

## ルネサス RX ファミリ

## IAR EWRX 向け QE for Capacitive Touch の使用方法

## 要旨

本アプリケーションノートでは、スマート・コンフィグレータで生成した IAR Embedded Workbench for Renesas RX (IAR EWRX) プロジェクトで、静電容量式タッチセンサ対応開発支援ツールのスタンドアロン版 QE for Capacitive Touch (スタンドアロン版 QE) を使用して、タッチセンサの自動調整とモニタリングをする手順について説明します。

## 動作確認評価キット

RSK-RX140(RTK551406BS00000BE) : CPU ボード(R5F51406BDFN)

## 動作環境

統合開発環境 (IDE)	IAR Embedded Workbench for Renesas RX バージョン 4.20.3 以降
コンフィグレーションツール	RX スマート・コンフィグレータ V2.15.0 以降
FIT モジュール	RX ファミリ RX Driver Package Ver.1.36 以降
	RX ファミリ用 Firmware Integration Technology (FIT)
	・ CTSU QE API (r_ctsu_qe): V2.10 (またはそれ以降)
	・ SCI Driver (r_sci_rx): V4.5 (またはそれ以降)
・ Touch QE API (rm_touch) : V2.10 (またはそれ以降)	
ツールチェーン	IAR EWRX Toolchain
QE	スタンドアロン版 QE for Capacitive Touch V3.1.0 以降

【注】 あらかじめ、以下の URL からツールをダウンロードしてください。

- IAR Embedded Workbench for Renesas RX のダウンロードサイト :  
<https://www.iar.com/products/architectures/renesas/iar-embedded-workbench-for-renesas-rx/>
- QE for Capacitive Touch のダウンロードサイト :
- <https://www.renesas.com/jp/ja/software-tool/qe-capacitive-touch-development-assistance-tool-capacitive-touch-sensors#tab-release-information>
- RX スマート・コンフィグレータのダウンロードサイト :  
<https://www.renesas.com/jp/ja/software-tool/rx-smart-configurator#download>
- RX Driver Package のダウンロードサイト :  
<https://www.renesas.com/jp/ja/software-tool/rx-driver-package#tab-release-information>
- RX スマート・コンフィグレータ ユーザーガイド: IAREW 編  
<https://www.renesas.com/jp/ja/document/mat/rx-smart-configurator-users-guide-iarew>
- Renesas Starter Kit for RX140 ユーザーズマニュアル  
<https://www.renesas.com/jp/ja/document/mat/renesas-starter-kit-rx140-users-manual>
- RX140 グループ ユーザーズマニュアル ハードウェア編 Rev.1.10  
<https://www.renesas.com/jp/ja/document/mah/rx140-group-users-manual-hardware-rev110>

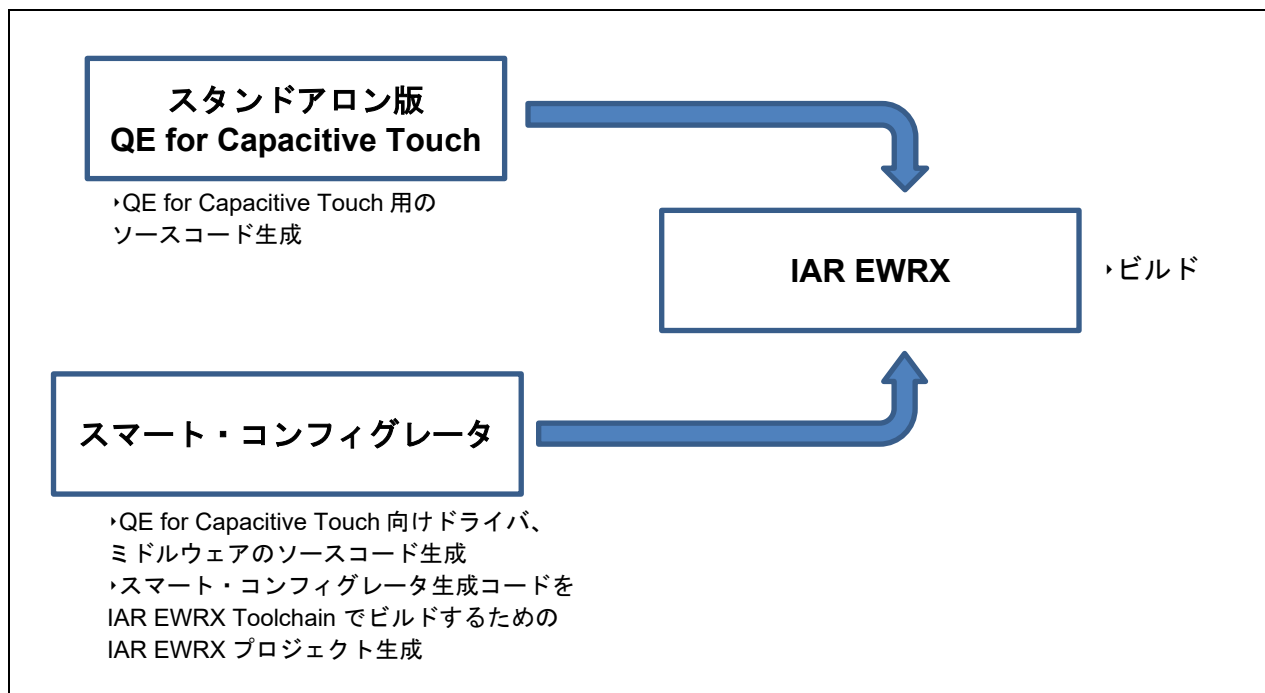


図 1 IAR EWRX で QE for Capacitive Touch を使用する場合の構成

## 目次

1.	インストール	4
1.1	IAR EWRX のインストール	4
1.2	スマート・コンフィグレータのインストール	4
1.3	スタンドアロン版 QE for Capacitive Touch のインストール	4
2.	プロジェクトの準備	4
2.1	コンフィグレーションファイルの作成	5
2.2	スマート・コンフィグレータの設定	8
2.3	コード生成	15
2.4	プロジェクトフォルダと対象マイコン設定	19
3.	タッチインタフェースの準備	20
3.1	タッチインタフェース構成の作成	20
3.2	タッチセンサ調整用ファイルの出力	22
3.3	調整用プログラムの実装	24
3.4	プロジェクトのビルド	26
4.	タッチセンサの調整	29
4.1	RSK-RX140 ボードと PC の接続	30
4.2	調整の開始	31
4.3	調整結果の取得	34
5.	実装と動作確認	36
5.1	モニタリング	36
5.2	シリアル接続の切断	39
	改訂記録	40

## 1. インストール

### 1.1 IAR EWRX のインストール

ダウンロードしたインストーラ起動後、インストーラの手順に従ってインストールします。インストールは、管理者権限で行ってください。

### 1.2 スマート・コンフィグレータのインストール

ダウンロードしたインストーラ起動後、インストーラの手順に従ってインストールします。インストールは、管理者権限で行ってください。

### 1.3 スタンドアロン版 QE for Capacitive Touch のインストール

ダウンロードした zip ファイルには、プラグイン版 QE for Capacitive Touch とスタンドアロン版 QE for Capacitive Touch が同梱されています。

- (1) ダウンロードした zip ファイルを展開します。
- (2) スタンドアロン版 QE for Capacitive Touch は、QE-CapTouch フォルダに格納されていますので、QE-CapTouch フォルダを任意のフォルダへ移動します。このとき、Windows のパス名の文字数制限 (260 文字) を超えないよう、ドライブのルートに近い場所のフォルダへ移動してください。

・移動先のフォルダ名の例 : C:\¥Renesas、C:\¥QeTouch など

なお、ツールは QE-CapTouch\¥eclipse フォルダ配下にある qe-captouch.exe より起動します。

## 2. プロジェクトの準備

スタンドアロン版 QE を利用する IAR EWRX のプロジェクトをスマート・コンフィグレータで作成します。スマート・コンフィグレータにはプロジェクト作成用に簡単なウィザードが用意されています。プロジェクトのロケーションとファイル名、プラットフォーム、ツールチェーン、プロジェクトタイプの指定、RTOS 設定、FIT モジュールロケーション設定をして、IAR EWRX プロジェクトを作成できます。以下の手順で新規 IAR EWRX プロジェクトを作成してください。

2.1 章から 2.3 章で、スタンドアロン版 QE ワークフロー図のプロジェクトの作成を行います。

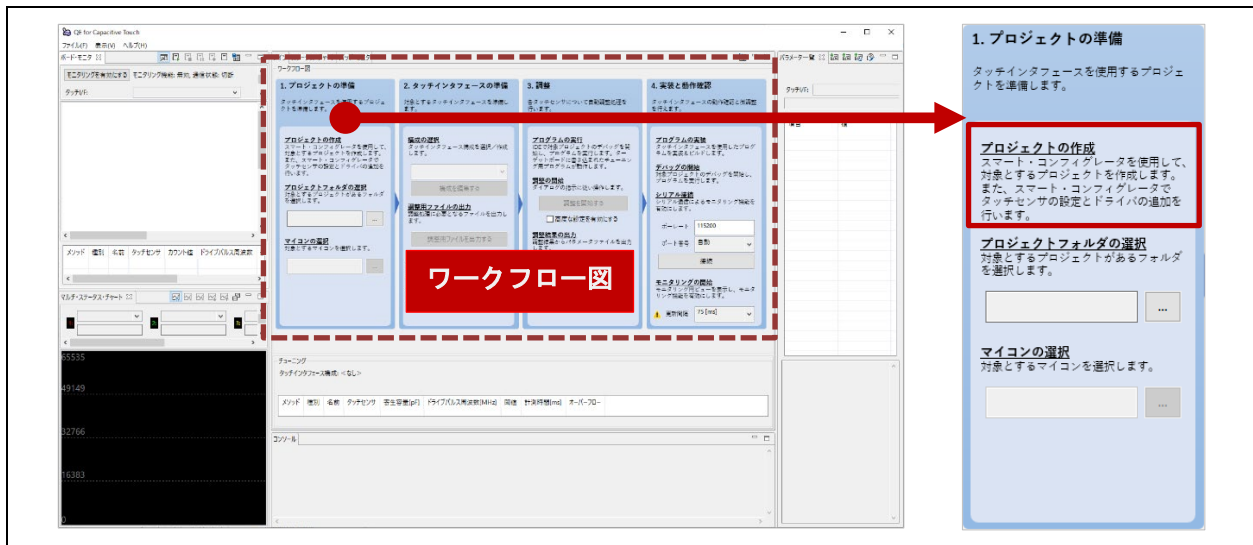


図 2-1 スタンドアロン版 QE - ワークフロー図 : プロジェクトの作成

## 2.1 コンフィグレーションファイルの作成

- (1) Windows スタートメニューから [Renesas Electronics Smart Configurator] → [Smart Configurator for RX Vx.x.x] を選択します。選択後、スマート・コンフィグレータのメインウィンドウが起動します。  
【注】 Vx.xx は、ご使用のバージョンに読み替えてください。

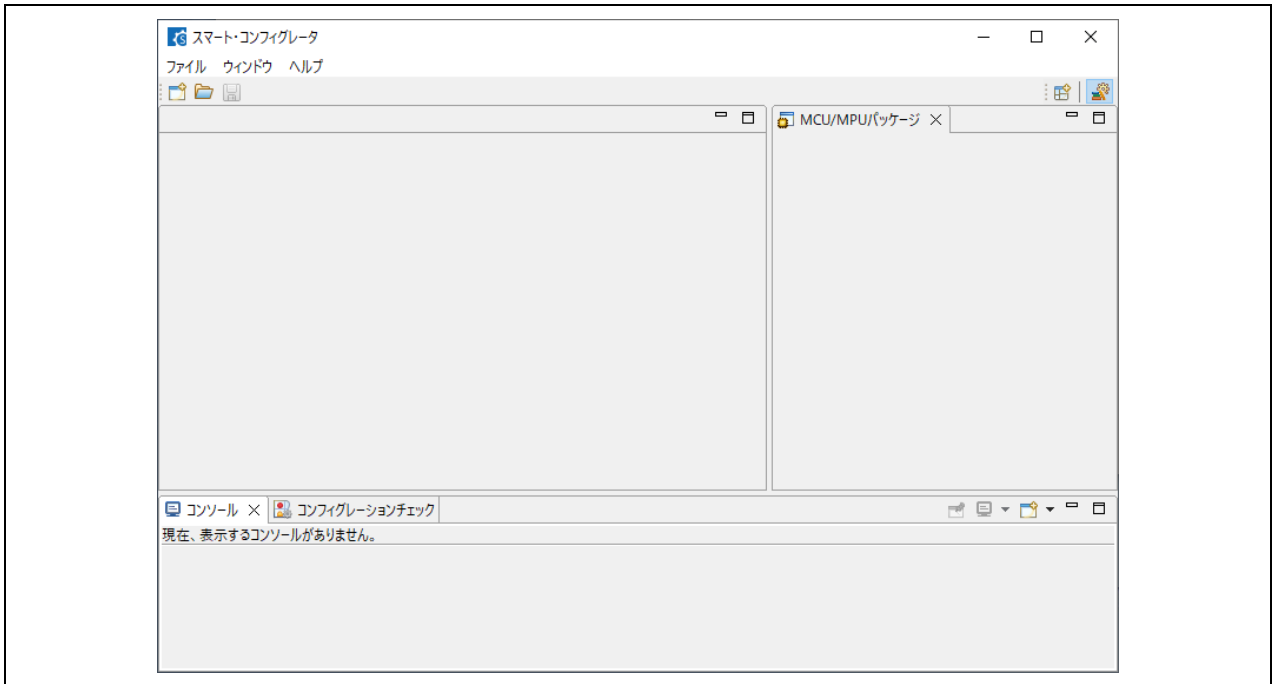



図 2-2 スマート・コンフィグレータ起動

- (2) [ファイル] メニュー → [新規...] の順に選択、またはメインツールバーの  [新規コンフィグレーションファイル] ボタンをクリックします。

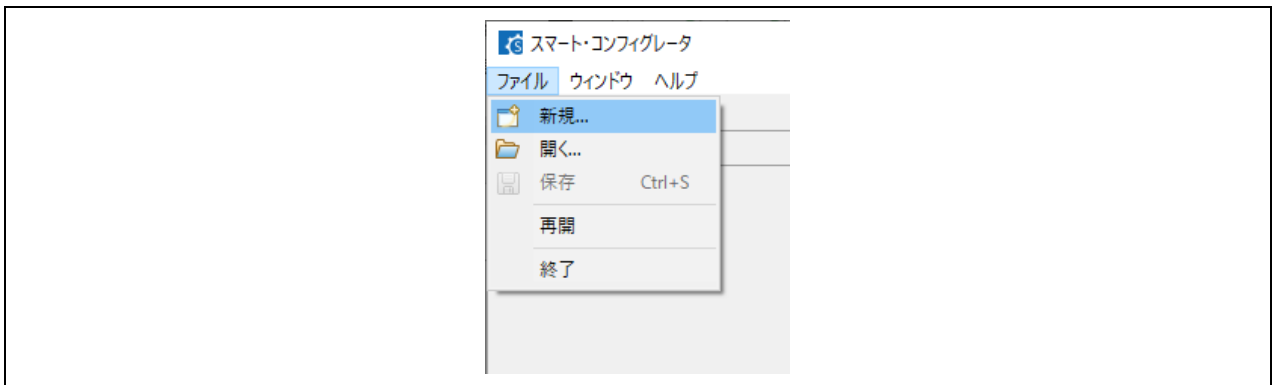


図 2-3 新規スマート・コンフィグレーションファイル作成

(3) スマート・コンフィグレータ設定のダイアログが表示されます。

- 次の各項目について、設定します。
  - ① [プラットフォーム:] RX140 を入力し、フィルタ後に表示される Board 配下の"RSKRX140 (R5F51406BxFN)"を選択します。
  - ② [ツールチェーン:] [IAR EWRX Toolchain]を選択します。
  - ③ [ファイル名:] 作成するプロジェクト名を入力します。
  - ④ [ロケーション:] プロジェクト作成フォルダを確認します。プロジェクト作成フォルダを変更したい場合は、[参照...]をクリックして保存先を選択します。
- [次へ(N) >]をクリックし、続けて FIT モジュールとミドルウェアを使用するための設定をします。

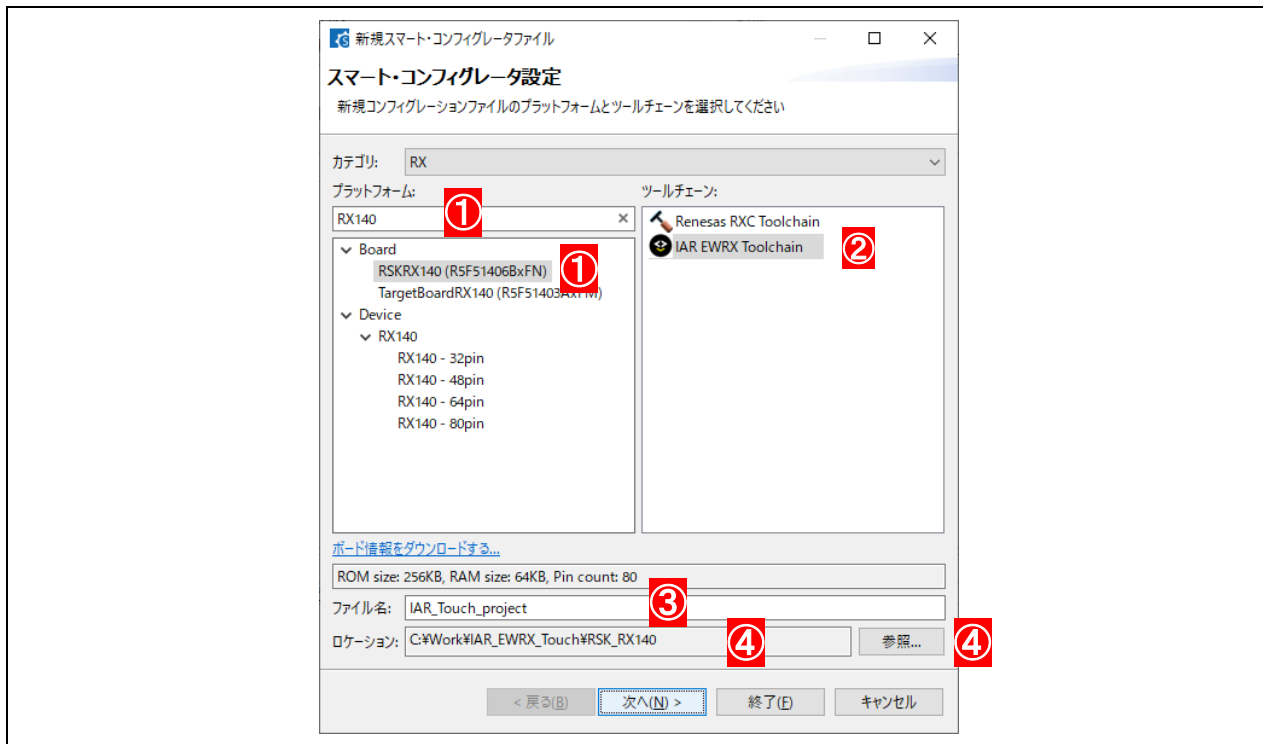


図 2-4 スマート・コンフィグレータ設定 (1)

- [RTOS 設定]の[RTOS:] None を確認し、FIT module location を設定します。  
ダウンロードした FIT モジュール zip ファイルを解凍します。  
[Browse...]をクリックし、解凍した FIT Modules フォルダを選択します。

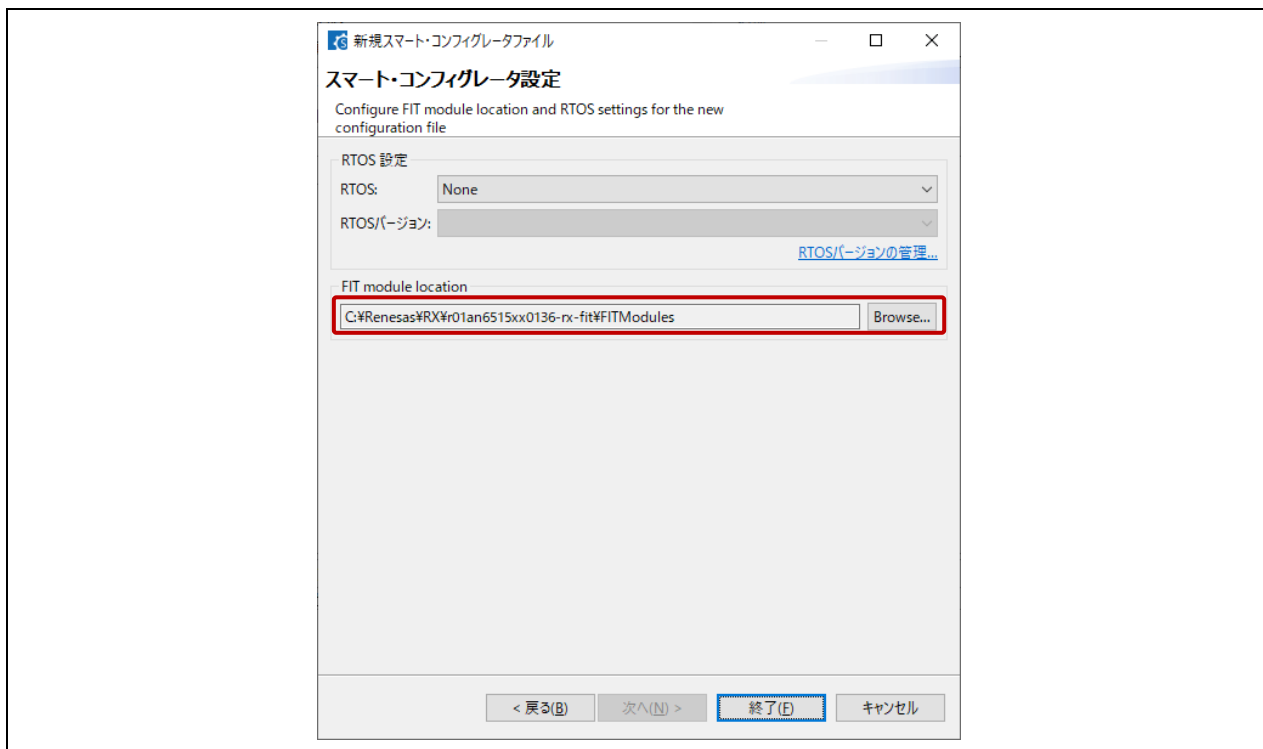


図 2-5 スマート・コンフィグレート設定 (2)

