更以外はデフォルトの設定とする。
	CACREF	端子割り当て：PB3 に設定

表 7.3 PWM モードタイマ(PWM モード 2)の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_MTU0		下記の変更以外はデフォルトの設定とする。
	出力端子の設定	
	MTIOCB0 の端子	端子初期出力は 0、コンペアマッチで 1 出力、カウンタクリアで 0 出力に変更
	PWM 出力設定	
	PWM 周期	4 カウントに変更
	TGRB 初期値	1 に変更
スマート・コンフィグレータ >> 端子 >> マルチファンクションタイマパルスユニット 3 >> MTU0		下記の変更以外はデフォルトの設定とする。
	MTIOC0B	端子割り当て : PB2 に設定

表 7.4 r_cmt_rx の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_cmt_rx		下記の変更以外はデフォルトの設定とする。
	Configurations	
	CMT interrupts priority level	1 に変更

7.2.2 RX140

表 7.5 BSP モジュールの設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_bsp		デフォルトの設定とする。(変更なし)
スマート・コンフィグレータ >> クロック		「クロック」タブを下記の設定とし r_bsp_config.h に反映させる。
	HOCO クロック >> リセット後、HOCO 発振が有効	リセット後、HOCO 発振が無効：チェックを外す

表 7.6 クロック周波数精度測定回路の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_CAC		下記の変更以外はデフォルトの設定とする。
	基準信号設定	
	クロック選択	CACREF 端子に変更
	周波数	0.000032 (MHz)に変更
	周波数測定設定	
	クロック選択	高速オンチップオシレータに変更
	周波数	1/32 に変更
	カウンタ値設定	
	上限値	0%に変更
	下限値	0%に変更
	割り込み設定	
	周波数エラー割り込みを許可(グループ BL0)	使用しない：チェックを外す
	測定終了割り込みを許可(MENDI) >> 優先順位(グループ BL0)	レベル 2 に変更
	オーバーフロー割り込みを許可(グループ BL0)	使用しない：チェックを外す
スマート・コンフィグレータ >> 端子 >> クロック周波数精度計測回路		下記の変更以外はデフォルトの設定とする。
	CACREF	端子割り当て：PC7 に設定

表 7.7 PWM モードタイマ(PWM モード 2)の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_MTU0		下記の変更以外はデフォルトの設定とする。
	出力端子の設定	
	MTIOCB0 の端子	端子初期出力は 0、コンペアマッチで 1 出力、カウンタクリアで 0 出力に変更
	PWM 出力設定	
	PWM 周期	4 カウントに変更
	TGRB 初期値	1 に変更
スマート・コンフィグレータ >> 端子 >> マルチファンクションタイマパルスユニット 3 >> MTU0		下記の変更以外はデフォルトの設定とする。
	MTIOC0B	端子割り当て : PA1 に設定

表 7.8 r_cmt_rx の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_cmt_rx		下記の変更以外はデフォルトの設定とする。
	Configurations	
	CMT interrupts priority level	1 に変更

7.2.3 RX231

表 7.9 BSP モジュールの設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_bsp		デフォルトの設定とする。(変更なし)
スマート・コンフィグレータ >> クロック		「クロック」タブを下記の設定とし r_bsp_config.h に反映させる。
	HOCO クロック >> リセット後、HOCO 発振が有効	リセット後、HOCO 発振が無効：チェックを外す
	HOCO クロック >> 周波数：32MHz	周波数：54MHz に変更

表 7.10 クロック周波数精度測定回路の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_CAC		下記の変更以外はデフォルトの設定とする。
	基準信号設定	
	クロック選択	CACREF 端子に変更
	周波数	0.000032 (MHz)に変更
	周波数測定設定	
	クロック選択	高速オンチップオシレータに変更
	周波数	1/32 に変更
	カウンタ値設定	
	上限値	0%に変更
	下限値	0%に変更
	割り込み設定	
	周波数エラー割り込みを許可(グループ BL0)	使用しない：チェックを外す
	測定終了割り込みを許可(MENDI) >> 優先順位(グループ BL0)	レベル 2 に変更
	オーバーフロー割り込みを許可(グループ BL0)	使用しない：チェックを外す
スマート・コンフィグレータ >> 端子 >> クロック周波数精度計測回路		下記の変更以外はデフォルトの設定とする。
	CACREF	端子割り当て：PC7 に設定

