# RX/RA/RL78 Family Renesas Synergy™ Platform

## QE for Capacitive Touch 3D Gesture Recognition Application Development Guide

### Introduction

This application note explains how to perform 3D gesture recognition incorporating AI using the capacitive touch sensor compatible development support tool (QE for Capacitive Touch).

This application note has been confirmed to work with RA2L1, but the method of performing 3D gesture recognition with QE for Capacitive Touch does not depend on the device. This application note can be used for devices equipped with capacitive touch IP on the RX / RA / RL78 family and Renesas Synergy ™ platform.

### Evaluation Kit to Be Operated

Capacitive Touch Evaluation System for RA2L1 (RSSK RA2L1)

3D Gesture Electrode Board

## Contents

# 1. Overview

## 1.1 System Overview

QE for Capacitive Touch is a development support tool that supports initial settings and sensitivity adjustment of the touch interface required for embedded system development using capacitive touch keys.

The “3D gesture recognition” feature of QE for Capacitive Touch supports the development of gesture applications with the use of AI. Though the development of AI applications involves many processes and a high level of difficulty, three features of the QE tool support development, allowing anyone to develop AI applications without the need for specialized AI skills.

The main functions of “3D gesture recognition” are as follows.

- Recording
  Gesture data can be registered just by actually performing the gesture that you want the AI to recognize on the capacitive sensor.
  The registered data are displayed in a list supporting the deletion of data from failed gestures and an importing feature.
- AI generation
  During the processes of creating the AI (data pre-processing, deep learning, and conversion to C source code), the QE tool automatically proceeds with optimum processing for gestures.
- Monitoring & Tuning
  You can monitor and tune the precision of the created AI on the actual machine. If a gesture has not been correctly recognized, you can immediately add the waveform to the learning data to give feedback to the AI. This eases proceeding with iterations to improve the AI engine.

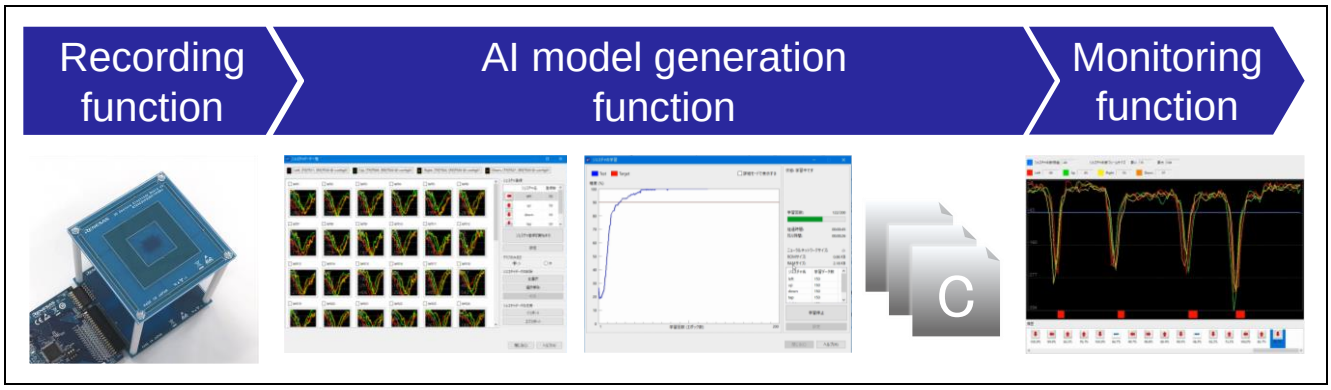


**Figure 1-1 Main Functions of QE for Capacitive Touch**

## 1.2 Operating Environment

The operating environment for 3D gesture recognition of QE for Capacitive Touch is shown in Table 1-1 and Figure 1-2.

In the figure in this application note, the target device is RA2L1, and it works with e$^2$ studio and J-Link. The method of performing 3D gesture recognition with QE for Capacitive Touch is device- and IDE-independent. This application note can be used for devices equipped with capacitive touch IP on the RX / RA / RL78 family and Renesas Synergy ™ platform.

The project used in this application note uses the project name “test”.