- [終了(E)]をクリックすると、コンフィグレーションファイル(\*.scfg)が作成されて、メインウィンドウが表示されます。

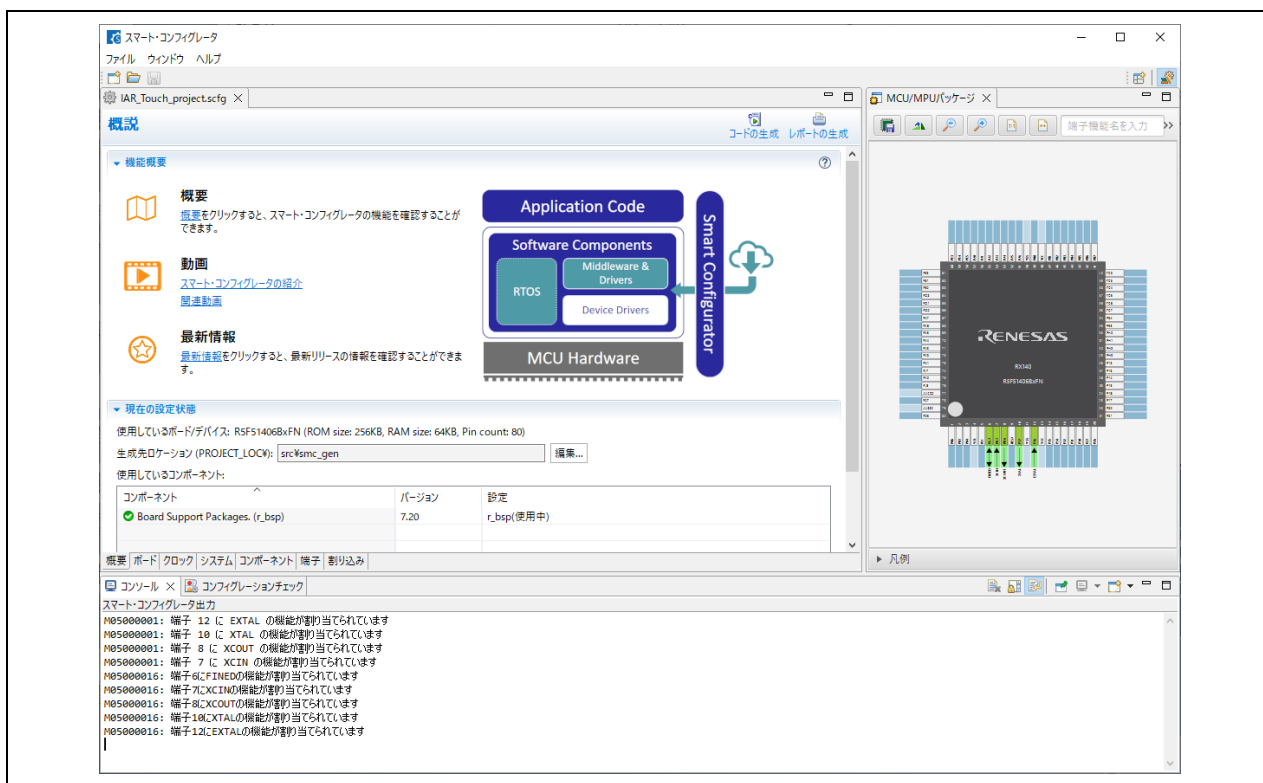


図 2-6 スマート・コンフィグレート メインウィンドウ

## 2.2 スマート・コンフィグレータの設定

スマート・コンフィグレータビューの下部のタブを選択すると表示内容が切り替わります。必要に応じてタブを選択して設定します。

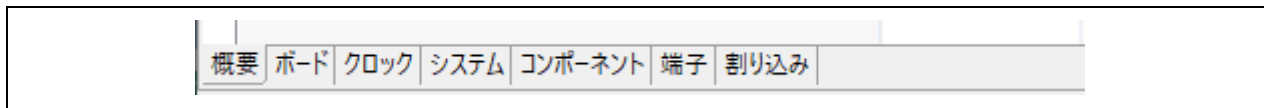


図 2-7 スマート・コンフィグレータビューのタブ

### (1) クロックの設定

[クロック]タブを選択します。

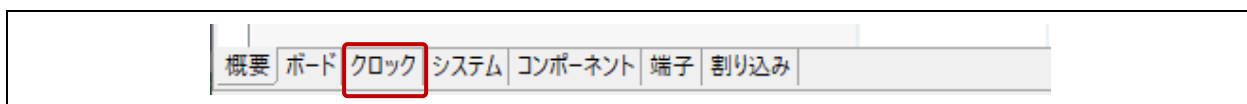


図 2-8 [クロック]タブの選択

- VCC: “5.0”を入力します。
- メインクロック: ✓をつけます。  
FCLK, ICLK, PCLKD が 48MHz、PCLKB が 24MHz となるように図 2-9 の通りに設定します。

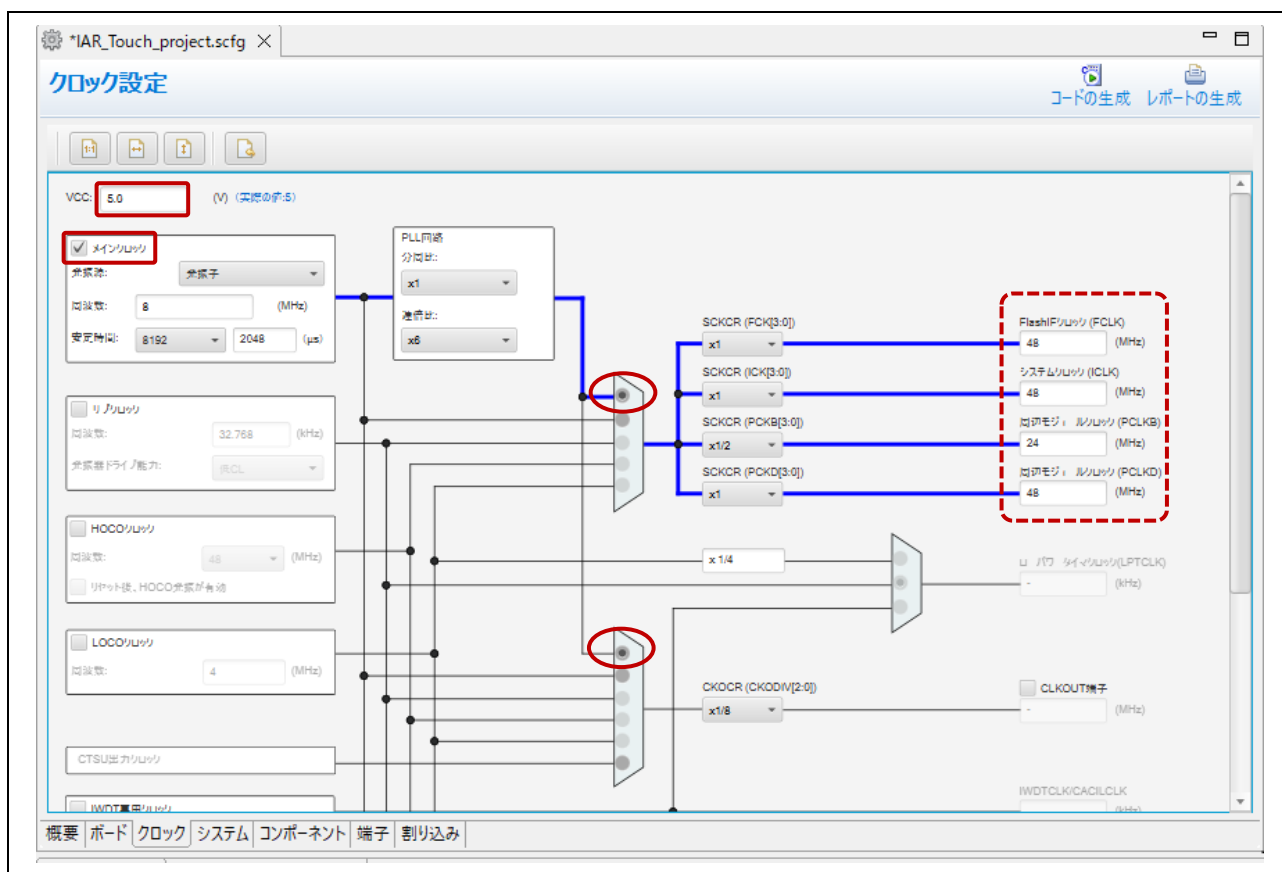


図 2-9 クロック設定



## (2) システム設定

[システム]タブを選択します。

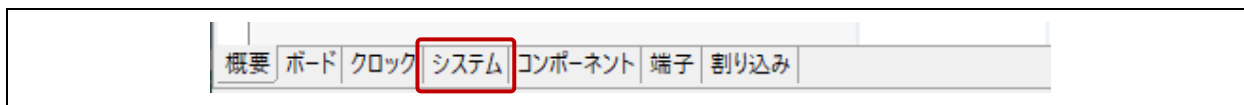


図 2-10 [システム]タブの選択

- [デバッグインタフェース設定]に"FINE"が選択されていることを確認します。

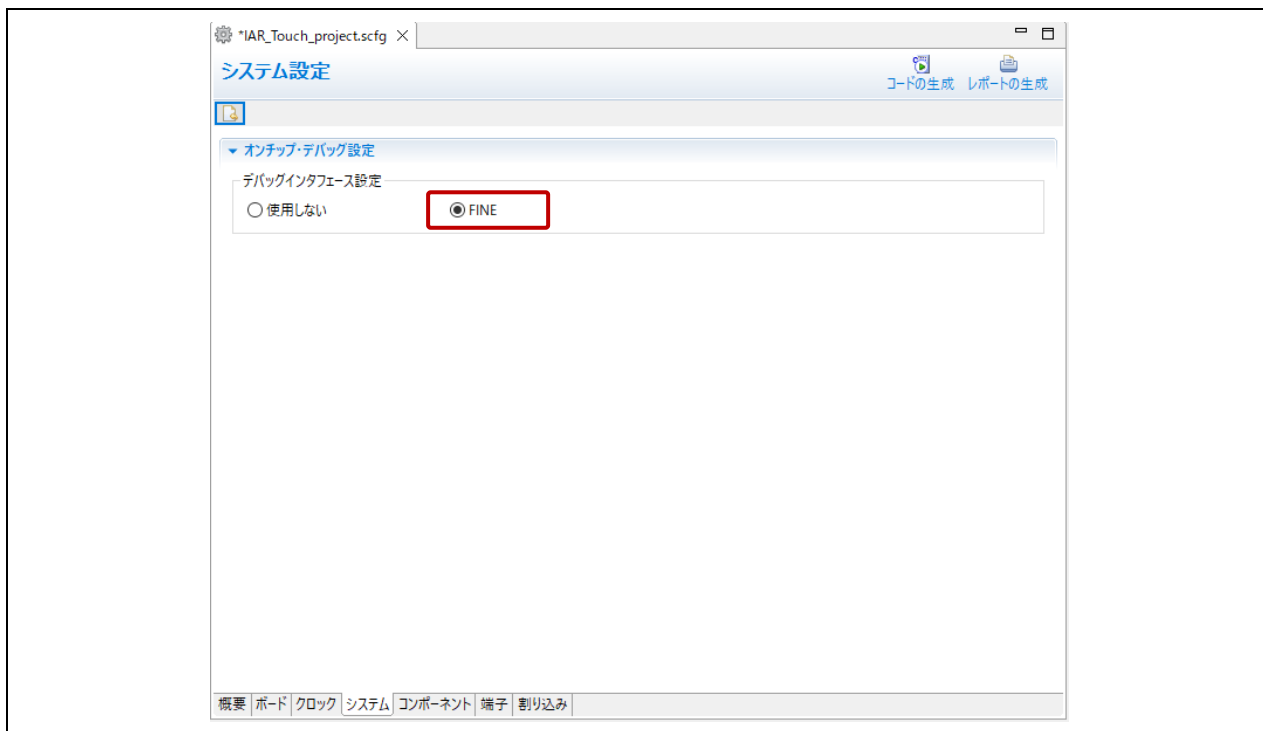


図 2-11 システム設定

(3) ソフトウェアコンポーネント設定

[コンポーネント]タブを選択します。



図 2-12 [コンポーネント]タブの選択

- ドライバ、ミドルウェア、およびシリアルインタフェースを追加して、設定をします。

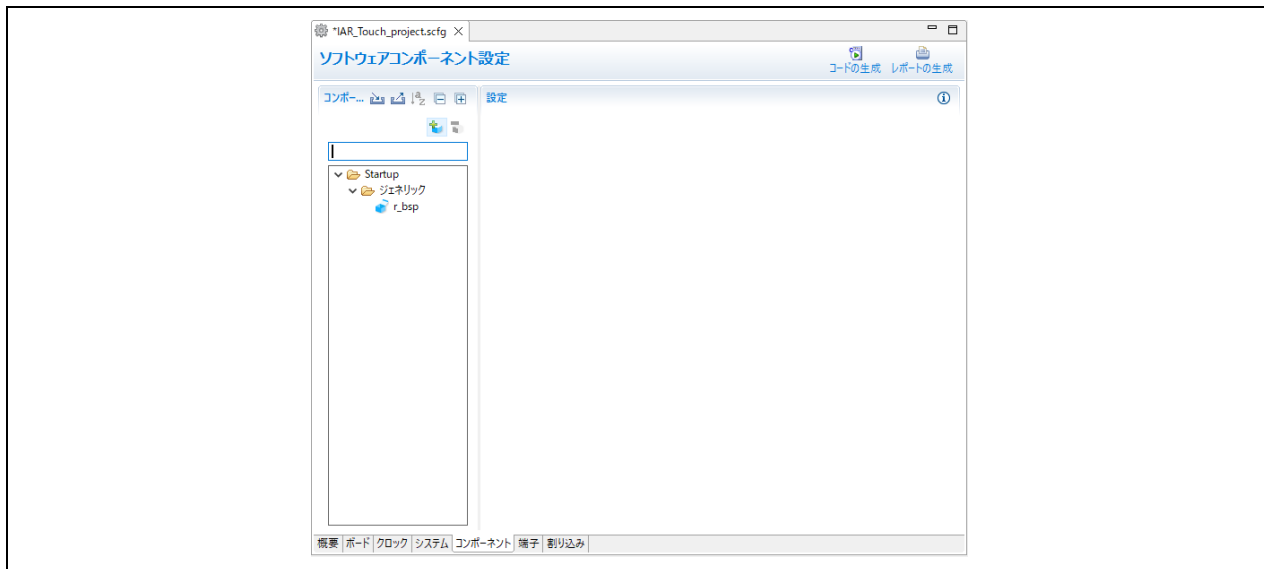



図 2-13 ソフトウェアコンポーネント設定

- コンポーネントの追加は、 [コンポーネントの追加]ボタンをクリックして、“コンポーネントの追加”ダイアログを開き、追加するコンポーネントを選択します。

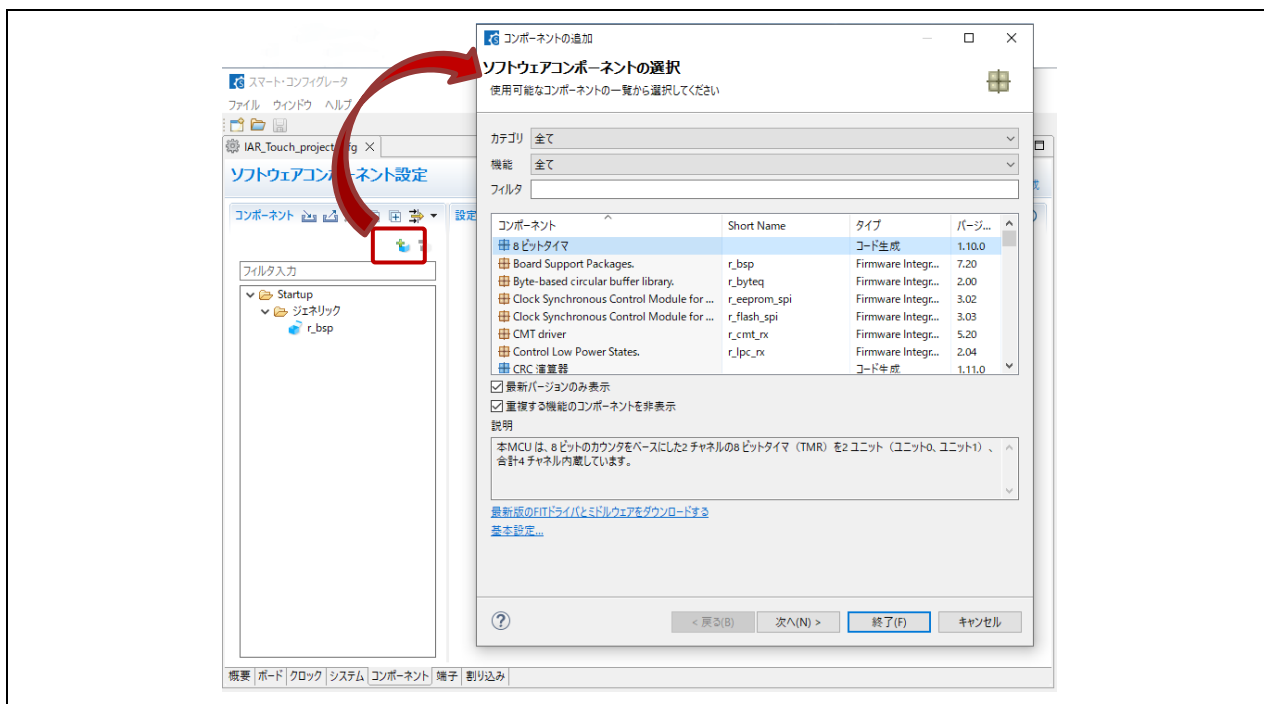


図 2-14 “コンポーネント追加”ダイアログの表示

## a. CTSU 用ドライバの追加

“コンポーネントの追加”ダイアログを開き、“r\_ctsu\_qe”を選択し、[終了(E)]をクリックします。

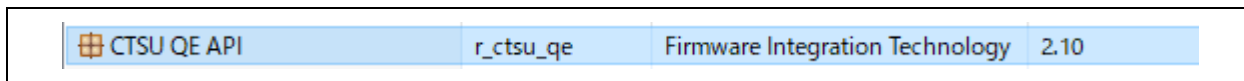


図 2-15 “r\_ctsu\_qe”の選択

- 追加した “r\_ctsu\_qe” をクリックし、TSCAP 端子と使用する 2 つの TS 端子を有効にします。TS 端子とタッチセンサの割り当ては、使用するターゲットボードのユーザーズマニュアルをご確認ください。

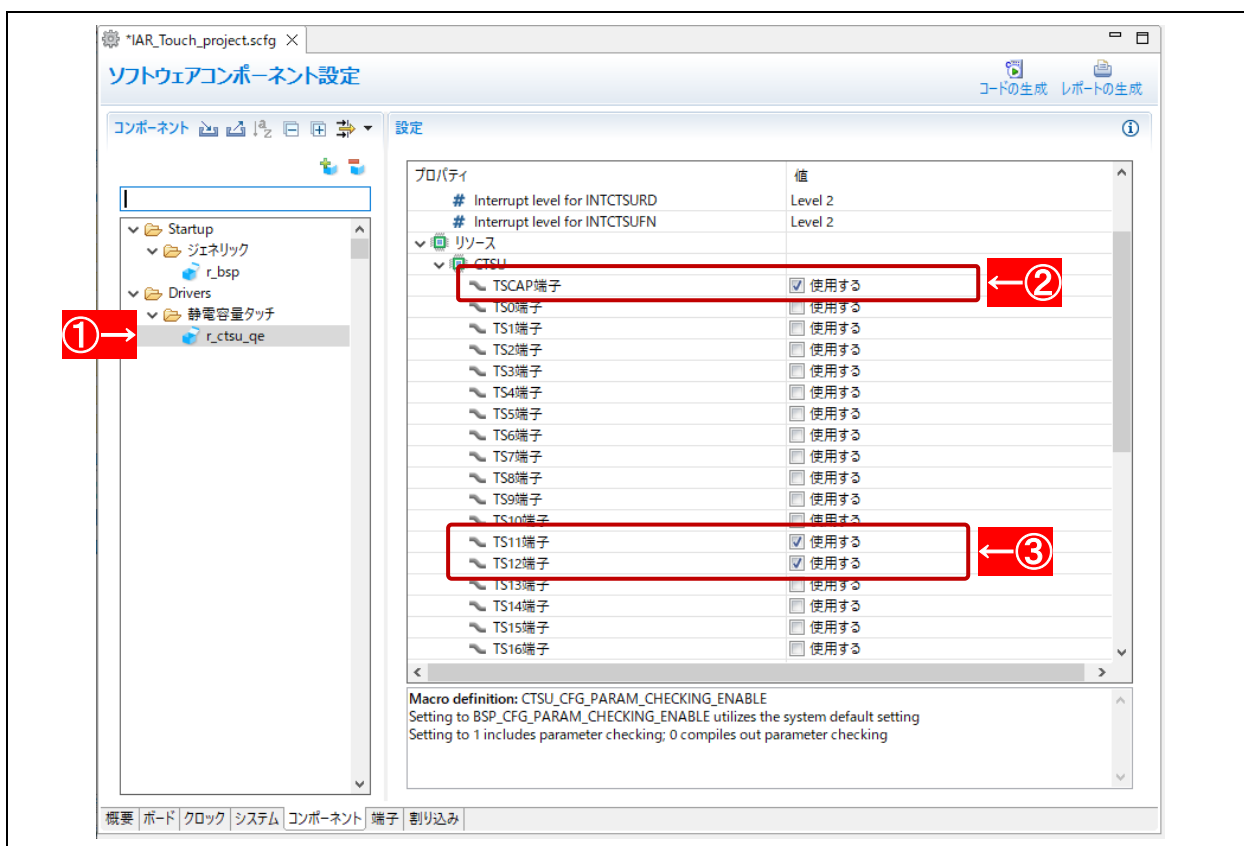


図 2-16 使用する端子の有効化 (①: r\_ctsu\_qe, ②: TSCAP 端子, ③: 2 つの TS 端子)

b. タッチ用ミドルウェアの追加

“コンポーネントの追加”ダイアログを開き、“rm\_touch\_qe”を選択し、[終了(E)]をクリックします。

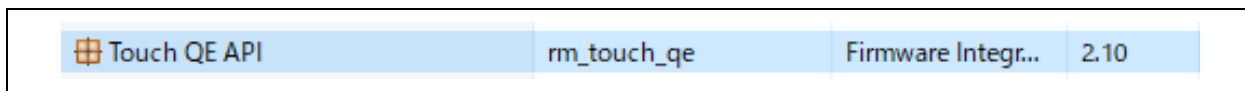


図 2-17 "rm\_touch\_qe"の選択

- 追加した"rm\_touch\_qe"をクリックし、UART を使用したシリアルモニタ通信の設定をします。以下のように"値"を選択します。

```
# Support for QE monitoring using UART : Include code to update sensor data for monitor
# Support for Serial tuning using UART : Include code to serial tuning
# UART channel : UART1
```