表 7.11 PWM モードタイマ(PWM モード 2)の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_MTU0		下記の変更以外はデフォルトの設定とする。
	出力端子の設定	
	MTIOCB0 の端子	端子初期出力は 0、コンペアマッチで 1 出力、カウンタクリアで 0 出力に変更
	PWM 出力設定	
	PWM 周期	4 カウントに変更
	TGRB 初期値	1 に変更
スマート・コンフィグレータ >> 端子 >> マルチファンクションタイマパルスユニット 3 >> MTU0		下記の変更以外はデフォルトの設定とする。
	MTIOC0B	端子割り当て : PA1 に設定

表 7.12 r_cmt_rx の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_cmt_rx		下記の変更以外はデフォルトの設定とする。
	Configurations	
	CMT interrupts priority level	1 に変更

7.2.4 RX671

表 7.13 BSP モジュールの設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_bsp		デフォルトの設定とする。(変更なし)
スマート・コンフィグレータ >> クロック		「クロック」タブを下記の設定とし r_bsp_config.h に反映させる。
	HOCO クロック >> リセット後、HOCO 発振が有効	リセット後、HOCO 発振が無効：チェックを外す

表 7.14 クロック周波数精度測定回路の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_CAC		下記の変更以外はデフォルトの設定とする。
	基準信号設定	
	クロック選択	CACREF 端子に変更
	周波数	0.000032 (MHz)に変更
	周波数測定設定	
	クロック選択	高速オンチップオシレータに変更
	周波数	1/32 に変更
	カウンタ値設定	
	上限値	0%に変更
	下限値	0%に変更
	割り込み設定	
	周波数エラー割り込みを許可(グループ BL0)	使用しない：チェックを外す
	測定終了割り込みを許可(MENDI) >> 優先順位(グループ BL0)	レベル 2 に変更
	オーバーフロー割り込みを許可(グループ BL0)	使用しない：チェックを外す
スマート・コンフィグレータ >> 端子 >> クロック周波数精度計測回路		下記の変更以外はデフォルトの設定とする。
	CACREF	端子割り当て：PC7 に設定

表 7.15 PWM モードタイマ(PWM モード 2)の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> Config_MTU0		下記の変更以外はデフォルトの設定とする。
	出力端子の設定	
	MTIOCB0 の端子	端子初期出力は 0、コンペアマッチで 1 出力、カウンタクリアで 0 出力に変更
	PWM 出力設定	
	PWM 周期	30 カウントに変更
	TGRB 初期値	14 に変更
スマート・コンフィグレータ >> 端子 >> マルチファンクションタイマパルスユニット 3 >> MTU0		下記の変更以外はデフォルトの設定とする。
	MTIOC0B	端子割り当て : PA1 に設定

表 7.16 r_cmt_rx の設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_cmt_rx		下記の変更以外はデフォルトの設定とする。
	Configurations	
	CMT interrupts priority level	1 に変更

8. 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。

9. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX13T グループ ユーザーズマニュアル ハードウェア編 (R01UH0822)

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

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

RX671 グループ ユーザーズマニュアル ハードウェア編 (R01UH0899)

(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)

(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。)

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Nov.30.19	—	初版発行
1.10	Jun.30.22	3	「1 仕様」を RX ファミリ向けの内容に修正
		4～11	「2.1 動作確認条件」に RX140、RX231、RX671 の動作確認条件と端子接続を追加
		12	「3 関連アプリケーションノート」を「3 使用ソフトウェアモジュール」に変更
		13	「4 ファイル構成」を追加
		14～18	「5.1 動作概要」を仕様の変更に合わせて修正
		19	「5.4 オプション設定メモリ」にを削除
		19～21	「5.3 定数一覧」、「5.4 変数一覧」、「5.5 関数一覧」及び「5.6 関数仕様」を仕様の変更に合わせて修正
		22～26	「5.9 フローチャート」を関数の処理に合わせて修正
		27, 28	「6.プロジェクトをインポートする方法」を追加
		29～36	「7.付録」の「測定基準クロック周波数変更方法」と「コンペアマッチタイマのカウント数変更方法」を削除し、「ソフトウェアモジュールの設定」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、入力信号や入出力ブルアップ電源を入れないでください。入力信号や入出力ブ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、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くと、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、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）、交通管制（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。