**Table 1-1 Operating Environment**

| Items | Contents |
|---|---|
| IDE | e$^2$ studio version 2021-10 or later<br>Flexible Software Package (FSP) v3.5.0 or later |
| Toolchains | GNU Arm Embedded Toolchain: 10.3-2021.10 or later<br>(GNU ARM Embedded 10.3.1.20210824) |
| QE | QE for Capacitive Touch V3.1.0 or later |
| e-AI Solution | e-AI Translator V2.1.0 or later |
| Evaluation board | RA2L1 CAP Touch CPU Board - RTK0EG0018C01001BJ |
| Sensor | 3D Gesture Electrode Board - RTK0EG0023B01002BJ |
| Emulators | Segger J-Link<br>E2 emulator Note<br>E2 emulator Lite Note |

Note: Please use a conversion cable because the connector shape is different.
RTE0T00020KCAC1000J: User-system Interface Cable for the E2 emulator (20-10 pins)

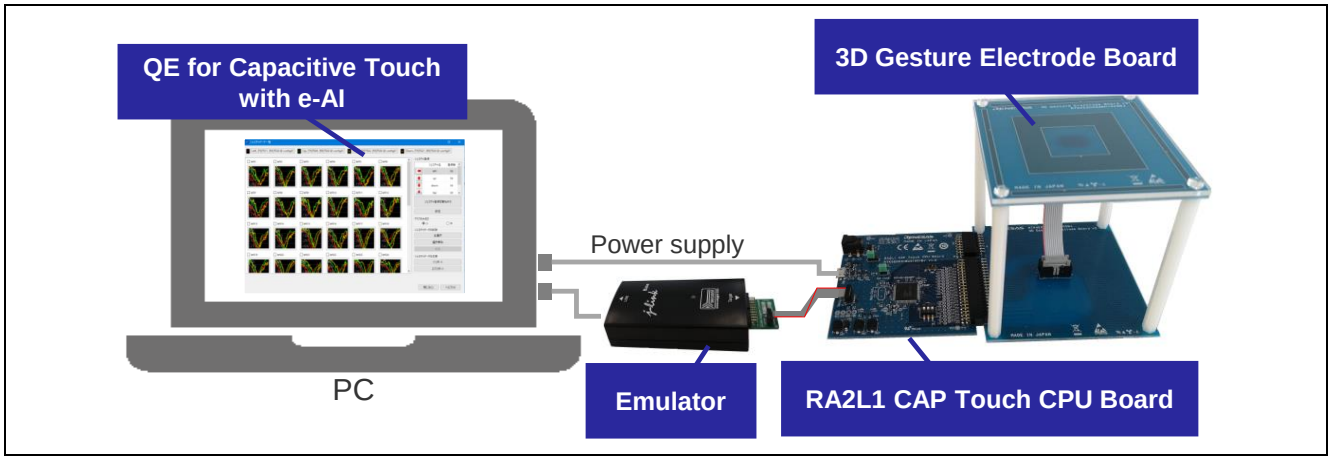


**Figure 1-2 Operating Environment**

## 1.3 References

- R01UH0853 RA2L1 Group User's Manual: Hardware
- R12QS0040 RA2L1 Group Renesas Solution Starter Kit
  Capacitive Touch Evaluation System Quick Start Guide

For references of QE for Capacitive Touch, refer to the following site of Renesas.

- QE for Capacitive Touch: Development Assistance Tool for Capacitive Touch Sensors Information for Users
  www.renesas.com/software-tool/qe-qe-for-capacitive-touch-support

# 2. Preparation

The preparations required for 3D gesture recognition with QE for Capacitive Touch are as follows.

- Install e² studio
- Apply the following tools to e² studio
  - Configuration tools
  - Toolchains
  - QE for Capacitive Touch
  - e-AI Solution (Requires regular license renewal)
- Connection to the board

This chapter describes the installation of QE for Capacitive Touch and the board connection. Applying the e-AI Solution is described in Chapter 4.3.1 because it requires the creation of a project.

## 2.1 Installation of the QE for Capacitive Touch

This chapter describes how to install QE from Renesas Software Installer of e² studio.

Select “Renesas Software Installer” from the e² studio menu. When the Renesas Software Installer window appears, select “Renesas QE” and click “Next”.