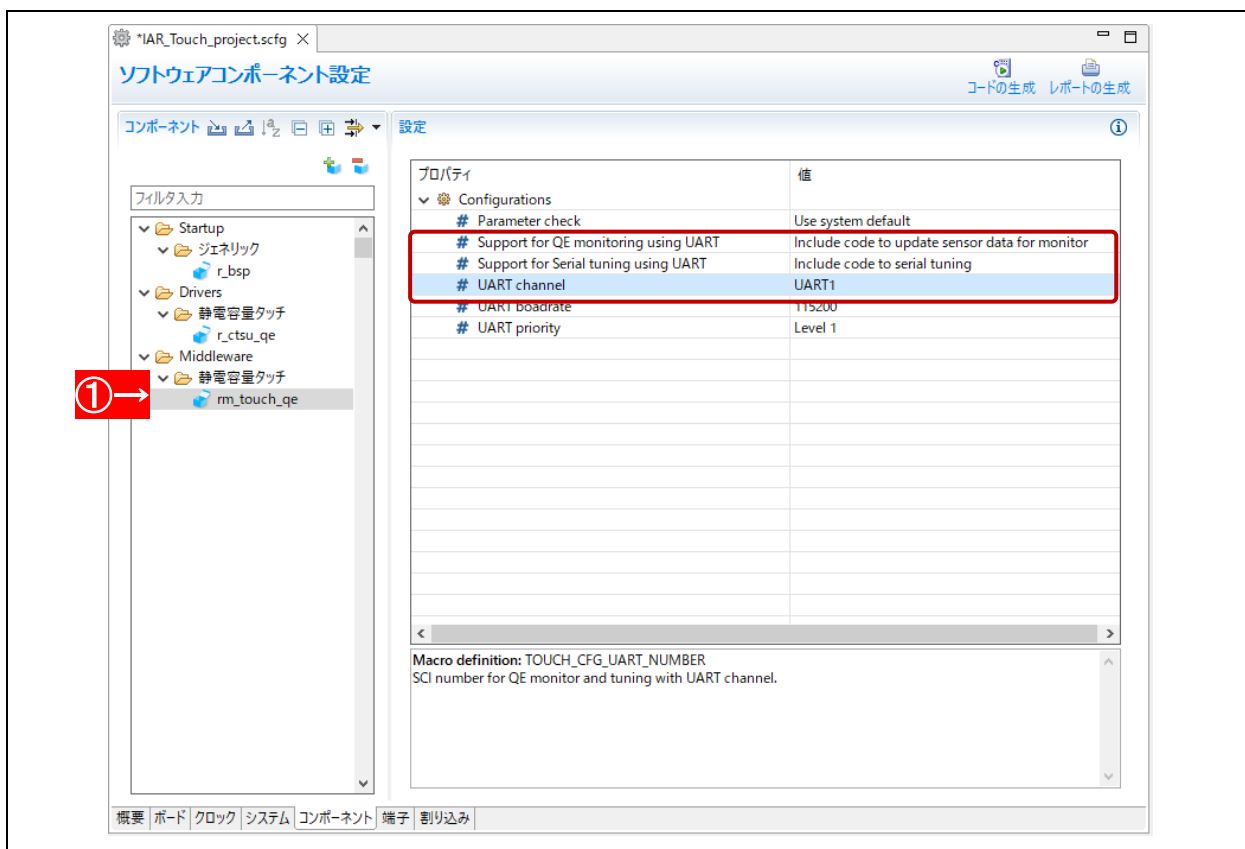


図 2-18 "rm\_touch\_qe"のプロパティ選択 (①: rm\_touch\_qe)

c. シリアルインタフェースの追加

“コンポーネントの追加”ダイアログを開き、“r\_sci\_rx”を選択し、[終了(E)]をクリックします。

SCI Driver	r_sci_rx	Firmware Integration Technology	4.40
------------	----------	---------------------------------	------

図 2-19 ” r\_sci\_rx”の選択

- 追加した” r\_sci\_rx”をクリックし、関連するポートを設定パネルに表示して、以下のように”値”を設定します。

# Use ASYNC mode : Include  
 # Include software support for channel 1 : Include  
 # Transmit end interrupt : Enable

リソース > SCI > SCI1 : ✓をつける。  
 RXD1, TXD1 : “使用する”に✓をつける。

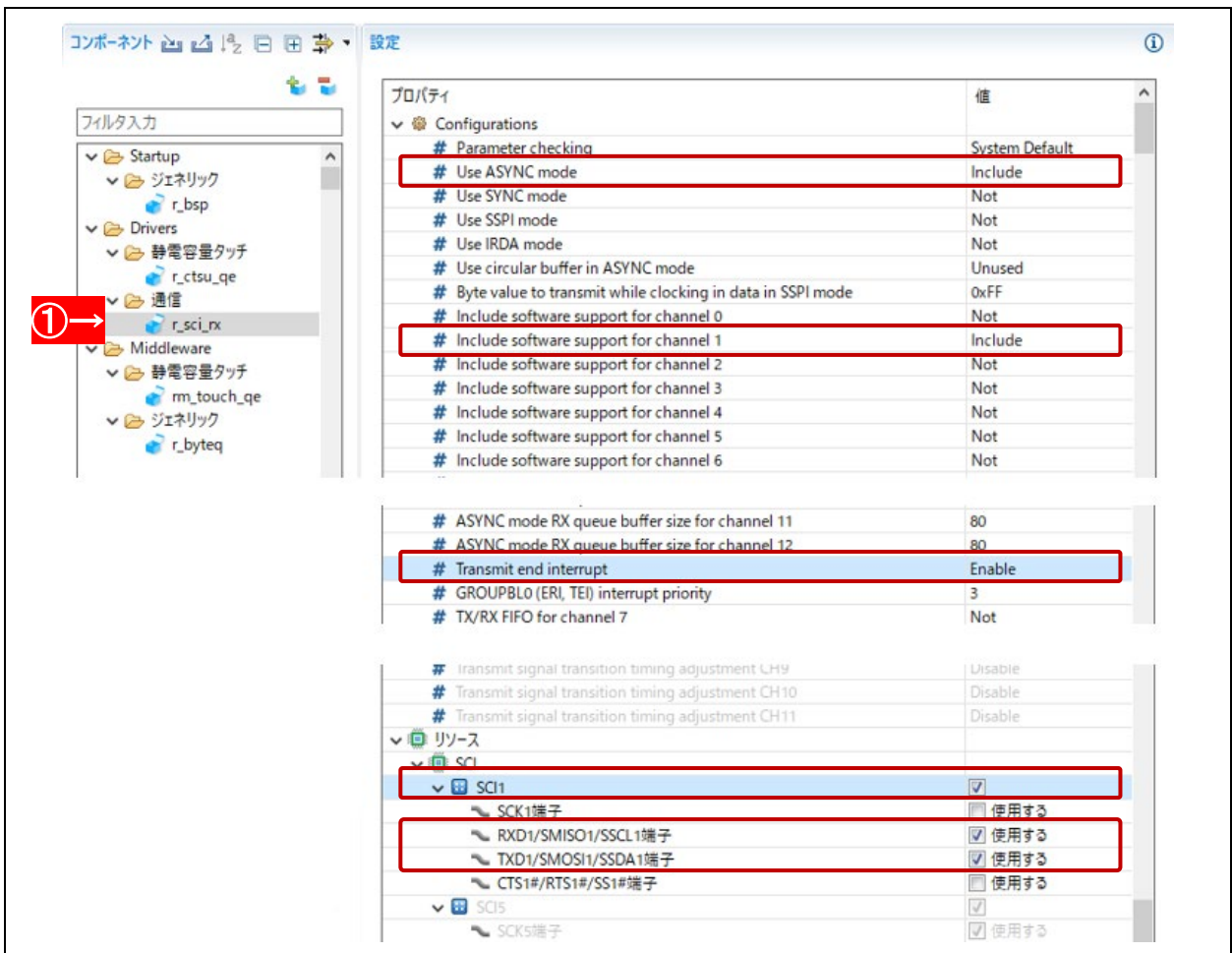


図 2-20 ” r\_sci\_rx”の設定 (①: r\_sci\_rx)

- SCI1 の端子機能の割り当てをします。

[端子]タブを選択します。

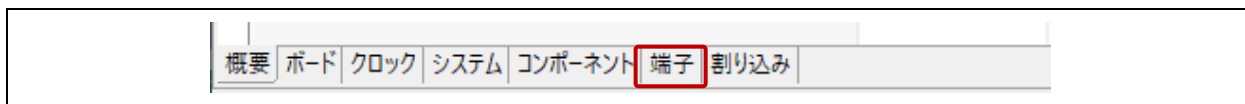


図 2-21 [端子]タブの選択

- [端子]ページの[端子機能]タブのツリーでシリアルコミュニケーションインタフェースを展開し、SCI1 を選択します。

“RXD1”機能に“P30”、“TXD1”機能に“P26”を割り当てます

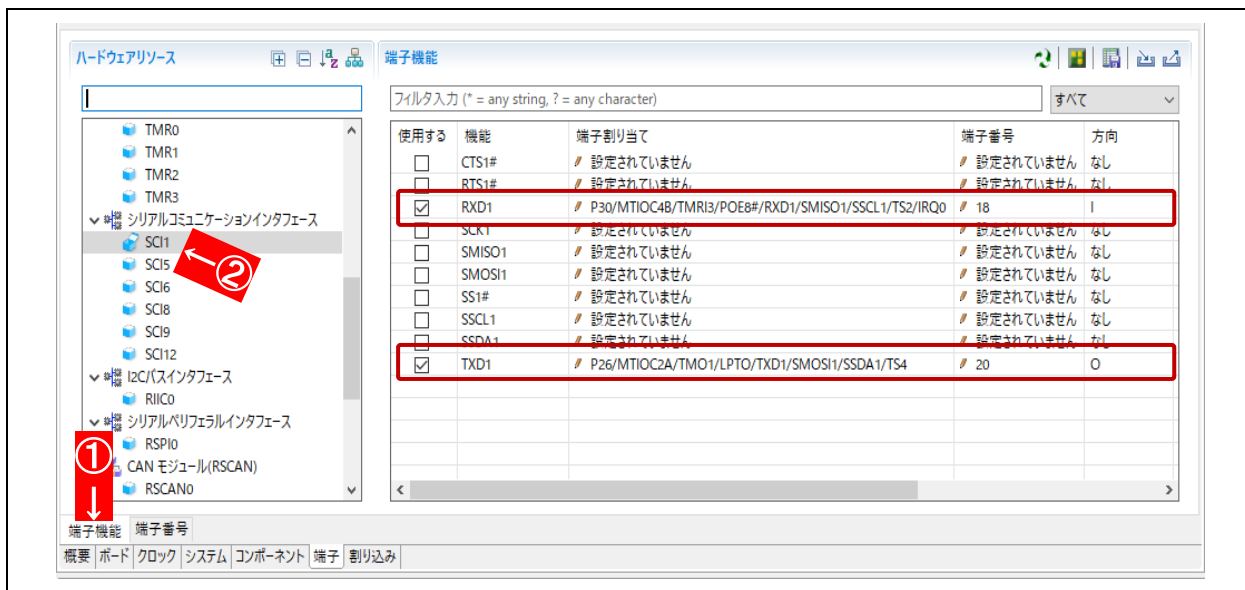



図 2-22 SCI1 機能の端子割り当て(①: [端子機能]タブ, ②: SCI1)

## 2.3 コード生成

コード生成を行います。

- (1) スマート・コンフィグレータの  ボタンをクリックし、コードの生成を行います。

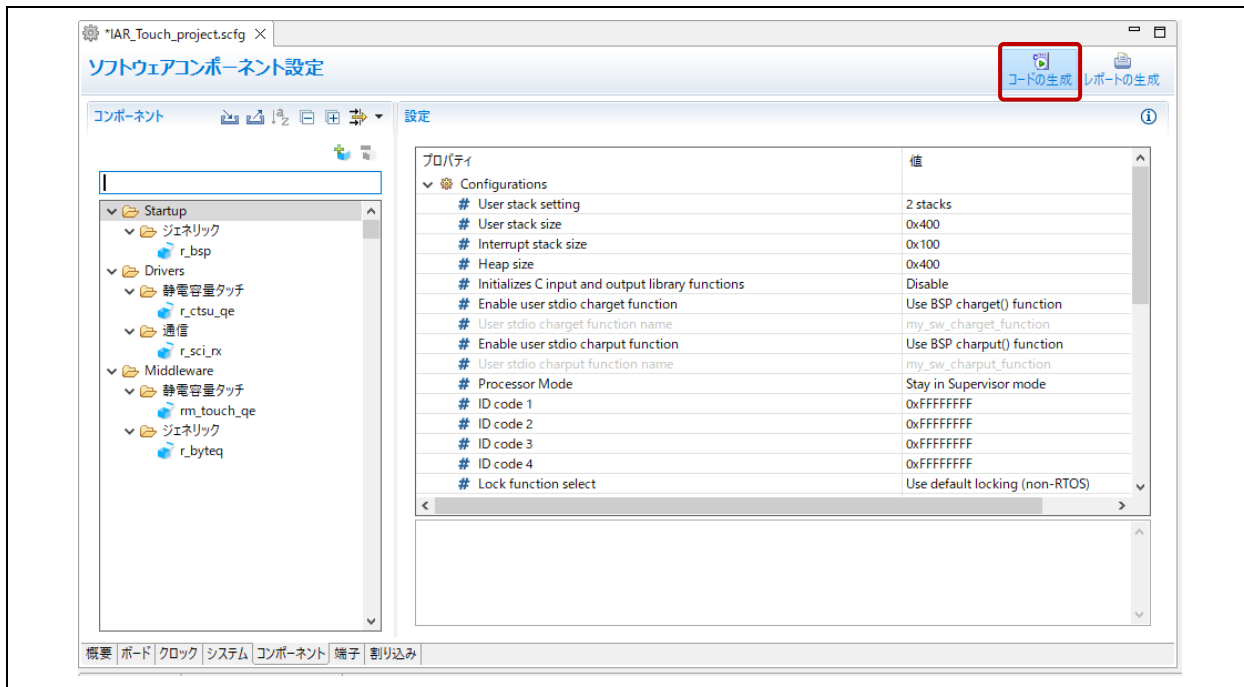


図 2-23 コード生成

次のメッセージボックスが表示された場合は、[続行(P)]をクリックします。

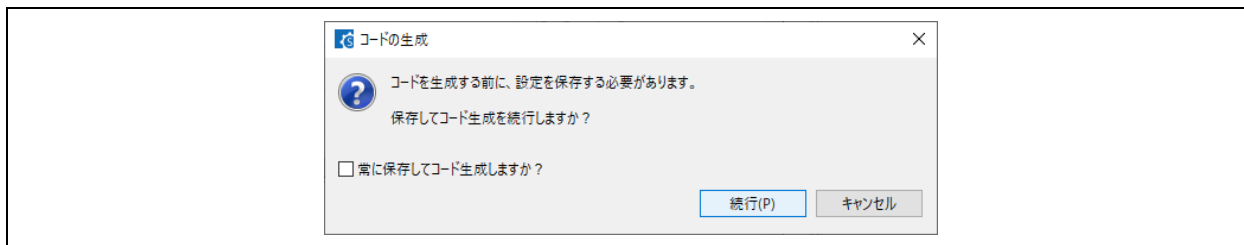


図 2-24 コード生成時のメッセージ

- (2) コンソールビューの "コード生成の終了"を確認します。

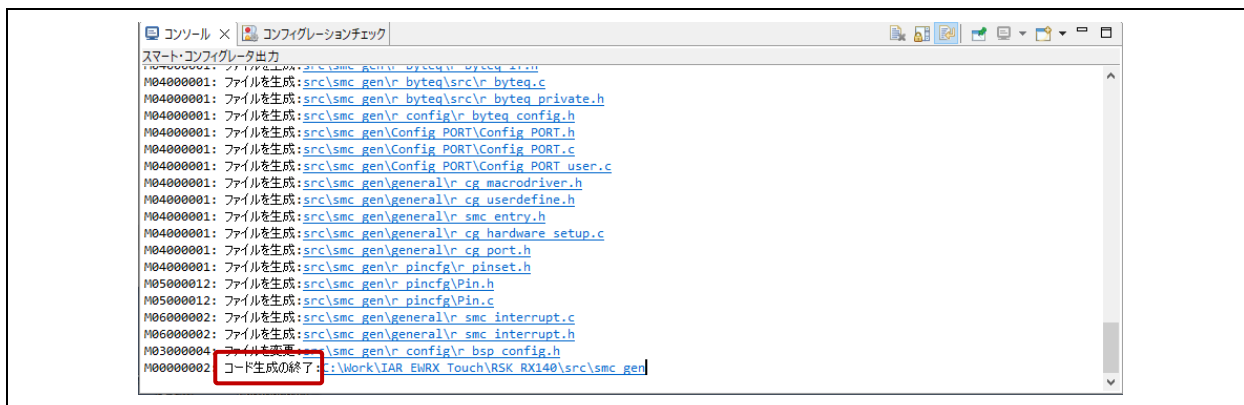


図 2-25 コード生成の終了

- (3) スマート・コンフィグレータを終了します。[ファイル]メニュー→[終了]の順に選択します。

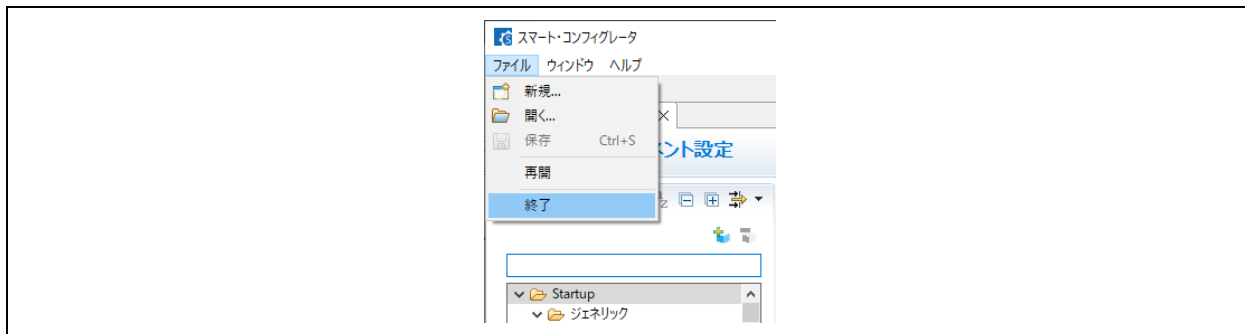



図 2-26 スマート・コンフィグレータの終了

- (4) Windows エクスプローラで 2.1.(3) 章でロケーション指定したフォルダに、 アイコンの IAR IDE Workspace ファイル(\*.eww)が作成されていることを確認します。

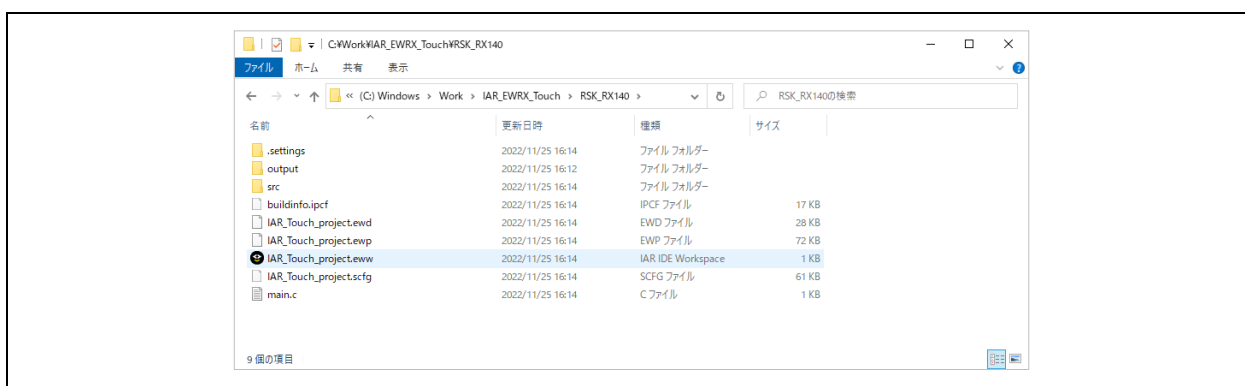


図 2-27 IAR EWRX ワークスペースフォルダ

- (5) IAR EWRX ウィンドウの[ツール(T)]メニューから、スマート・コンフィグレータを起動するための設定をします。

Windows エクスプローラで作成された IAR IDE Workspace ファイルをダブルクリックして IDE を起動します。IAR EWRX ウィンドウが開いたら、[ツール(T)]メニュー→[ツールの設定(C) ...]を順に選択し、[ツールの設定]ダイアログを開きます。

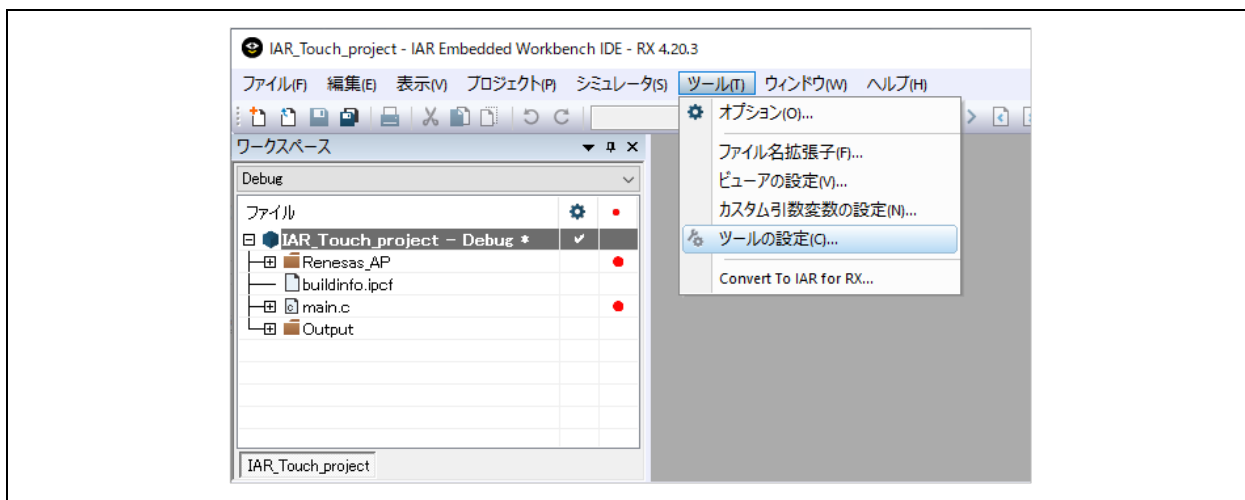


図 2-28 ツール



- [ツールの設定] ダイアログで、[新規作成(N)]をクリックして、以下の②から⑤を入力します。
  - ① メニュー内容(M):      メニューテキストの入力内容が表示されます。
  - ② メニューテキスト(T): Smart Configurator for RX
  - ③ コマンド(C):           [インストール先フォルダ]¥eclipse¥SmartConfigurator.exe
  - ④ 引数(A):               \$PROJ\_FNAME\$.scfg
  - ⑤ 初期ディレクトリ(D): \$PROJ\_DIR\$

[OK]をクリックして、“ツールの設定”画面を閉じます。



図 2-29 ツールの設定

- [ツール(T)]メニューで、スマート・コンフィグレータの選択ができるようになります。

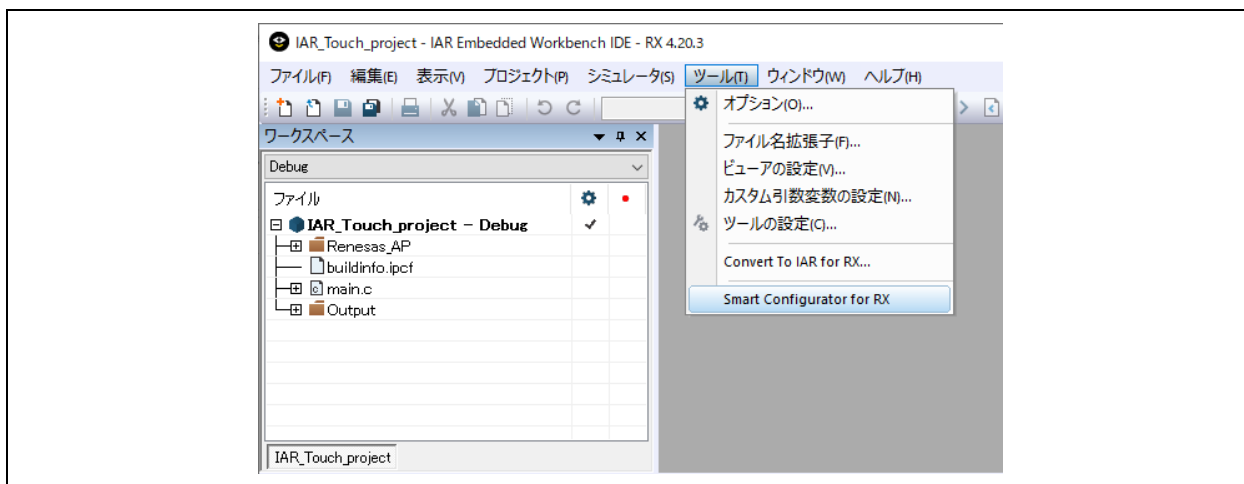


図 2-30 [ツール(T)]メニュー設定の反映

- (6) 次に[ワークスペース]ウィンドウでプロジェクト名を右クリックし、[すべてを再ビルド(B)] を選択して、ビルドします。ビルド終了後に[ビルド]ウィンドウまたはメインウィンドウ下部のステータスバー表示でエラーがないことを確認してください。

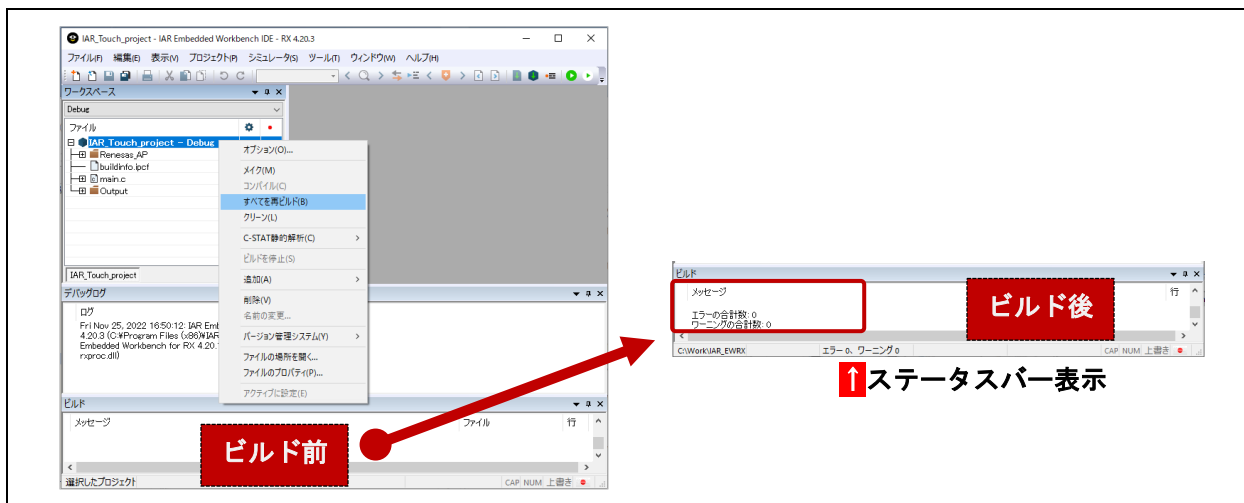


図 2-31 ビルド

- (7) [ファイル(F)]メニュー → [終了(X)]を選択します。[IarIdePm]ダイアログ表示されるので、[はい(Y)]をクリックして IAR EWRX ワークスペースを終了します。

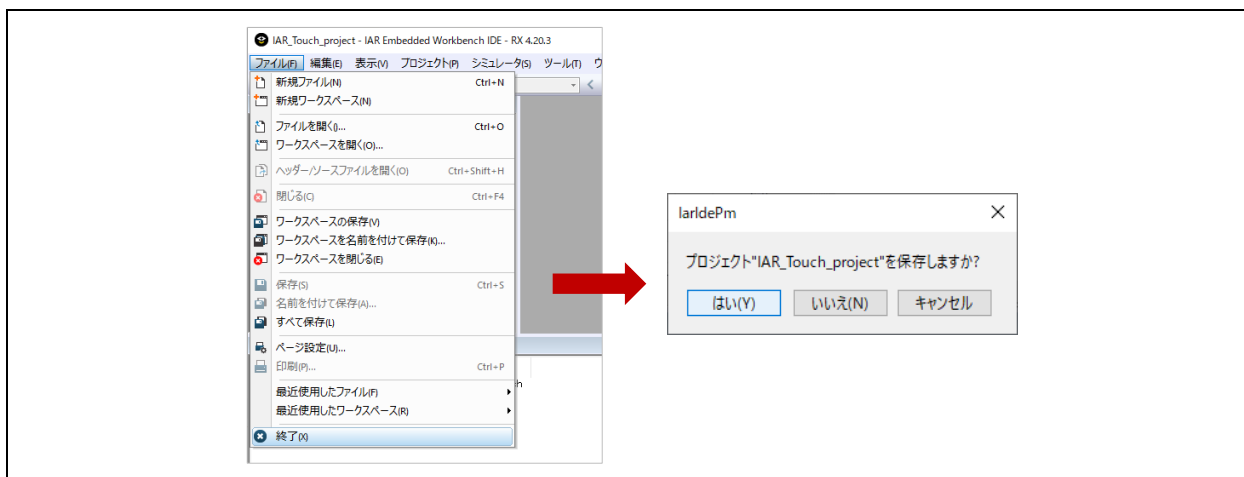


図 2-32 IAR EWRX ワークスペースの終了

## 2.4 プロジェクトフォルダと対象マイコン設定

プロジェクトの作成を終えたら、スタンドアロン版 QE を起動します。ワークフロー図の "1. プロジェクトの準備" に従い、作成したプロジェクトのフォルダとマイコン製品情報を設定します。

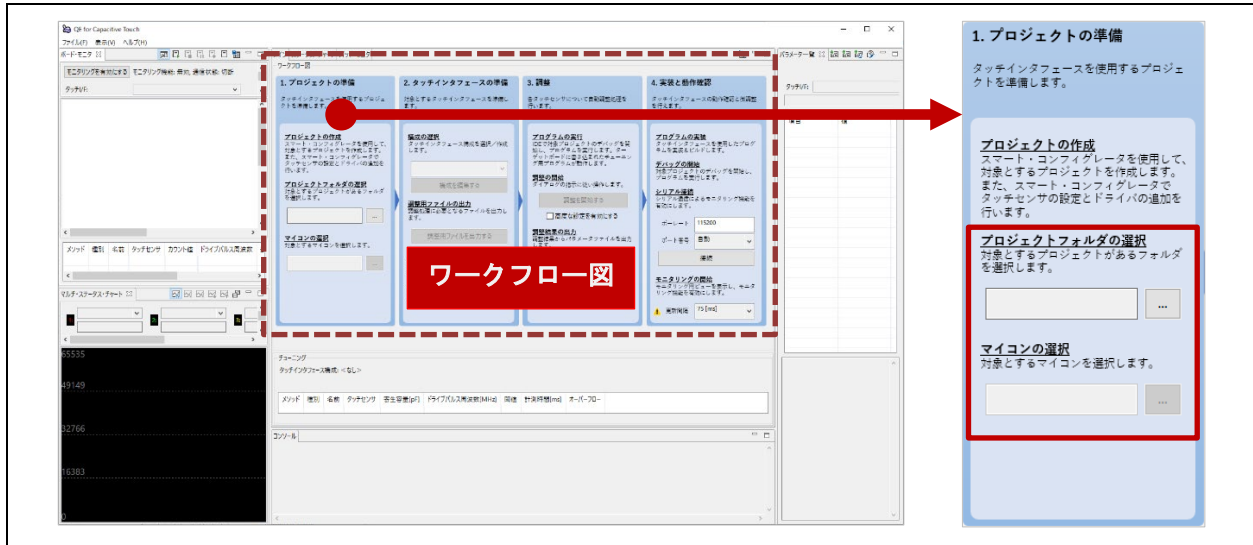


図 2-33 スタンドアロン版 QE - ワークフロー図 :Step1 プロジェクトの準備

(1) プロジェクトフォルダを選択します。

“プロジェクトフォルダの選択”の入力領域右の[...]をクリックして、プロジェクトを格納しているフォルダを選択します。

(2) マイコンの製品情報を選択します。

“マイコンの選択”の入力領域右の[...]をクリックすると、“対象マイコンの製品名”ダイアログが表示されるので、▼をクリックして選択します。

- ① ファミリ名 RX を選択します。
- ② グループ名 RX140 を選択します。
- ③ ピン数 LQFP/80pin:FN を選択します。
- ④ ROM 容量 256KB:6 を選択します。
- ⑤ 製品名 上記の①から④を正しく選択すると、R5F51406xxFN が表示されます。

[OK]をクリックして、閉じます。

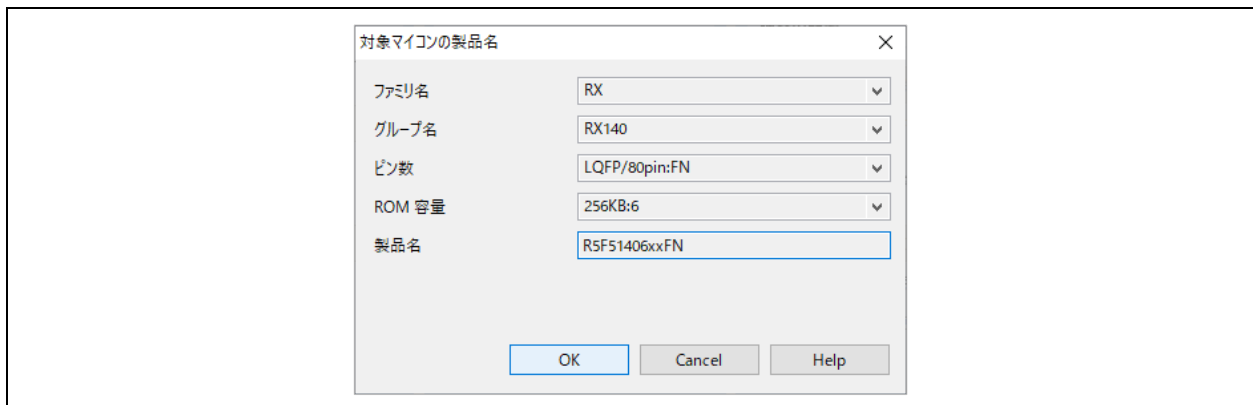


図 2-34 対象マイコンの製品名

### 3. タッチインタフェースの準備

ワークフロー図の“2. タッチインタフェースの準備”に従い、設定します。

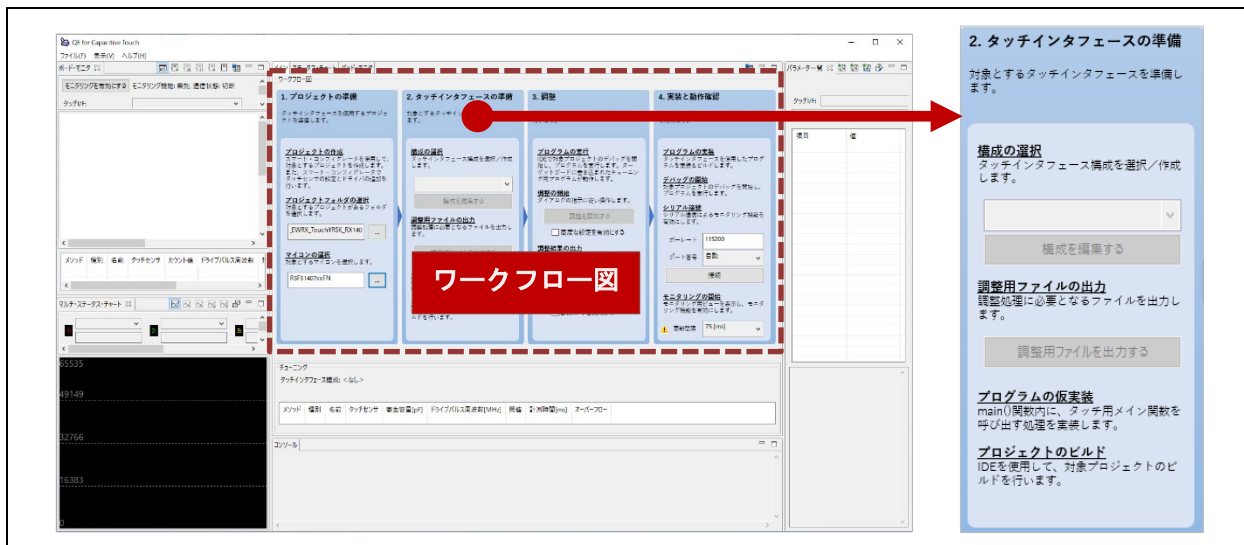


図 3-1 スタンドオン版 QE - ワークフロー図 :Step2 タッチインタフェースの準備

#### 3.1 タッチインタフェース構成の作成

タッチインタフェース構成を新規作成します。

- (1) “構成の選択”の  をクリックし、“タッチインタフェース構成の新規作成”を選択します。“タッチインタフェース構成の作成”ウィンドウが開き、タッチインタフェースを配置する領域が表示されます。

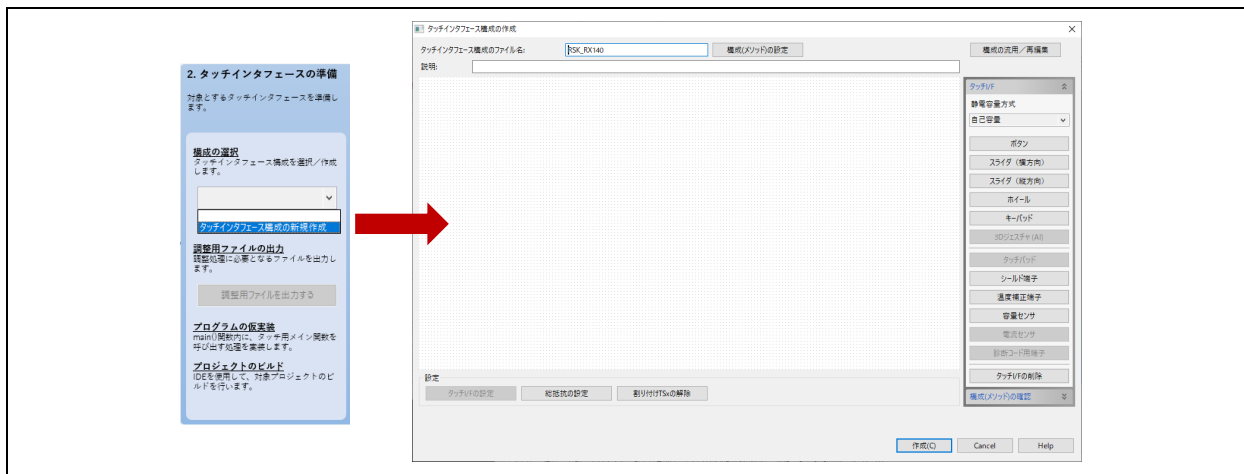


図 3-2 タッチインタフェース構成の作成

- (2) “タッチインタフェース構成の作成”ウィンドウ右側の“タッチ I/F”パネルから[ボタン]をクリックすると、カーソルがボタンを配置できる状態になり、配置領域でクリックするとボタンを配置できます。以下のように2つのボタンを配置し、[ESC]キーを押して、タッチインタフェースの追加を終了します。