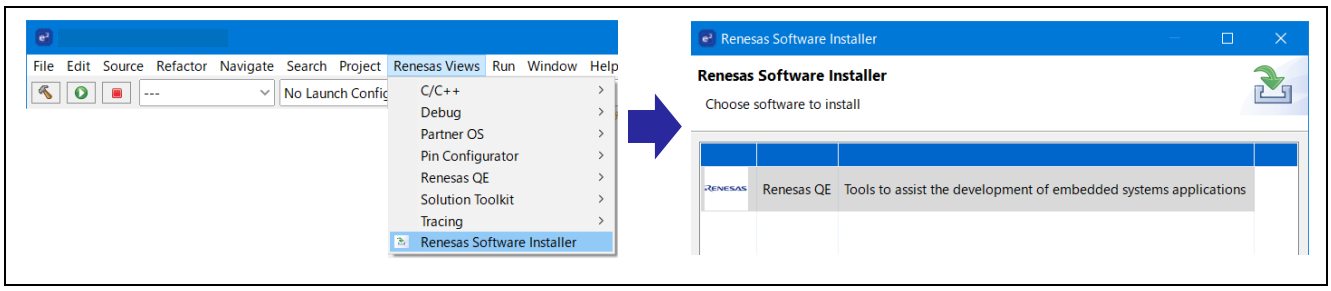


**Figure 2-1 Renesas Software Installer**

When “Install Extensions” is displayed, check “QE for Capacitive Touch” and click “Finish”.

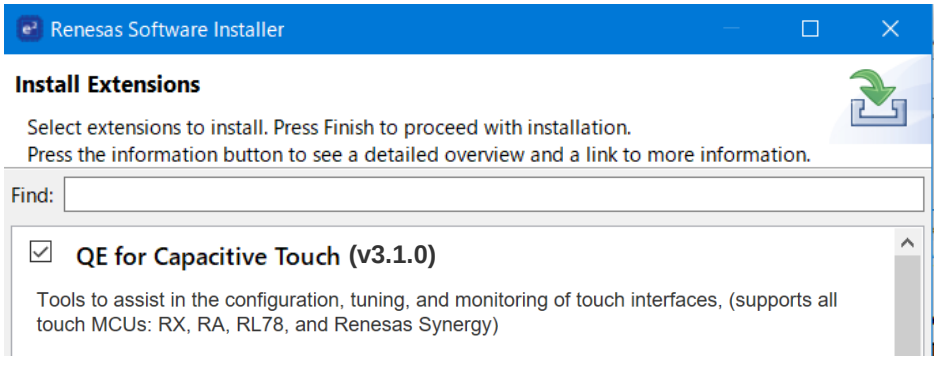


**Figure 2-2 Install Extensions**

Check “Renesas QE for Capacitive Touch” what you want to install and click “Next”.

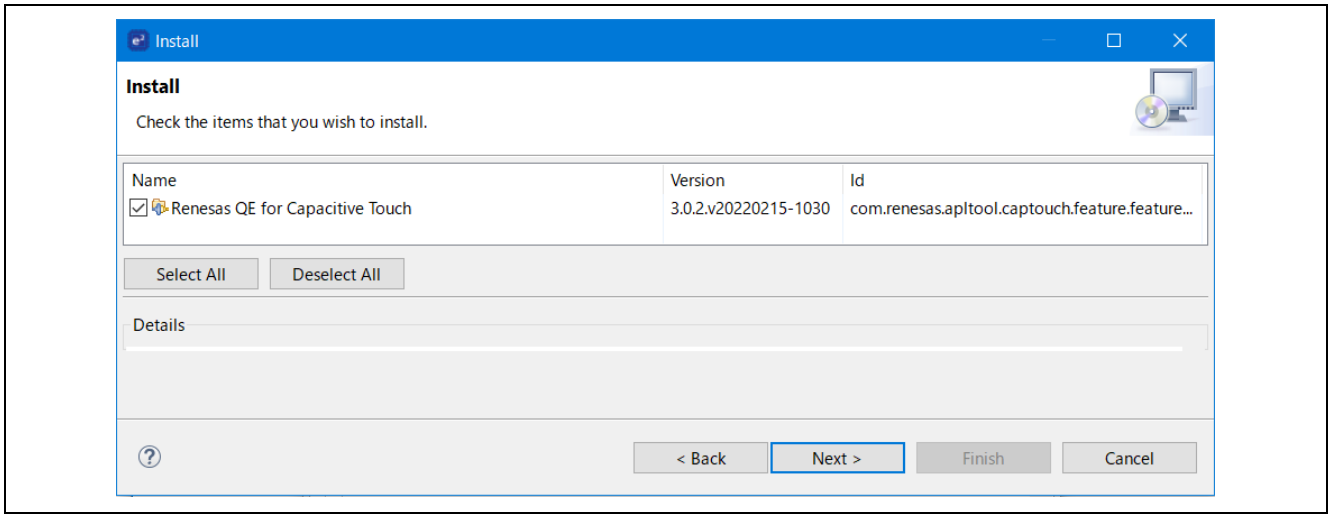


**Figure 2-3 Install of QE**

If you see the following window, set it as trusted signers and the installation will execute.