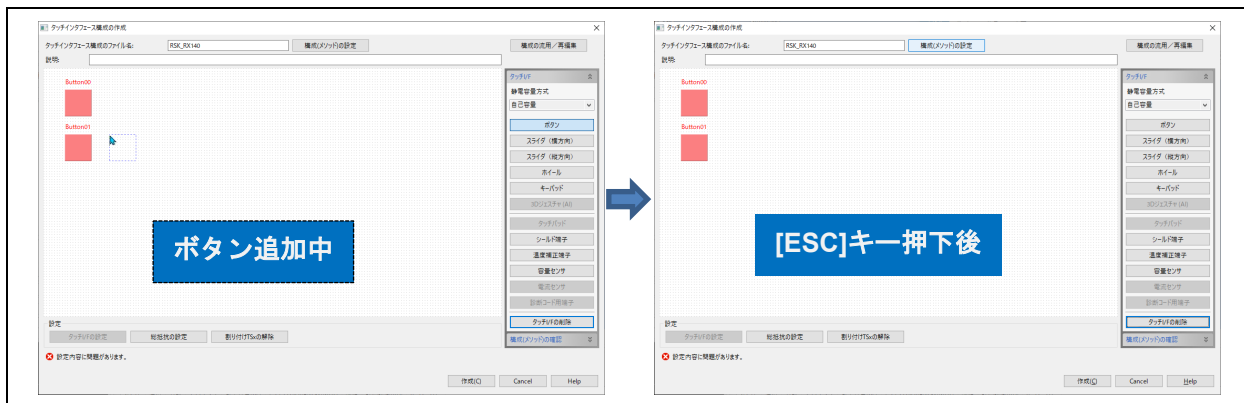


図 3-3 ボタンの追加

- (3) 配置したボタンをダブルクリックし、表示された“タッチインタフェースの設定”ダイアログで、以下のように設定します。

Button00 : [名前]TS\_B1 [タッチセンサ]TS11 [抵抗値]560

Button01 : [名前]TS\_B2 [タッチセンサ]TS12 [抵抗値]560

タッチインタフェースの設定が正しく設定されるとボタンの色が緑色に変わります。

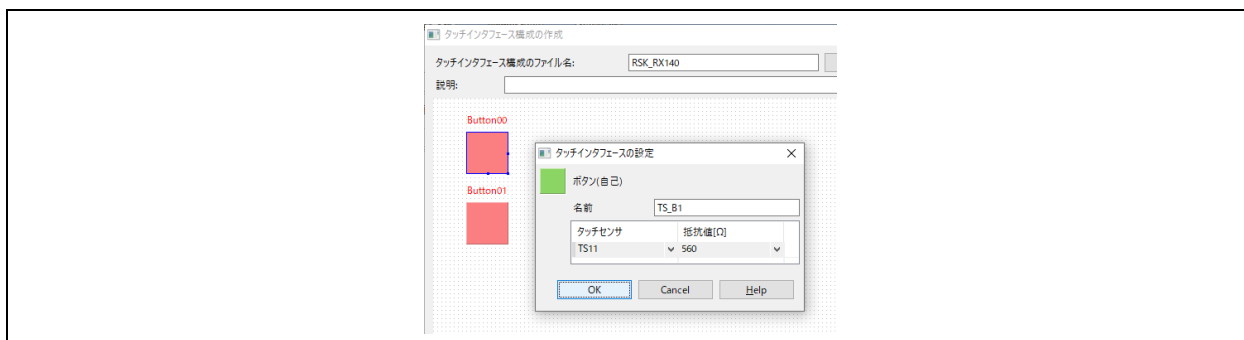


図 3-4 button00 のタッチインタフェースの設定

- (4) 2つの配置したボタンが緑色あることを確認し、[作成]をクリックして閉じます。

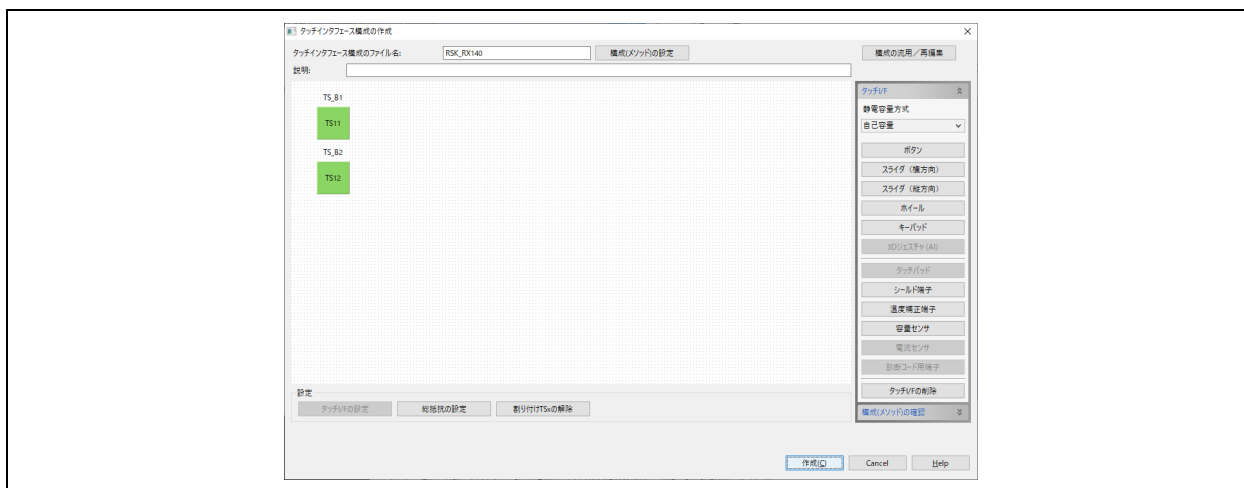


図 3-5 タッチインタフェース構成の作成の終了

(5) “メイン”の“チューニング”パネルにタッチインタフェースの構成が表示されます。

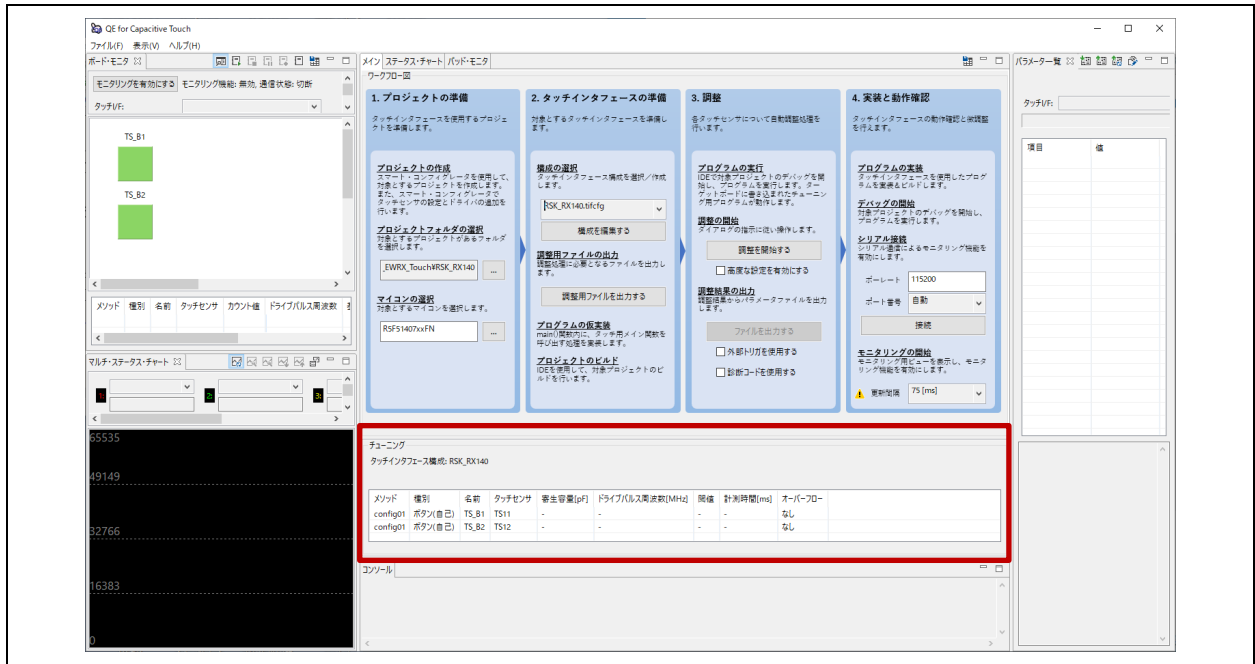


図 3-6 タッチインタフェースの構成

### 3.2 タッチセンサ調整用ファイルの出力

指定したフォルダにタッチセンサの調整用ファイルの出力後にマイコンの周辺モジュールクロックの周波数、マイコンへの供給電圧の設定をします。

- (1) [調整用ファイルを出力する]をクリックします。“パラメータファイルの出力先”ダイアログが表示されるので出力先のフォルダをクリックして、[フォルダの選択]をクリックして設定します。

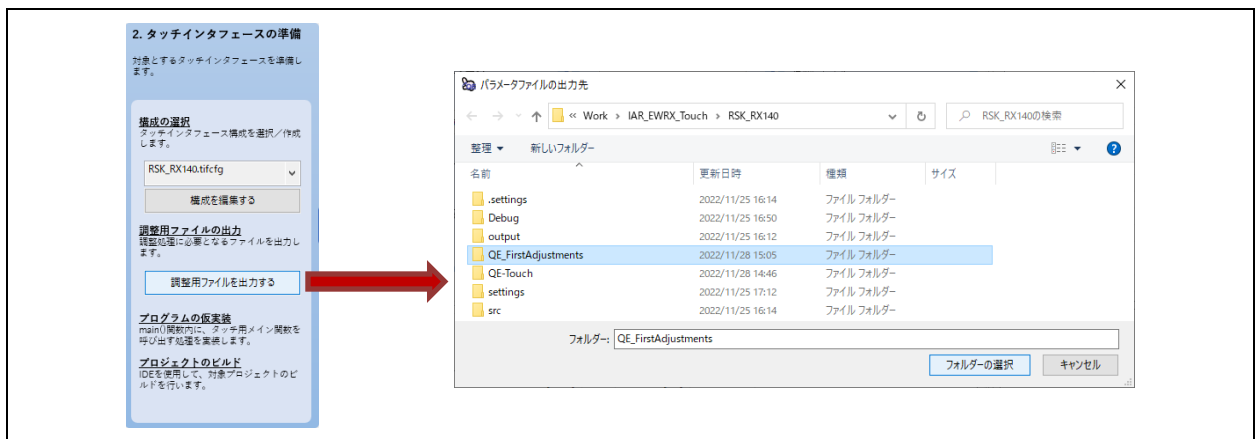


図 3-7 調整用ファイルの出力

この図の例では、プロジェクトフォルダ配下に”QE\_FirstAdjustments”フォルダを作成し、パラメータの出力先に選択しています。

- (2) 出力先のフォルダを選択すると、“周辺モジュールクロック(PCLKB または PCLKL)の周波数”ダイアログが表示されますので、周波数に“48”を入力し、[OK]をクリックします。

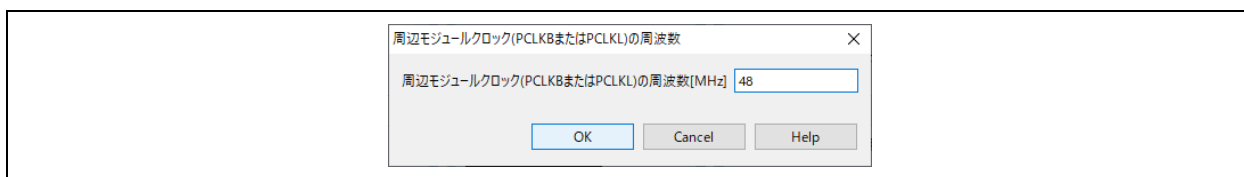


図 3-8 周辺モジュールクロック(PCLKB または PCLKL)の周波数の設定

- (3) 続いて表示された“マイコンの動作電圧”ダイアログでは、 をクリックし、電圧値に[5.0V : (4.5V ≤ VCC ≤ 5.5V)]を選択して[OK]をクリックします。外部電源を利用しますので使用するマイクロコントローラの電気的特性をご確認ください

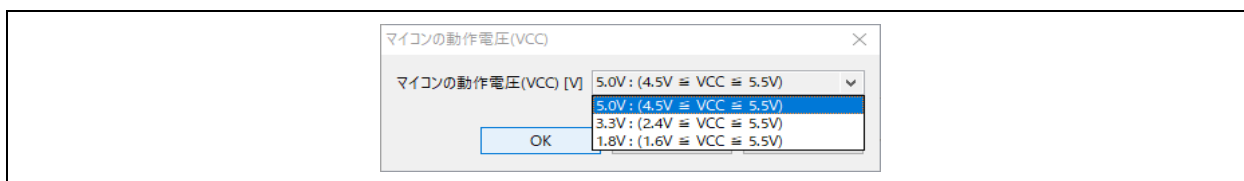


図 3-9 マイコンへの動作電圧(VCC)の設定

- (4) 最後に“QE for Capacitive Touch”ダイアログが表示されます。この後、ダイアログ内の指示に従い、設定を行っていきます。また、ダイアログの内容は画面下部の“コンソール”パネルにも同じメッセージ内容が表示されます。

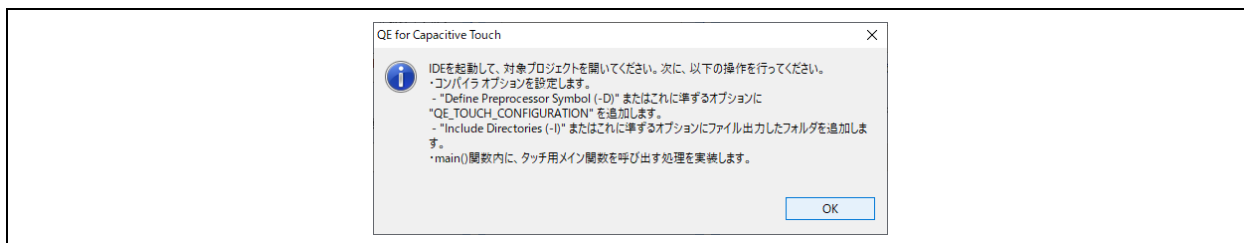


図 3-10 “QE for Capacitive Touch”ダイアログ



図 3-11 “コンソール”パネル

### 3.3 調整用プログラムの実装

IAR EWRX プロジェクトで、調整用プログラムを動作させるためにプログラムの修正と調整用ファイルの追加、コンパイラオプションの追加、デバッグ向けの設定を行います。

- (1) Windows エクスプローラで、📁 ワークスペースファイル名.eww をダブルクリックし、IAR EWRX ウィンドウを開きます。

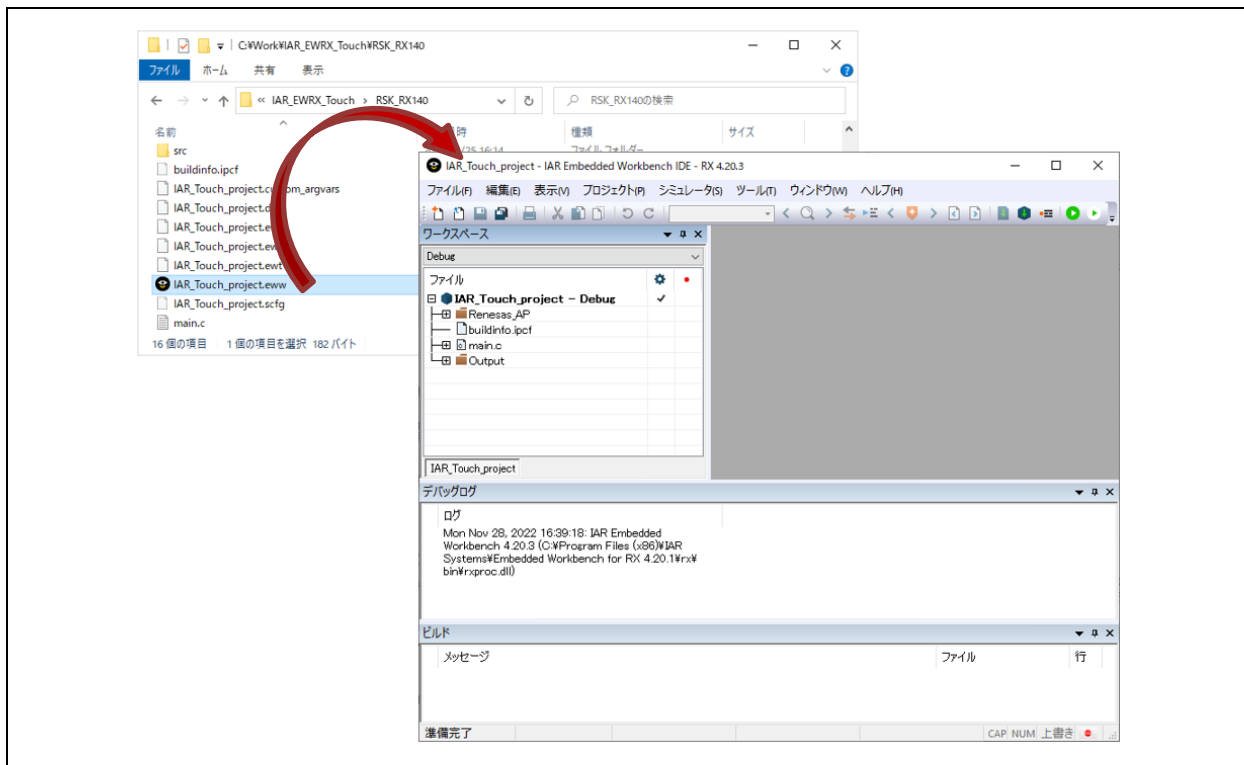


図 3-12 IAR EWRX ウィンドウ

- (2) メイン関数を変更します。

main()関数内に、タッチ用メイン関数を呼び出す処理を実装します。

- main.c をダブルクリックして開きます。タッチ用メイン関数の宣言と呼び出しを追加し、保存して終了します。  
5 行目 : extern void qe\_touch\_main(void);      9 行目 : qe\_touch\_main();

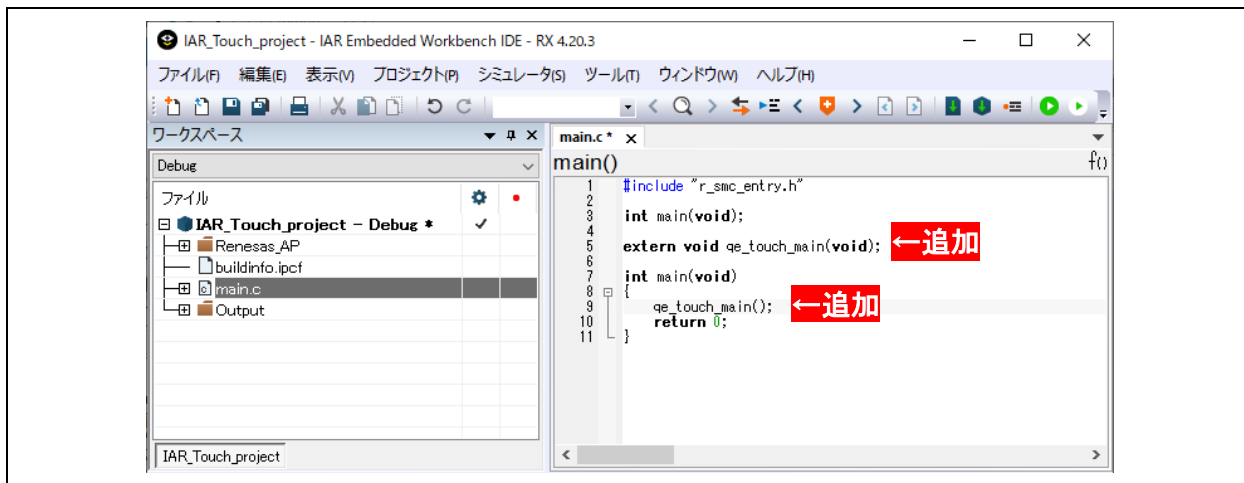


図 3-13 main() プログラムの修正



(3) 調整用ファイルを追加します。

調整用ファイルを”qe\_gen”フォルダにコピーして、プロジェクトに調整用ファイルを追加します。

- a. ワークスペースファイル格納先フォルダに”qe\_gen”フォルダを作成します。調整用ファイルを保存したフォルダ内のすべてのファイルを”qe\_gen”フォルダへコピーします。

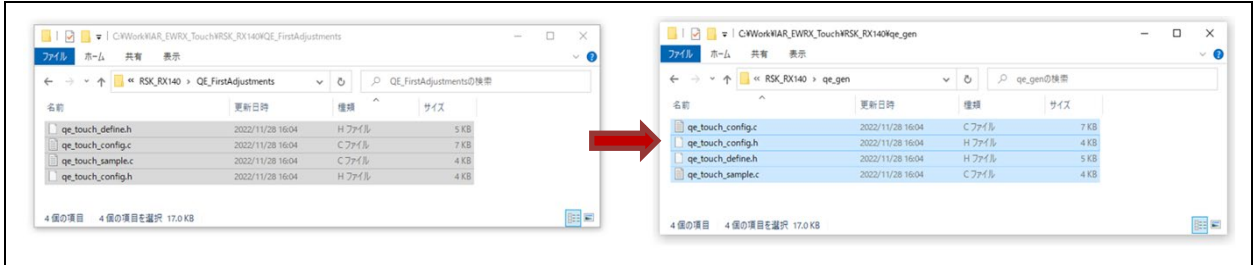


図 3-14 調整用ファイルのコピー

- b. [ワークスペース]ウィンドウでプロジェクト名を右クリックし、[追加(A)] → [グループ(G)...]を順に選択します。[グループの追加]ダイアログで、”qe\_gen”を入力し、[OK]をクリックします。

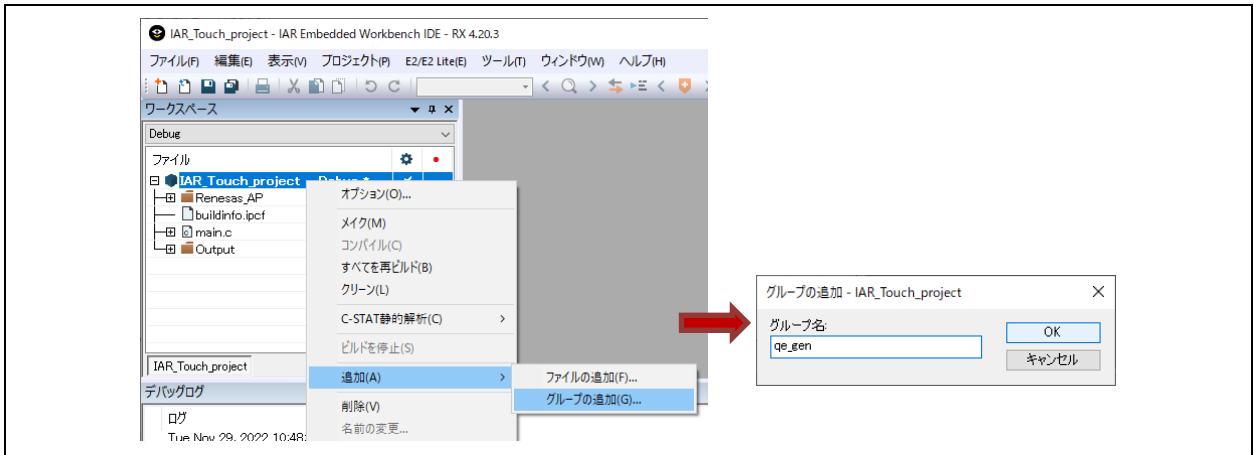


図 3-15 グループの追加

- c. 追加したグループ名を右クリックし、[追加(A)] → [ファイルの追加(F)...]を順に選択します。[ファイルの追加]ダイアログで、すべてのファイルを選択し、[開く(O)]をクリックします。

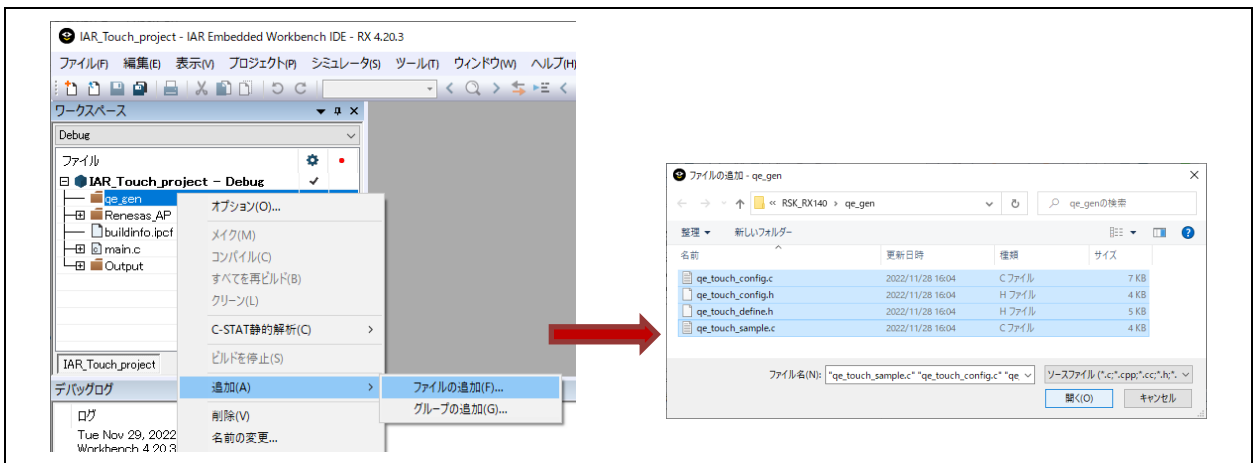


図 3-16 ファイルの追加

## 3.4 プロジェクトのビルド

## (1) コンパイラオプションを追加します。

プリプロセッサの設定に調整用ファイルのインクルードパスとシンボル定義を追加します。

- [ワークスペース]ウィンドウでプロジェクト名を右クリックし、[オプション(O)...]を選択して、[オプション]を表示します。
- [カテゴリ:]に“C/C++ コンパイラ”を選択し、[プリプロセッサ]タブを選択します。

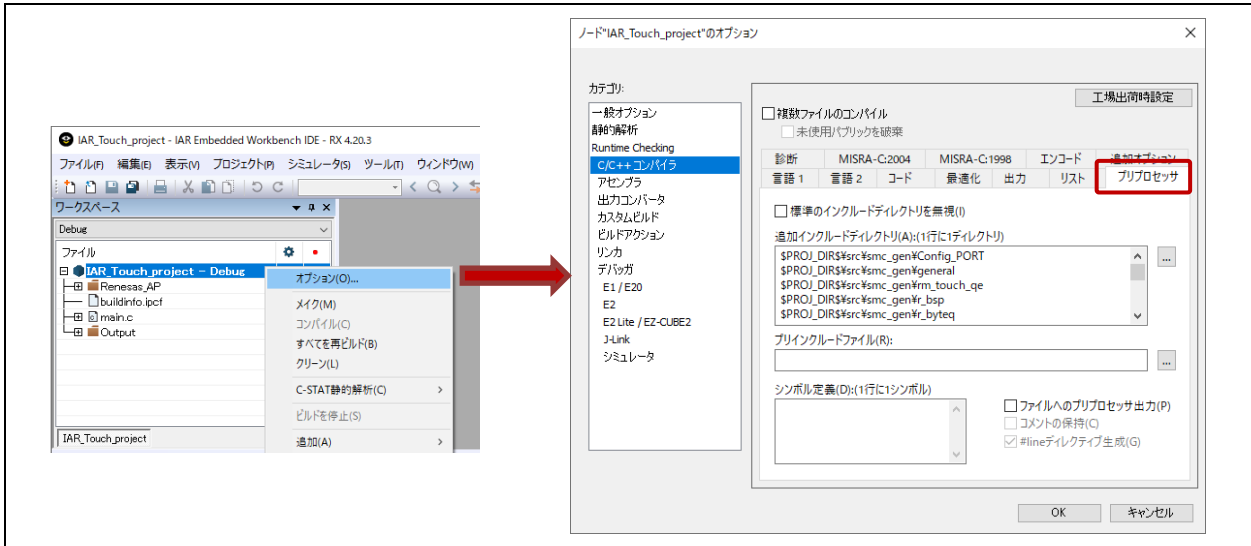


図 3-17 “C/C++ コンパイラ”の[プリプロセッサ]タブの表示

## a. インクルードパスを追加します。

- ① [プリプロセッサ]タブのページで、“追加のインクルードディレクトリ(A):(1行に1ディレクトリ)”の[...]をクリックし、[インクルードディレクトリを編集]ダイアログを開きます。
- ② [インクルードディレクトリを編集]ダイアログで一番下までスクロールし、<クリックして追加>をクリックします。
- ③ [インクルードディレクトリを追加]ダイアログが開くので、調整用ファイルの格納フォルダを選択し、[フォルダーの選択]をクリックします。

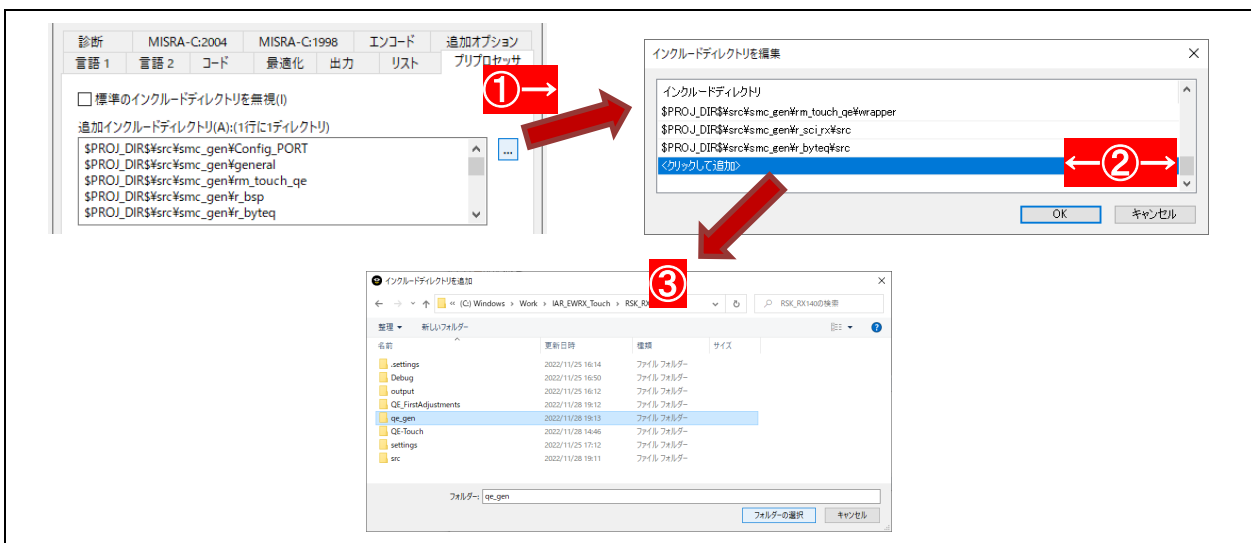


図 3-18 インクルードディレクトリの追加

- ④ 追加したインクルードディレクトリを絶対パス指定から相対パス指定に変更します。  
 [インクルードディレクトリを編集]ダイアログで対象ディレクトリの[▼]をクリックし、  
 \$PROJ\_DIR\$で始まる相対ディレクトリを選択します。[インクルードディレクトリを編集]ダイア  
 グの表示が相対パスに変更されたことを確認し、[OK]をクリックします。

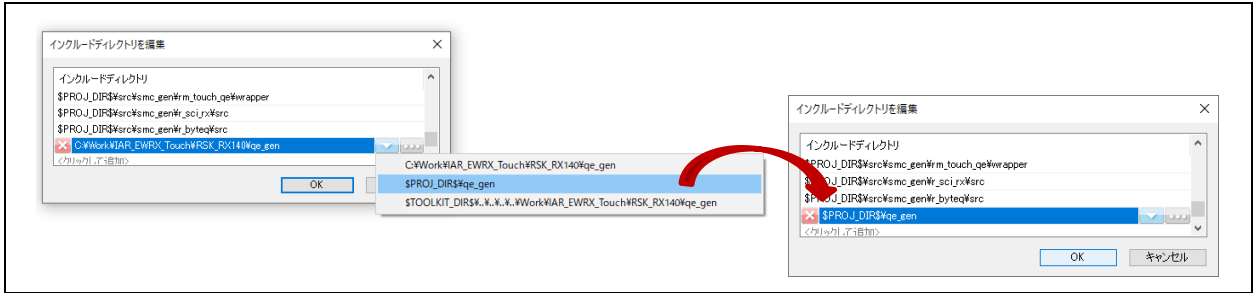


図 3-19 インクルードディレクトリを相対パスへ変更

- b. [シンボル定義(D): (1 行に 1 シンボル)] に"QE\_TOUCH\_CONFIGURATION"を入力します。



図 3-20 シンボル定義の追加

- (2) 続いて、デバッガの設定をします。

- [カテゴリ:] に“デバッガ”を選択し、[設定]タブを選択します。
- [ドライバ(D):] の ▼ をクリックし、“E2 Lite / EZ-CUBE2”を選択します。
- [指定位置まで実行(R):] に“main”を入力して、✓をつけます。
- [OK]をクリックして、[オプション]ダイアログを閉じます。

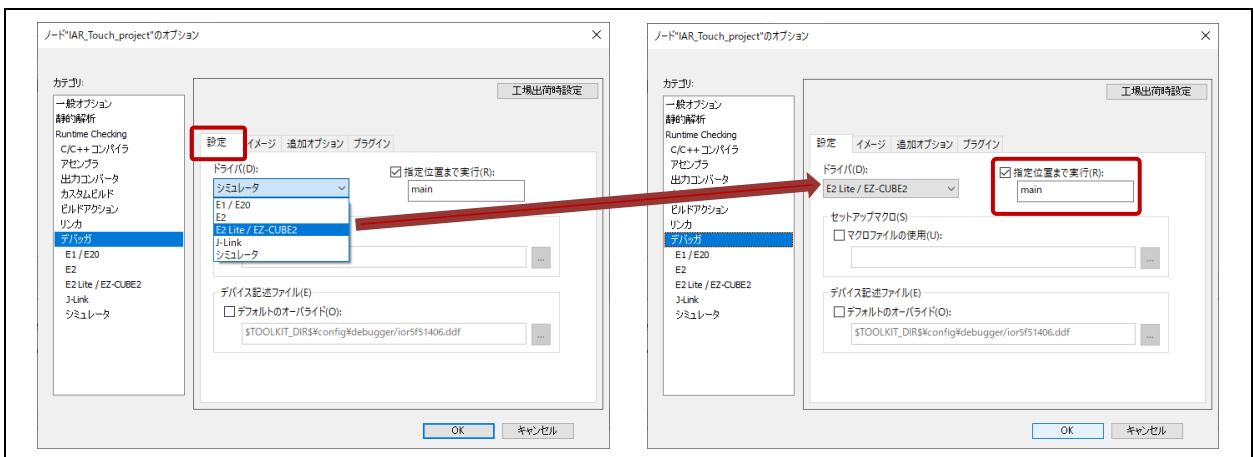


図 3-21 デバッガ - 設定タブ

## (3) ハードウェア設定をします。

- a. メニューバーで、[E2/E2 Lite(E)]メニューが選択可能になるので、[E2/E2 Lite(E)]メニュー → [ハードウェア設定(H)...]を選択します。

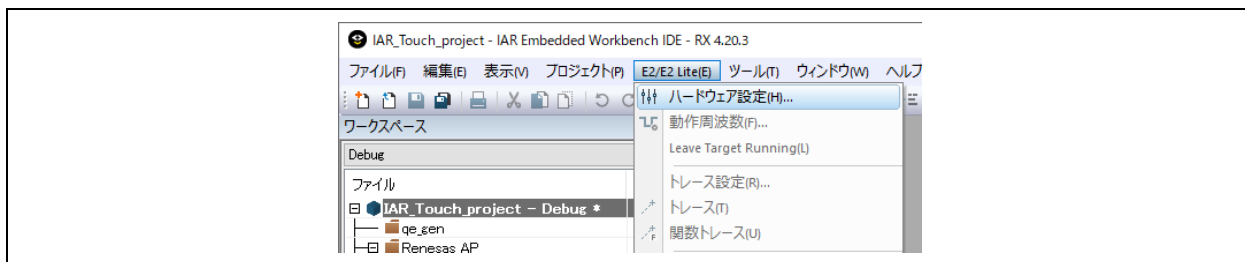


図 3-22 ハードウェア設定の選択

- b. [ハードウェア設定]ダイアログで[MCU]タブを選択し、設定します。

- [MCU]の[ICLK frequency (I):] "48.000" を入力します。
- [電源]の"エミュレータからターゲットに電源を供給しますか?(最大 200mA)"  
✓がついていれば、クリックして✓をはずします。

[OK]をクリックします。

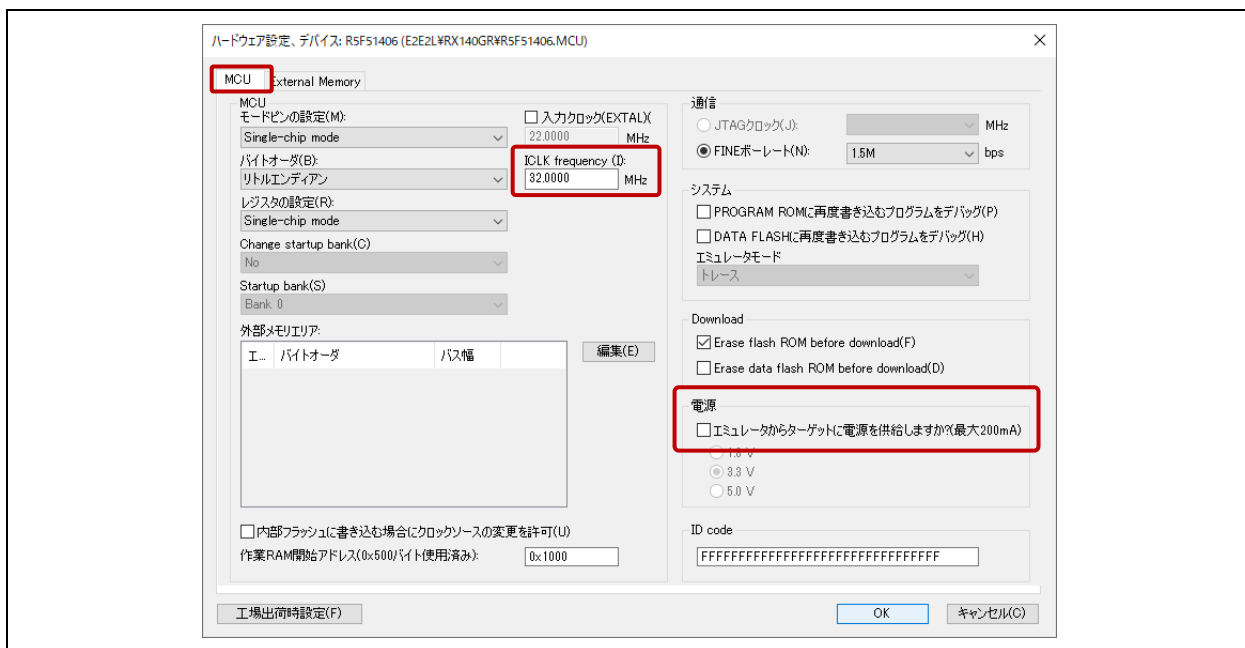


図 3-23 ハードウェア設定-[MCU]タブ

(4) 調整用ファイルを追加したプロジェクトをビルドします。

[ワークスペース]ウィンドウでプロジェクト名を右クリックし、[すべてを再ビルド(B)] を選択して、ビルドします。  
 ビルド終了後に[ビルド]ウィンドウまたはメインウィンドウ下部のステータスバー表示でエラーがないことを確認してください。

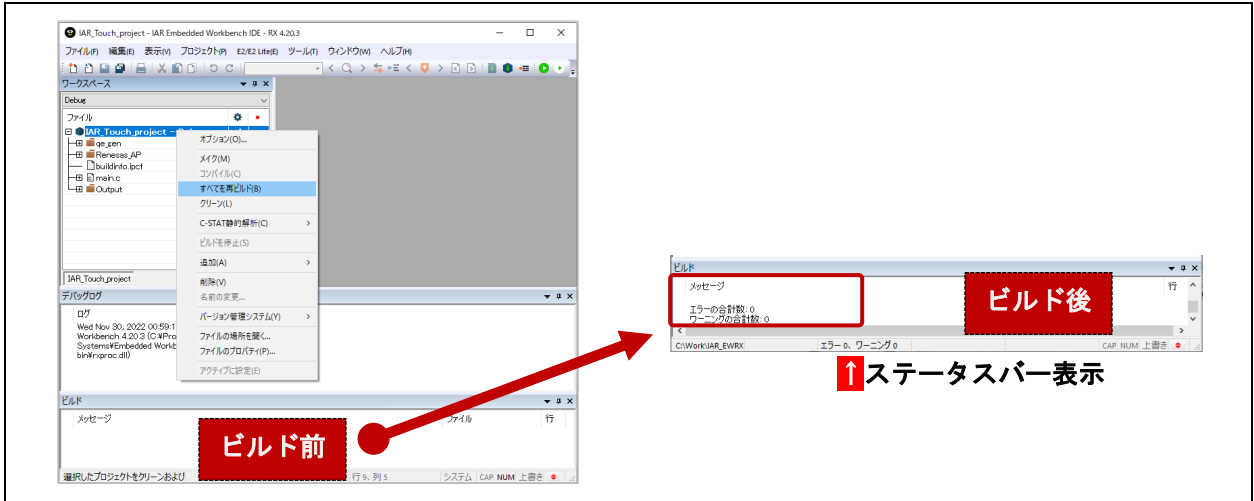


図 3-24 調整用ファイルを含めたビルド

4. タッチセンサの調整

RSK-RX140 とホスト PC を接続します。IAR EWRX プロジェクトのデバッグを開始し、スタンドアロン版 QE でタッチセンサの自動調整処理を行います。