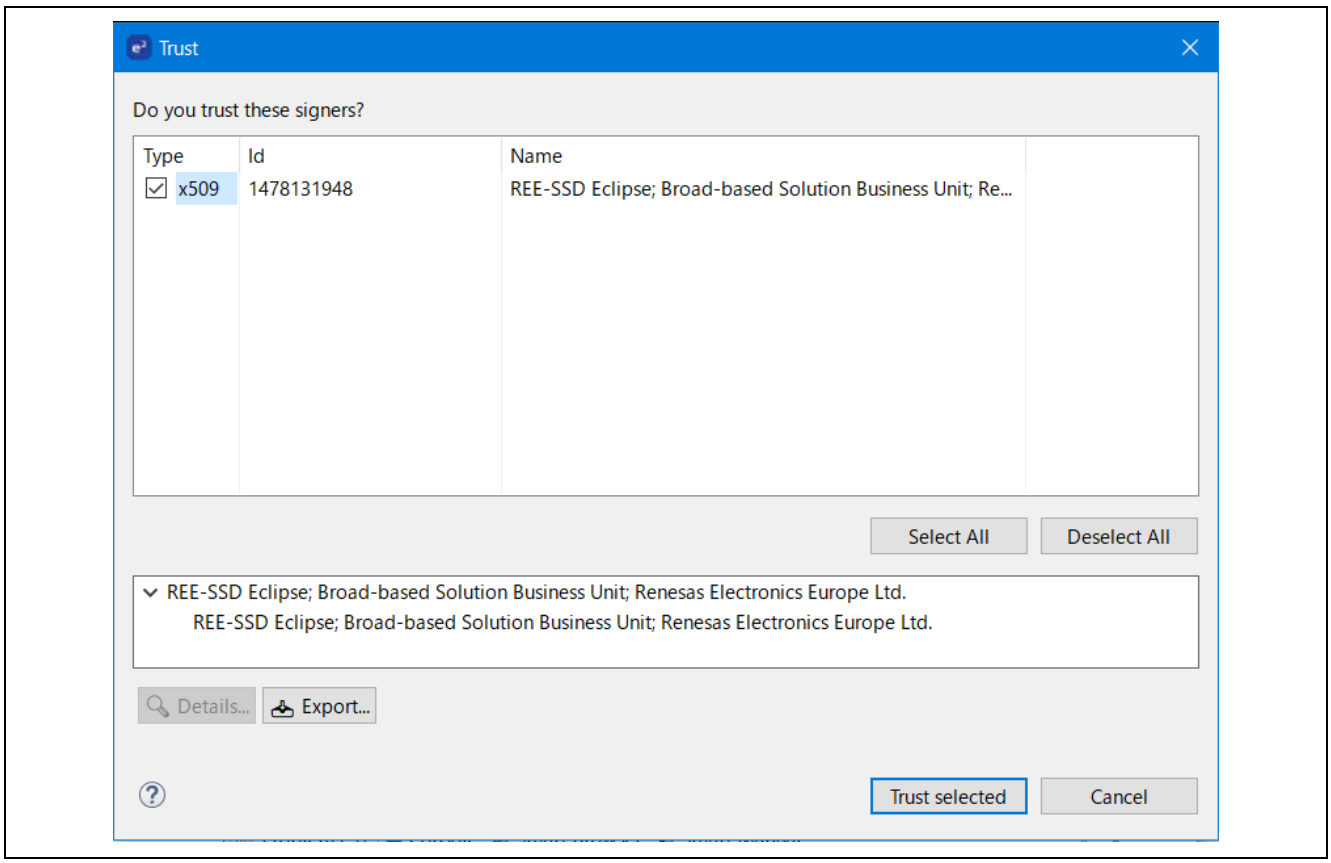


**Figure 2-4 Trust of Tool**

## 2.2 Connection of Target Board

This chapter describes how to connect the PC to the evaluation board and how to supply power. Power can be supplied from the AC adapter or USB. An emulator is used to connect the PC to the evaluation board.