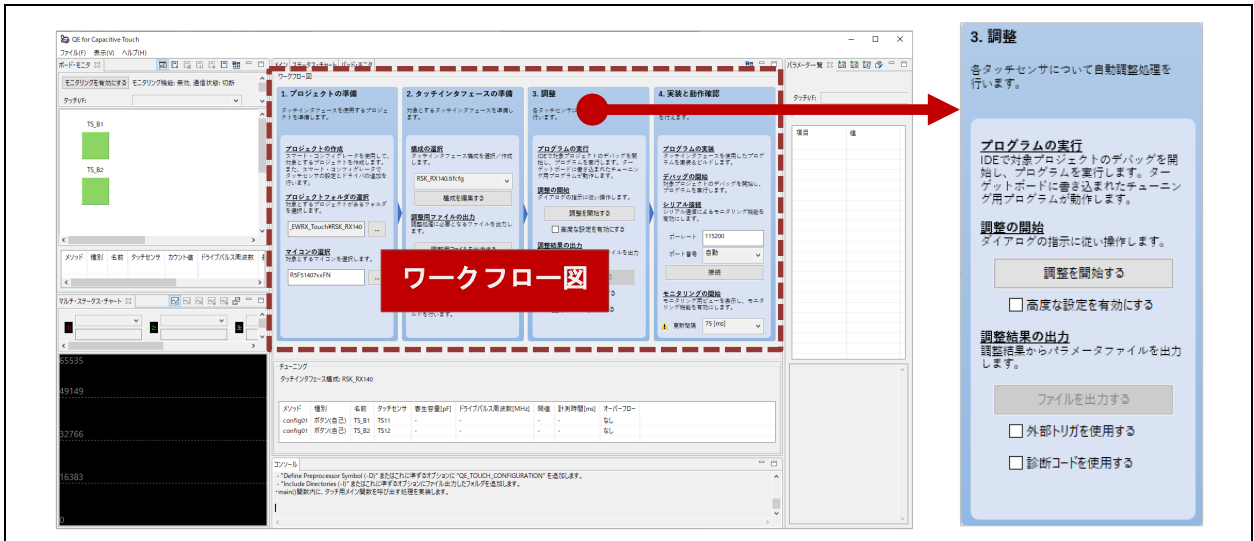


図 4-1 スタンドアロン版 QE - ワークフロー図 :Step3 調整

#### 4.1 RSK-RX140 ボードと PC の接続

下図のように接続します。ボードに接続する電源については必ず、安定化された(最小 10W)DC 出力でセンタープラスの電源をご使用ください

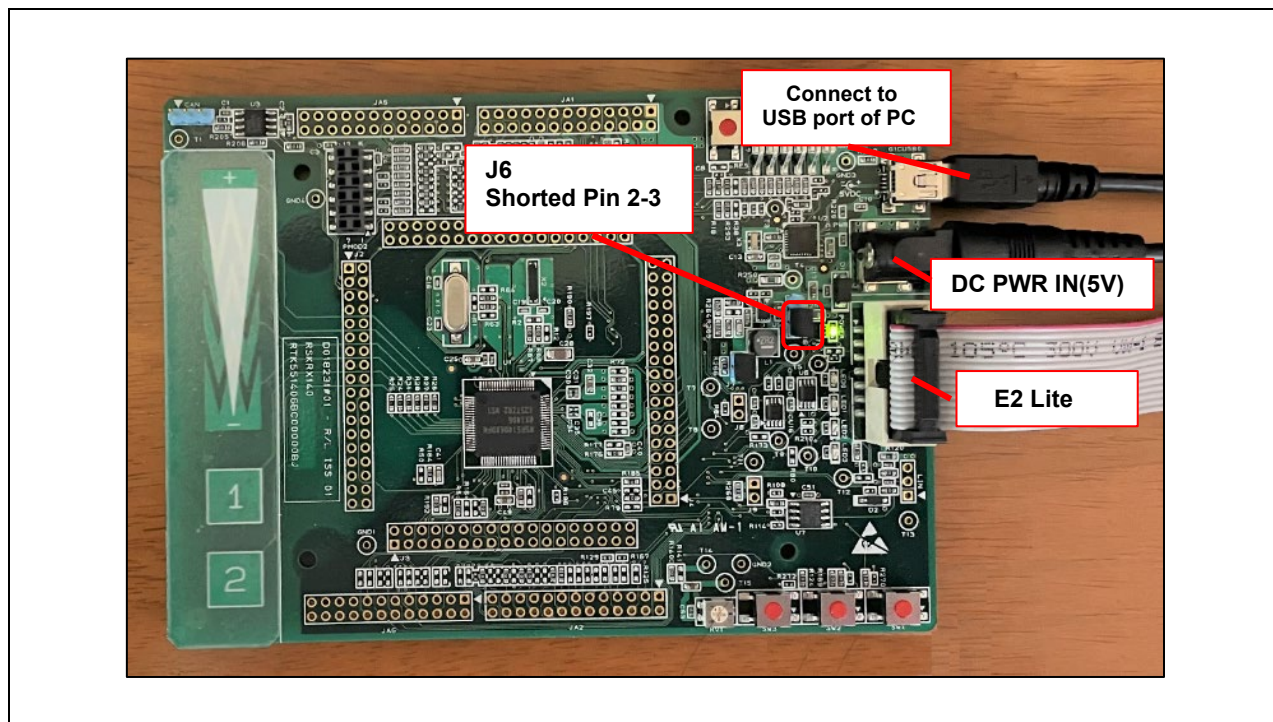


図 4-2 RSK-RX140 ボードの接続

## 4.2 調整の開始

IAR EWRX で調整プログラムを実行し、スタンドアロン版 QE で調整開始します。

(1) IAR EWRX で調整用プログラムを実行します。

- ① [プロジェクト(P)]メニュー → [ダウンロードしてデバッグ(D)] を選択し、デバッグを開始します。
- ② デバッグの開始後に main() プログラムで停止します。
- ③ [デバッグ(D)]メニュー → [実行(G)] を選択して、実行します。

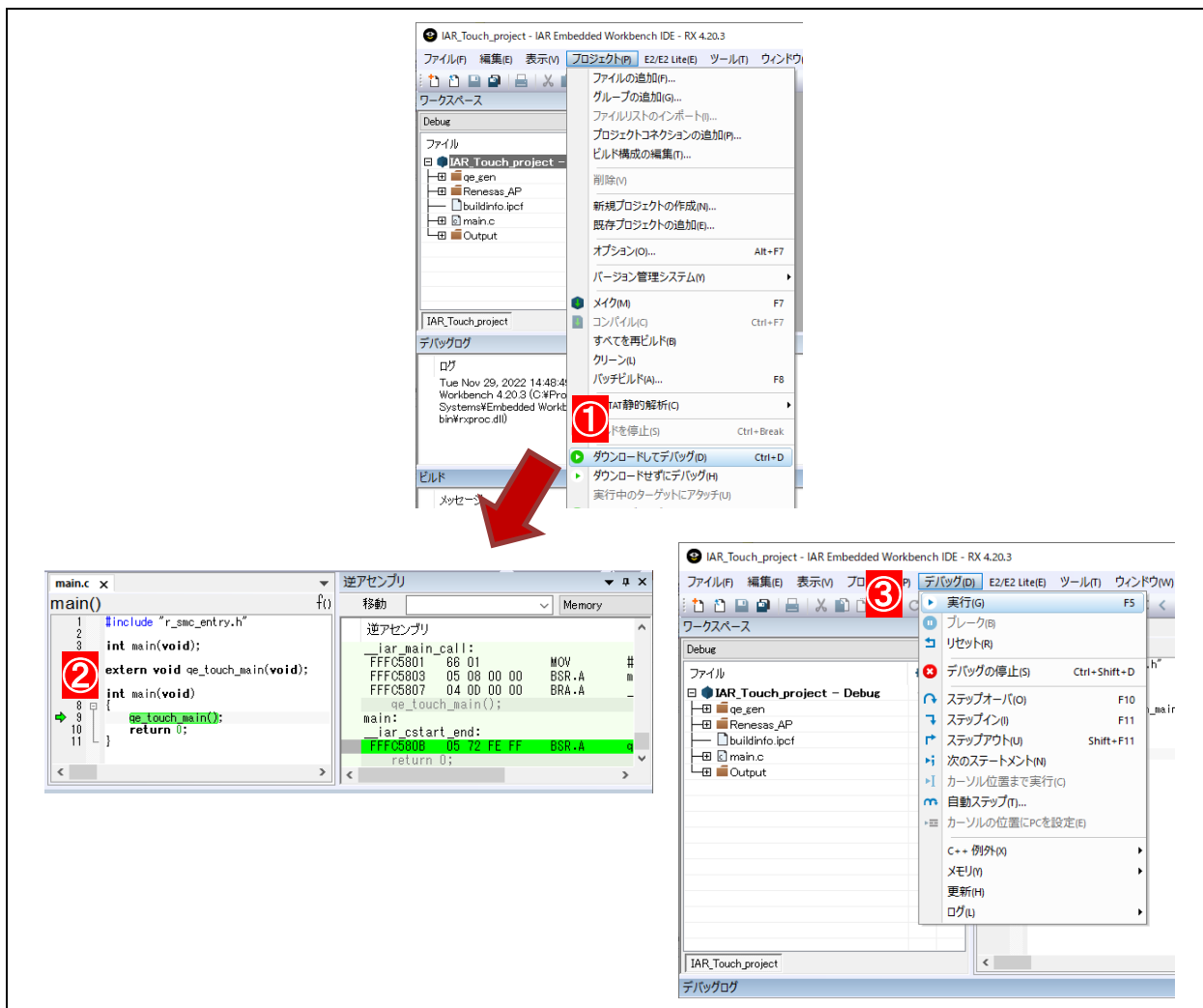


図 4-3 Step3: プログラムの実行

## (2) スタンドアロン版 QE で、自動調整を開始します。

ワークフローの[調整を開始する]をクリックし、QE for Capacitive Touch の自動調整が開始します。自動調整処理中に対話形式の手順がありますので、案内のダイアログに従ってください。

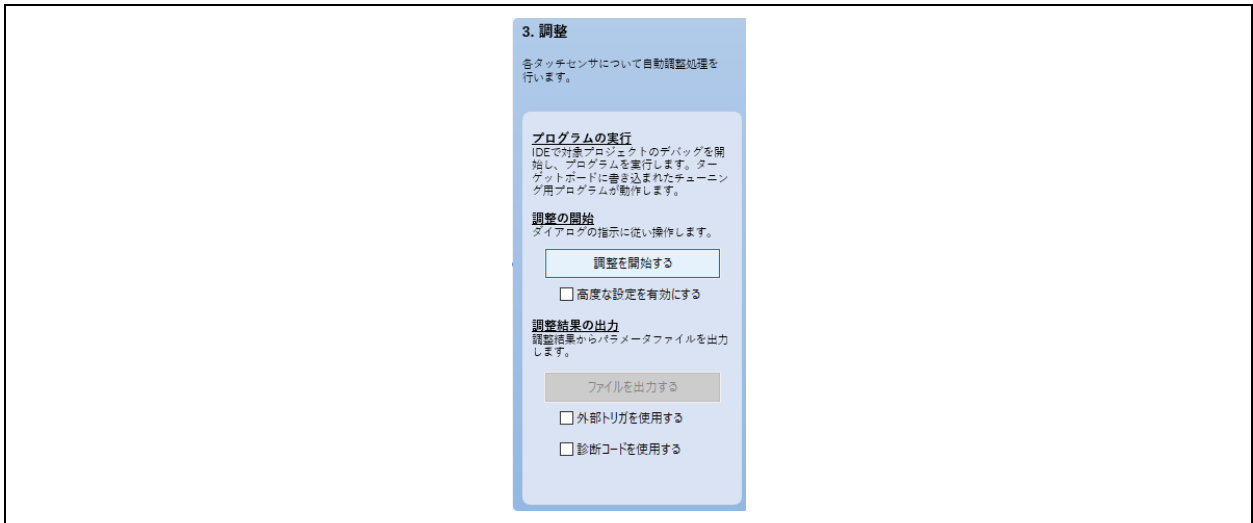


図 4-4 自動調整処理中

- a. 自動調整プロセスをガイドする”自動調整処理中”ダイアログを適宜、確認してください。通常、初期調整のプロセス中は操作を必要としません。

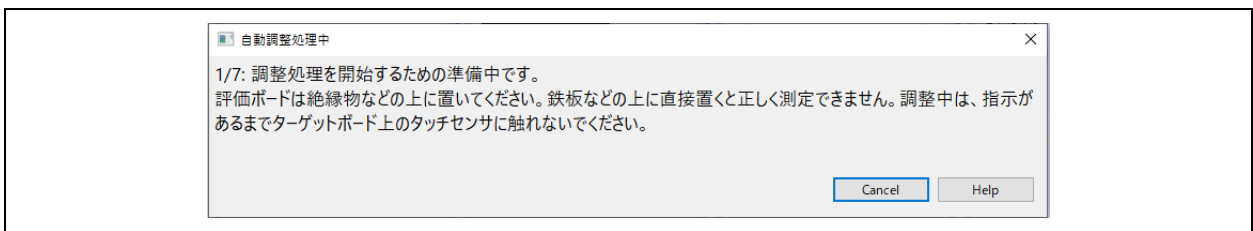


図 4-5 自動調整処理中 初期調整

- b. いくつかの工程を経て、タッチ感度の計測が可能になると、以下のような案内をするダイアログが表示されます。調整プロセスにおける最初の対話形式の手順です。

- ここではダイアログで表示されているセンサ (TS\_B1/TS11) をタッチします。
- センサに触れているとき、バーグラフは右に増加し、数値で示すカウント値が増えます。
- センサに触れたまま、PC のキーボードのいずれかのキーを押して計測を確定します。

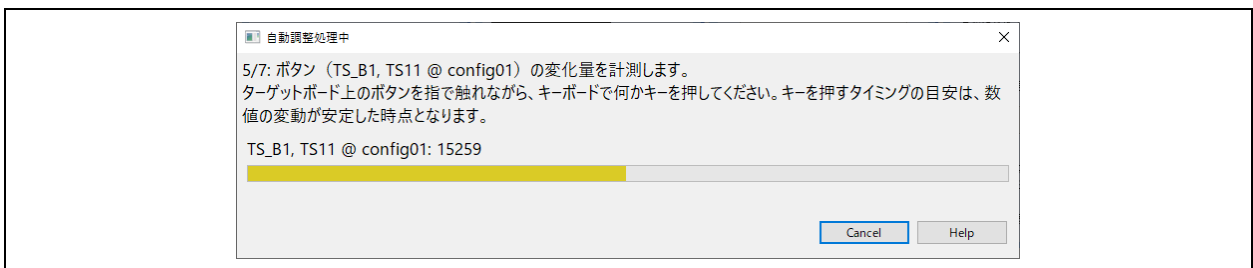


図 4-6 自動調整処理中 タッチ感度の計測 (TS\_B1)



- c. 次に、TS\_B2/TS12 に対して、前の手順を繰り返します。

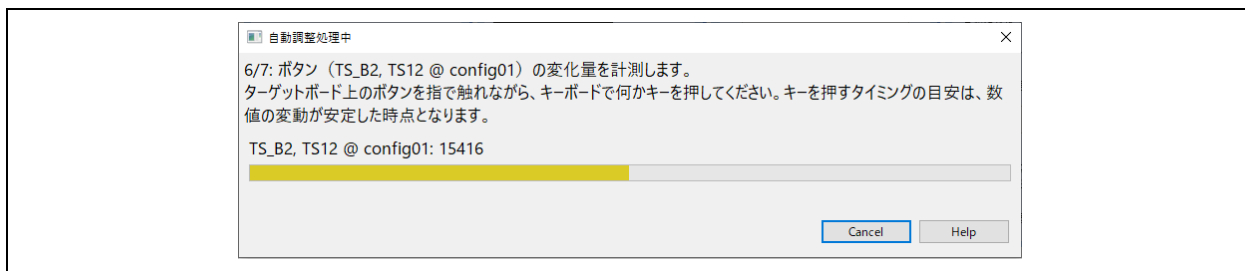


図 4-7 自動調整処理中 タッチ感度の計測 (TS\_B2)

- d. 自動調整が完了すると、以下のようなダイアログが表示され閾値を確認できます。この閾値はミドルウェアでタッチのイベント判定に使用されます。

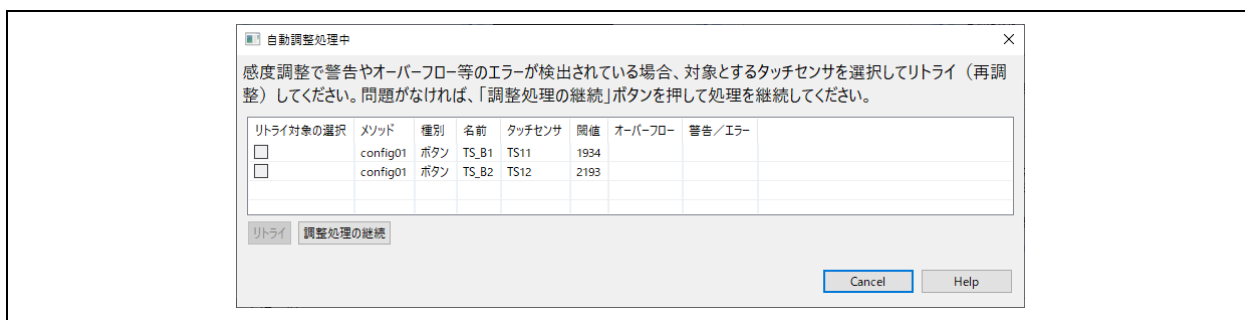


図 4-8 自動調整処理中 計測結果の表示

(3) 調整の終了

タッチセンサの閾値を表示するダイアログの[調整処理を継続]をクリックします。これで自動調整プロセスは終了し、ターゲットボードとのデバッグセッションを切断します。

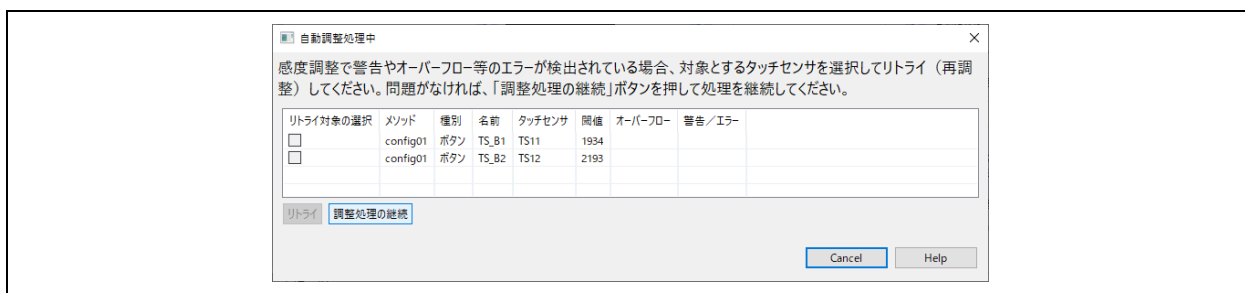


図 4-9 自動調整処理中 調整処理を継続

### 4.3 調整結果の取得

スタンドアロン版 QE で調整結果を取得します。

#### (1) IAR EWRX のデバッグを終了します。

[デバッグ(D)]メニューで[デバッグの停止(S)]を選択します。

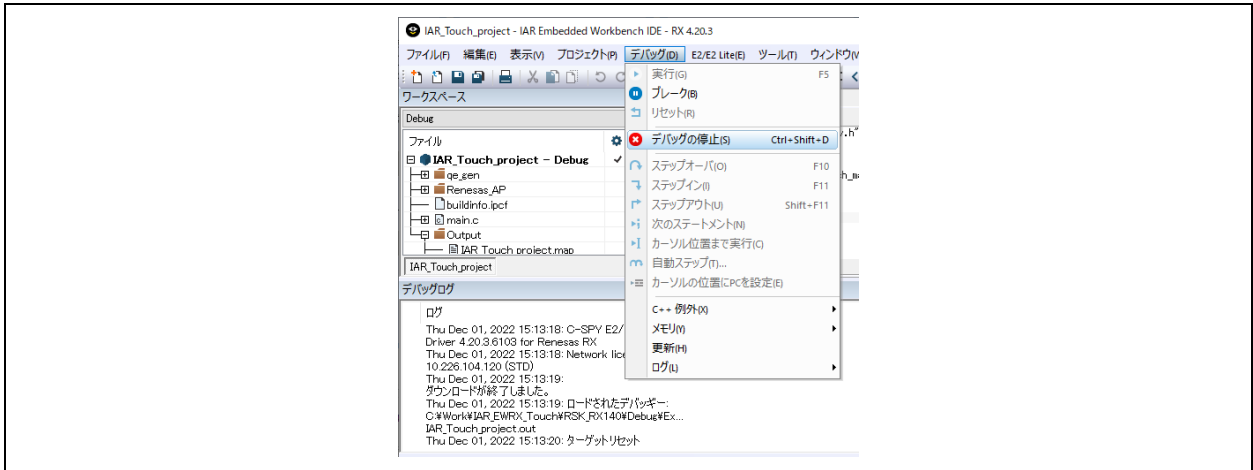


図 4-10 調整関連

#### (2) スタンドアロン版 QE で[ファイルを出力する]をクリックします。

“パラメータファイルの出力先”ダイアログで、フォルダを選択し、[フォルダの選択]をクリックします。ここでは、あらかじめ”qe\_tuning\_r1”フォルダを作成して、このフォルダを選択しています。

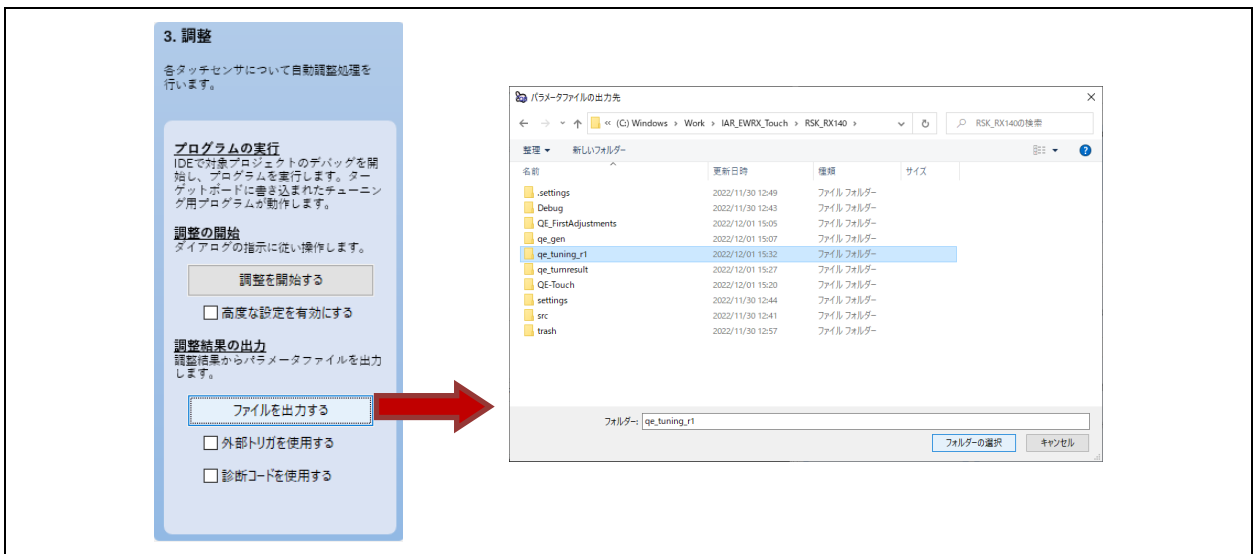


図 4-11 調整結果の出力

## (3) IAR EWRX のプロジェクトにタッチインタフェースの調整値を反映します。

格納したフォルダを開き、以下の3つのファイルをワークスペースフォルダ配下の"qe\_gen"フォルダへコピーします。調整前の内容が不要であればファイルを上書きコピーします。必要な場合は、退避させてください。

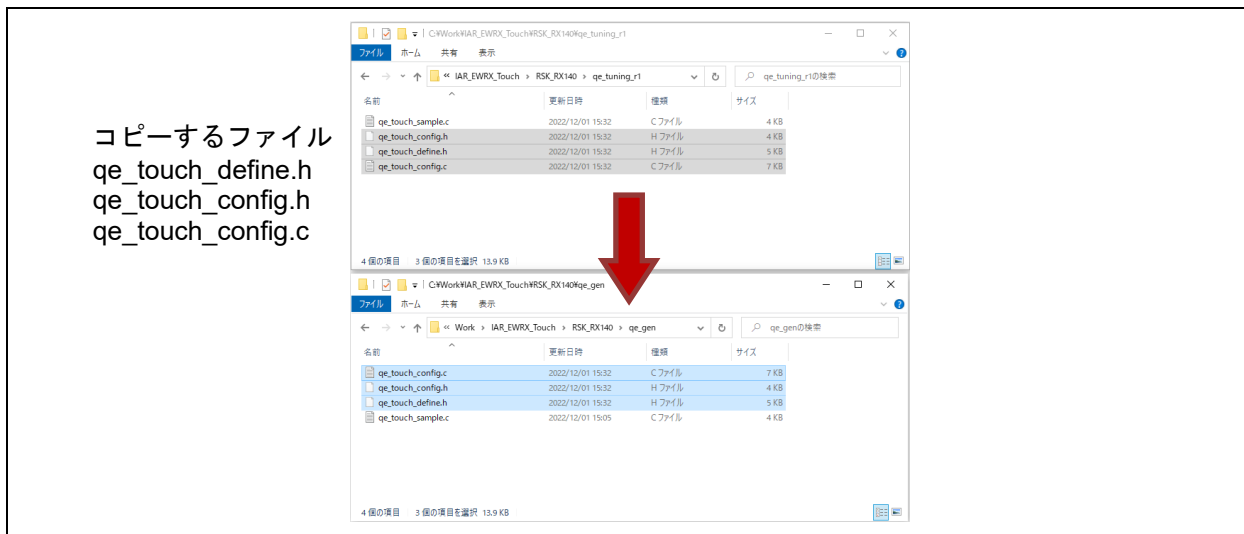


図 4-12 調整値ファイルのコピー

(4) IAR EWRX プロジェクトをビルドして、タッチインタフェースの調整値を反映したモジュールを生成します。ビルドの方法は、[3.4.\(4\)](#) 章を参照してください。

## 5. 実装と動作確認

ここでは、自動調整に利用した自動調整処理で得られたパラメータファイルでビルドした 4.3.(4) 章のモジュールを利用し、スタンドアロン版 QE でモニタリングをします。  
 スタンドアロン版 QE のワークフロー図、“2. タッチインタフェースの準備” で取得した調整用ファイルの “qe\_touch\_sample.c” に含まれている関数は、タッチセンサの状態のスキャンを実装しています。特にプログラムの変更することなく、モニタリングで動作確認します。

### 5.1 モニタリング

IAR EWRX でビルドしたプログラムを、4.2.(1) 章の手順に従って実行します。

#### (1) モニタリングの開始

ワークフロー図の“4. 実装と動作確認”の“シリアル接続”にある[接続]をクリックします。接続が完了するまで、少し待ちます。

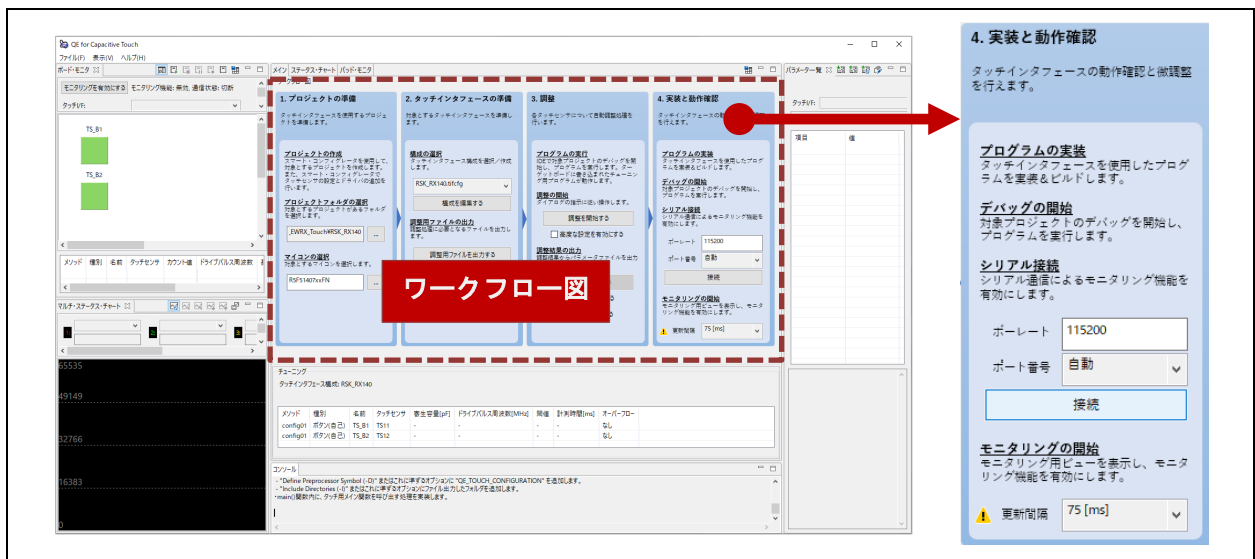


図 5-1 Step4. 実装と動作確認 接続

(2) 接続するとボード・モニタビューの“通信状態:”にシリアル通信 (UART/USB) で接続中”と表示されます。

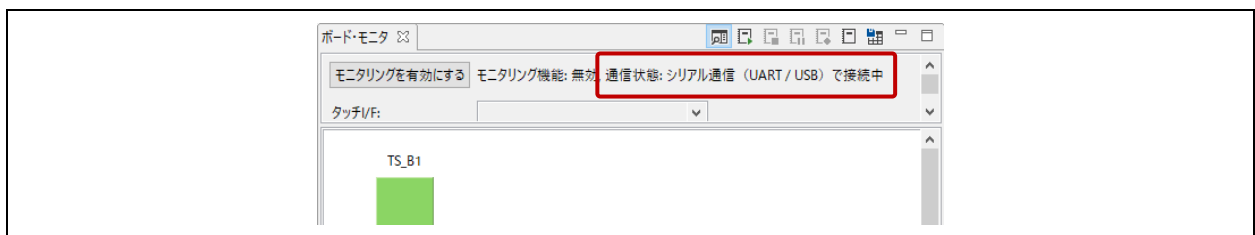


図 5-2 シリアル通信状態

- (3) [モニタリングを有効にする]をクリックします。“モニタリング機能: 無効”が“モニタリング機能: 有効”に切り替わります。

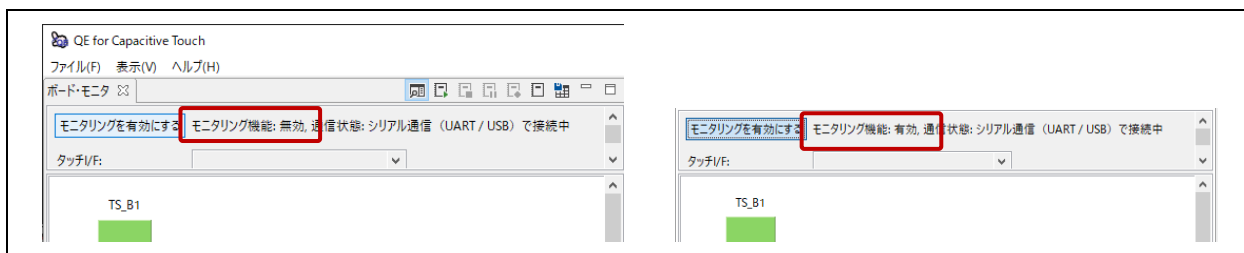


図 5-3 モニタリング機能の有効

- (4) ターゲットボード上のボタン TS\_B2 をタッチします。“ボード・モニタ”では、以下のように、タッチした状態をボタン上の指アイコンで表します。

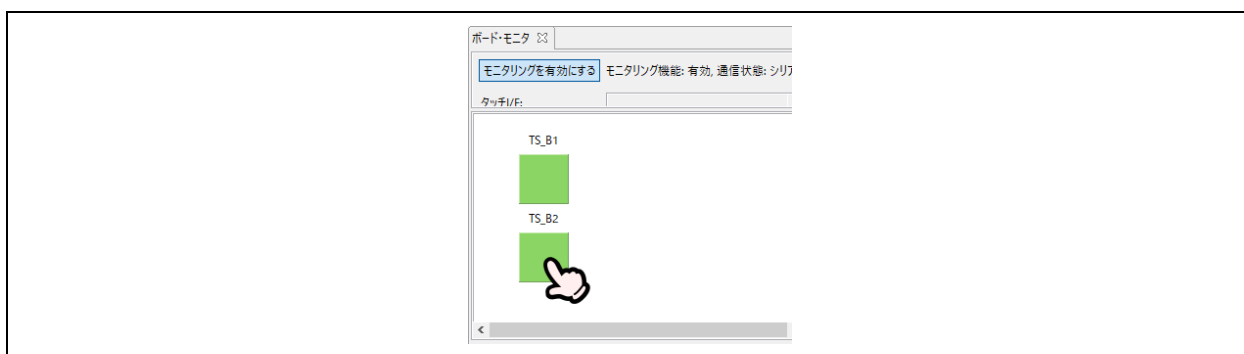


図 5-4 ボード・モニタ: 指アイコン

- (5) カウント値をグラフィカルに表示するには、以下のように [ステータス・チャート] タブをクリックします。

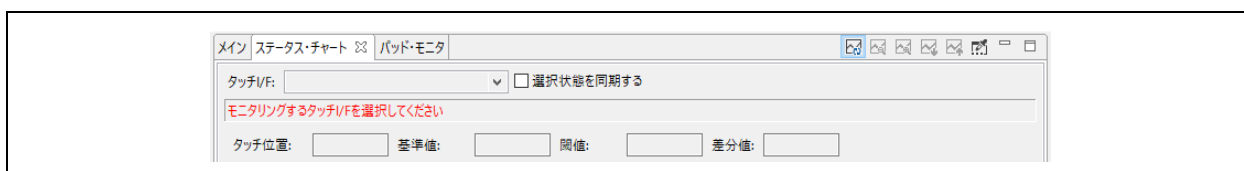


図 5-5 [ステータス・チャート] タブ選択

- プルダウンメニューから“TS\_B1@ config01”を選択します。

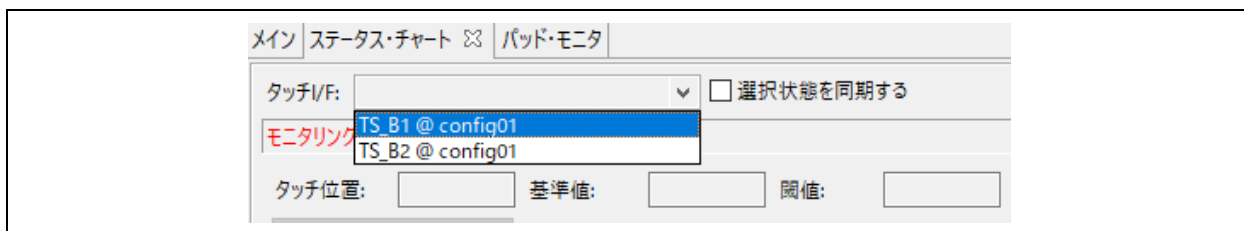


図 5-6 タッチ I/F 選択

- グラフには実行中の値が表示されます。ターゲットボード上のボタン TS\_B1 をタッチすると、カウント値がグラフ上で増加することを確認できます。緑のラインは閾値を表し、rm\_touch\_qe FIT モジュールでボタンが操作/タッチされているかどうかを判断するために使用されます。グラフ下部の赤い矩形はカウント値が閾値を超えてタッチと判定されたことを表します。

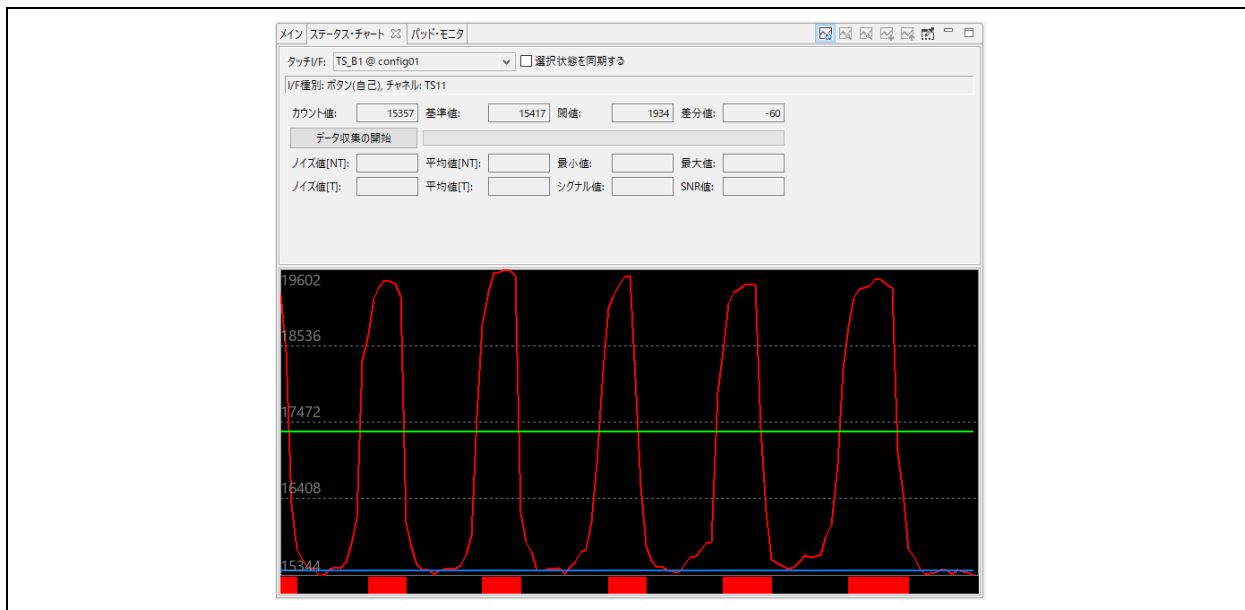


図 5-7 ステータス・チャート

- 複数のタッチセンサのタッチカウント値をグラフィカルに表示するには、[マルチ・ステータス・チャート]タブを使用します。タッチインタフェースを選択すると表示されます。

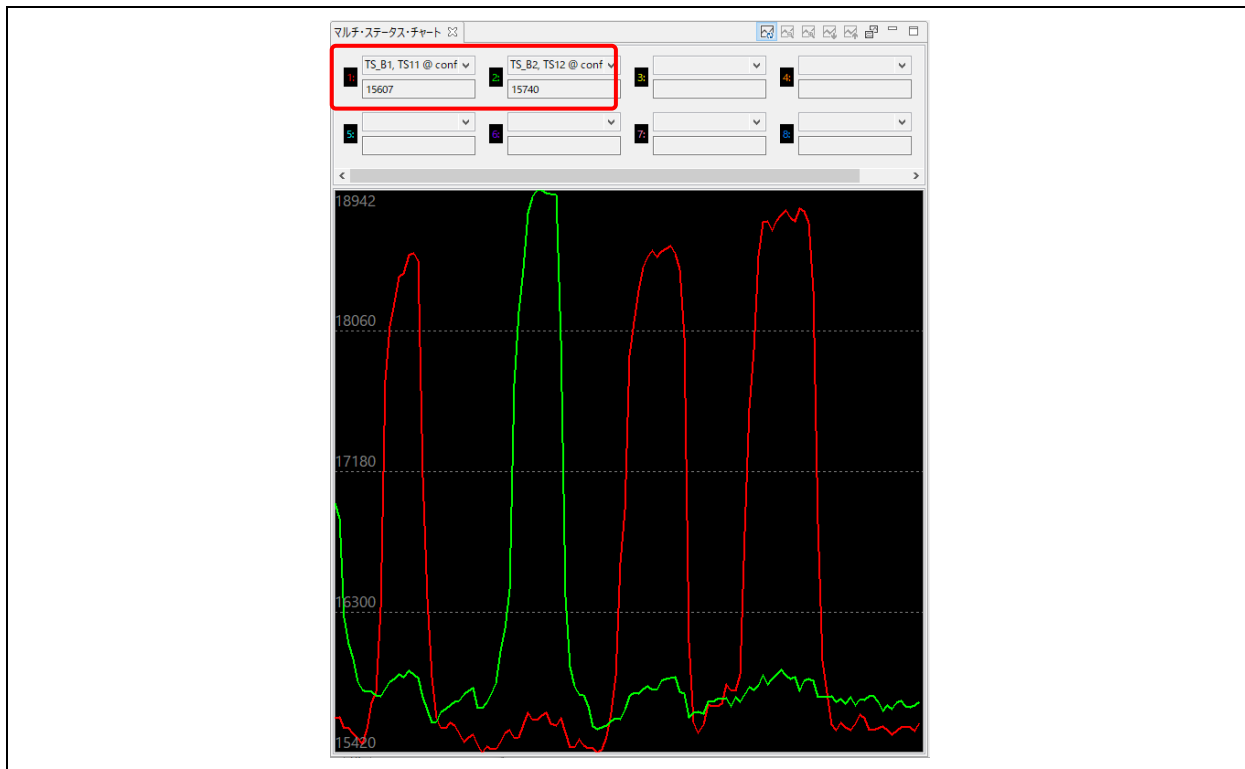


図 5-8 マルチ・ステータス・チャート]タブ

## 5.2 シリアル接続の切断

### (1) モニタリングの終了

モニタリング機能が有効となっているときは、[モニタリングを有効にする]をクリックし、モニタリング機能が無効となったことを確認します。

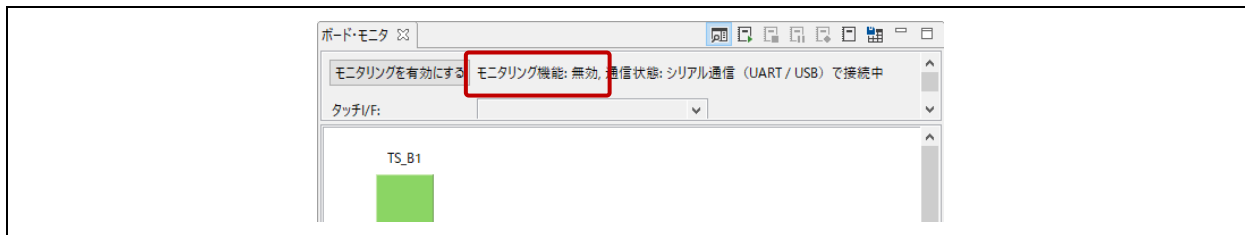


図 5-9 シリアル通信状態

### (2) シリアル接続を切断します。

ワークフロー図の"4. 実装と動作確認"の"シリアル接続"にある[切断]をクリックします。切断が完了するとコンソールビューに"¥¥.¥COMx との接続を切断しました。"が表示され、[切断]が[接続]にボタン表示が変わります。

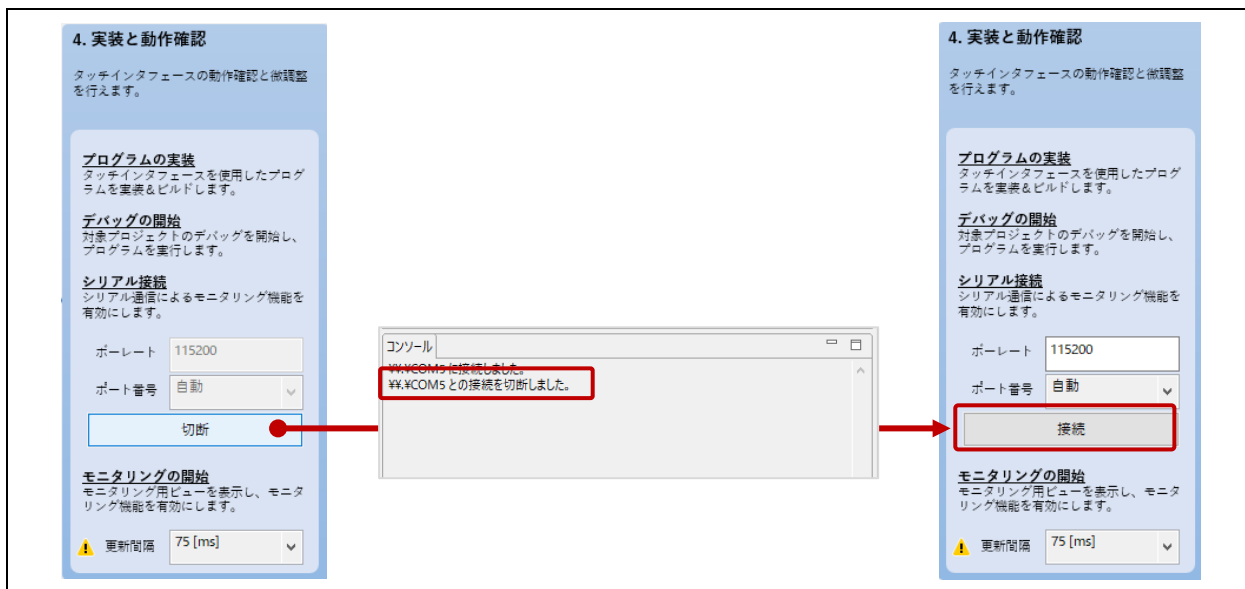


図 5-10 Step4. 実装と動作確認 切断

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2022.12.22	—	初版発行



## 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しないでください。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えば、フラッシュメモリ、レイアウトパターンなどの相違などにより、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。型名が異なる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まれます。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、リパ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、改造、複製、リパ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、不正な変更、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)

## 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。

## 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/)