- Figure 2-5 shows the connection diagram when power is supplied from USB.
  - J1 : Connecting an emulator
  - JP2 : 2-3 connecting (short)
  - CN1, CN2 : Connecting the sensor
  - CN5 : Power supply by USB

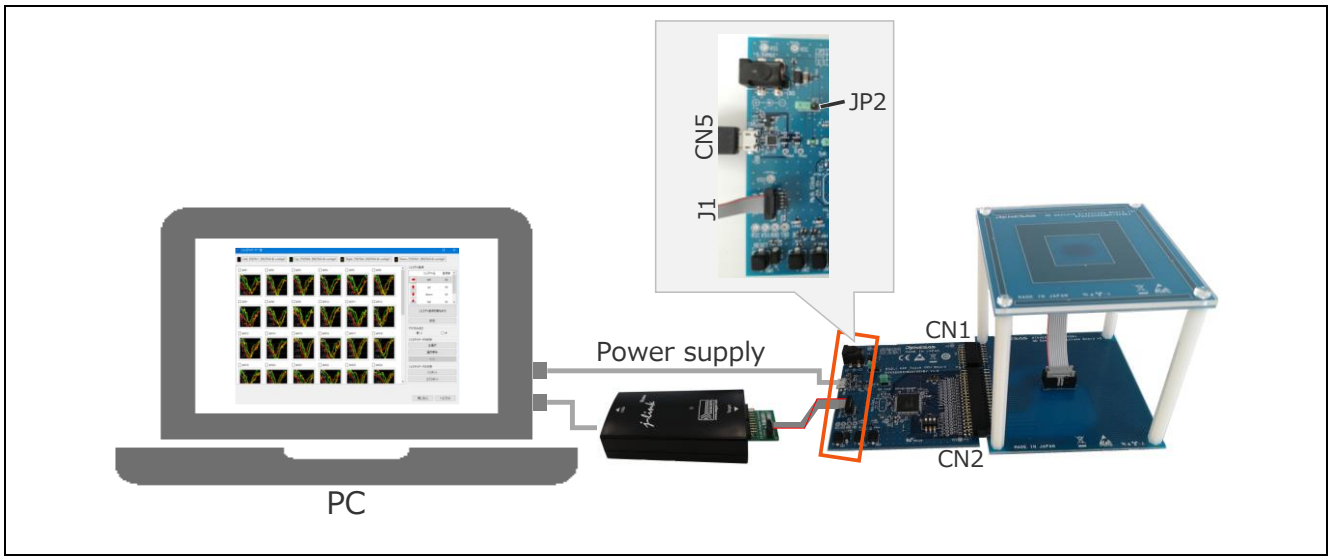


**Figure 2-5 Connection Diagram when Power is supplied from USB**

- Figure 2-6 shows the connection diagram when power is supplied from AC Adapter.
  - J1 : Connecting an emulator
  - JP2 : 1-2 connecting (short)
  - CN1, CN2 : Connecting the sensor
  - CN3 : Power supply by AC adapter

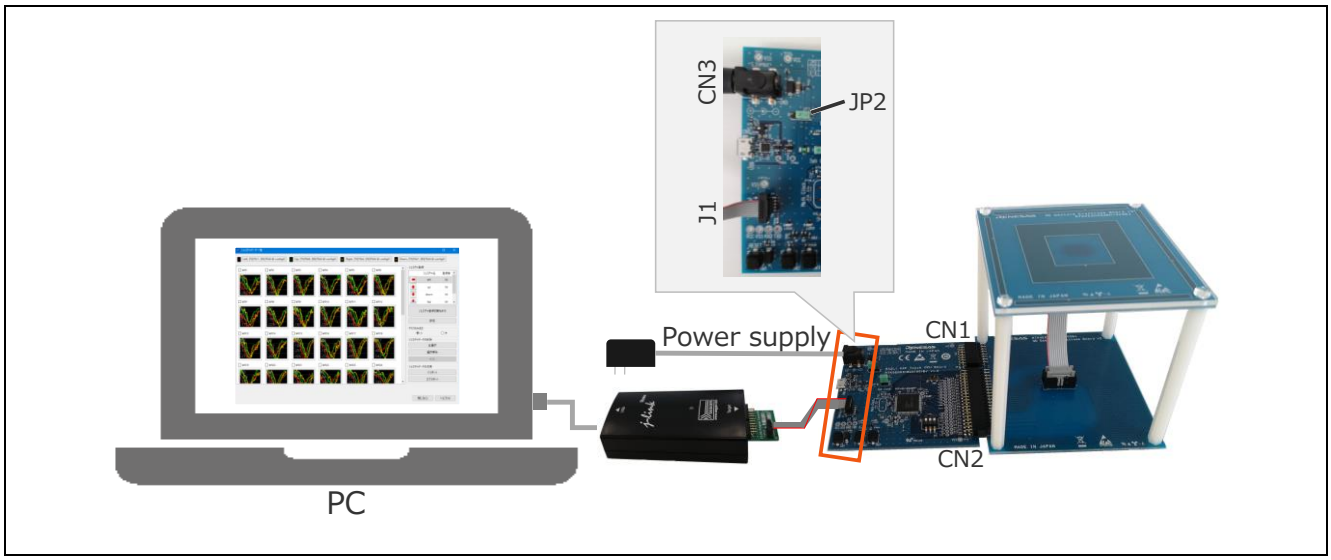


**Figure 2-6 Connection Diagram when Power is supplied from AC Adapter**

## 3. Workflow of the 3D Gesture Recognition

By setting according to the workflow of QE for Capacitive Touch, 3D gesture recognition using AI is possible. Each item is shown in Table 3-1. The chapter numbers in this table are linked to related chapter. Click the chapter number to see how to use it.

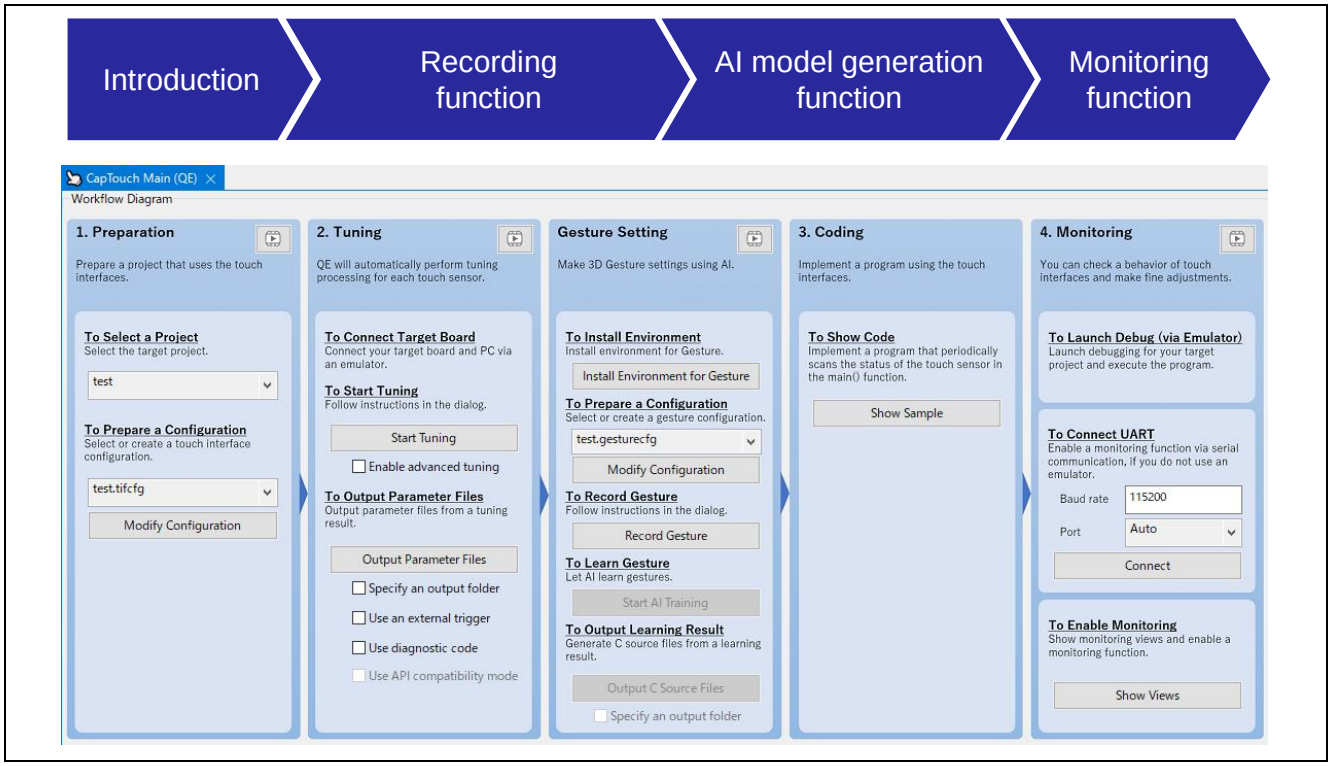


**Figure 3-1 Gesture Recognition Procedure**

**Table 3-1 Items and Contents of QE for Capacitive Touch**

| Items | | Contents | Chapter |
|---|---|---|---|
| Preparation | Selection of a Project | Creation the project and setting to the target project | 4.1.2 |
| | Preparation of a Configuration | Placement of 3D Gesture interface | 4.1.3 |
| | - Setup Touch Interface | Setting terminals according to sensor specifications | 4.1.3.1 |
| Tuning | Start Tuning | Adjustment the electrode sensitivity of the sensor | 4.2.1 |
| | Output Parameter Files | Output the C source files used for monitoring | 4.2.2 |
| Gesture Setting | Install Environment for Gesture | Apply e-AI Solution etc. | 4.3.1 |
| | Preparation of a Configuration | Setting the gesture to be recognized | 4.3.2 |
| | Recording Gesture | Register the gesture made on the sensor | 4.3.3 |
| | Learning Gesture | Learning recognized gestures with AI | 4.3.4 |
| | Output Learning Result | Output the C source files used for monitoring | 4.3.5 |
| Coding | Showing the Code Example | Output the C source files used for monitoring | 4.4.1 |
| | - Calling Gesture Main Function | Adding the call code of the generated function | 4.4.1.1 |
| Monitoring | Launching Debugging | Starting the operation of the implemented program | 4.5.1 |
| | Enable Monitoring | Launching a monitoring environment | 4.5.2 |
| | - Recognition Gesture | Monitoring gesture recognition status | 4.5.2.1 |
| | - Misrecognition Gesture Registration | Additional registration of misrecognition gesture data | 4.5.2.2 |

# 4. Examples of the 3D Gesture Recognition

This chapter describes an example of creating a program that recognizes two types of gestures, as shown below. The explanation is given below according to the items of QE for Capacitive Touch.

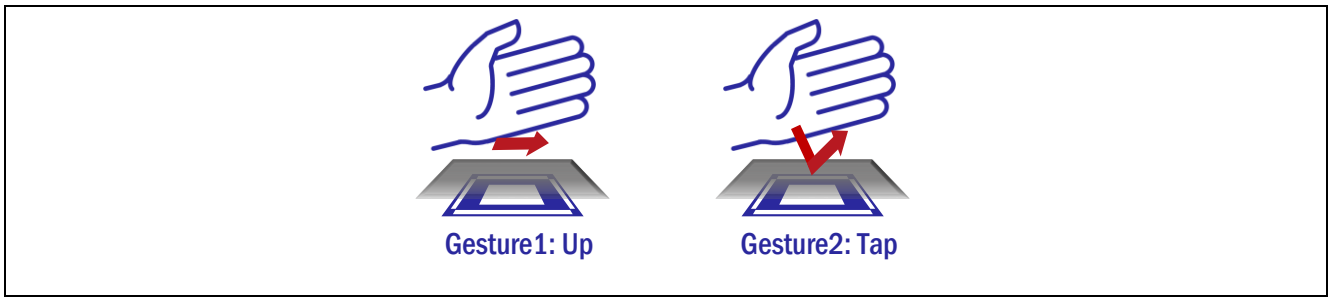


**Figure 4-1 4. Examples of the 3D Gesture Recognition**

## 4.1 Preparation

### 4.1.1 Launch the QE for Capacitive Touch

Select “CapTouch Main (QE)” from the e² studio menu as shown below to display the CapTouch Main window of QE for Capacitive Touch. At this state, the “Gesture Setting” item is not displayed.

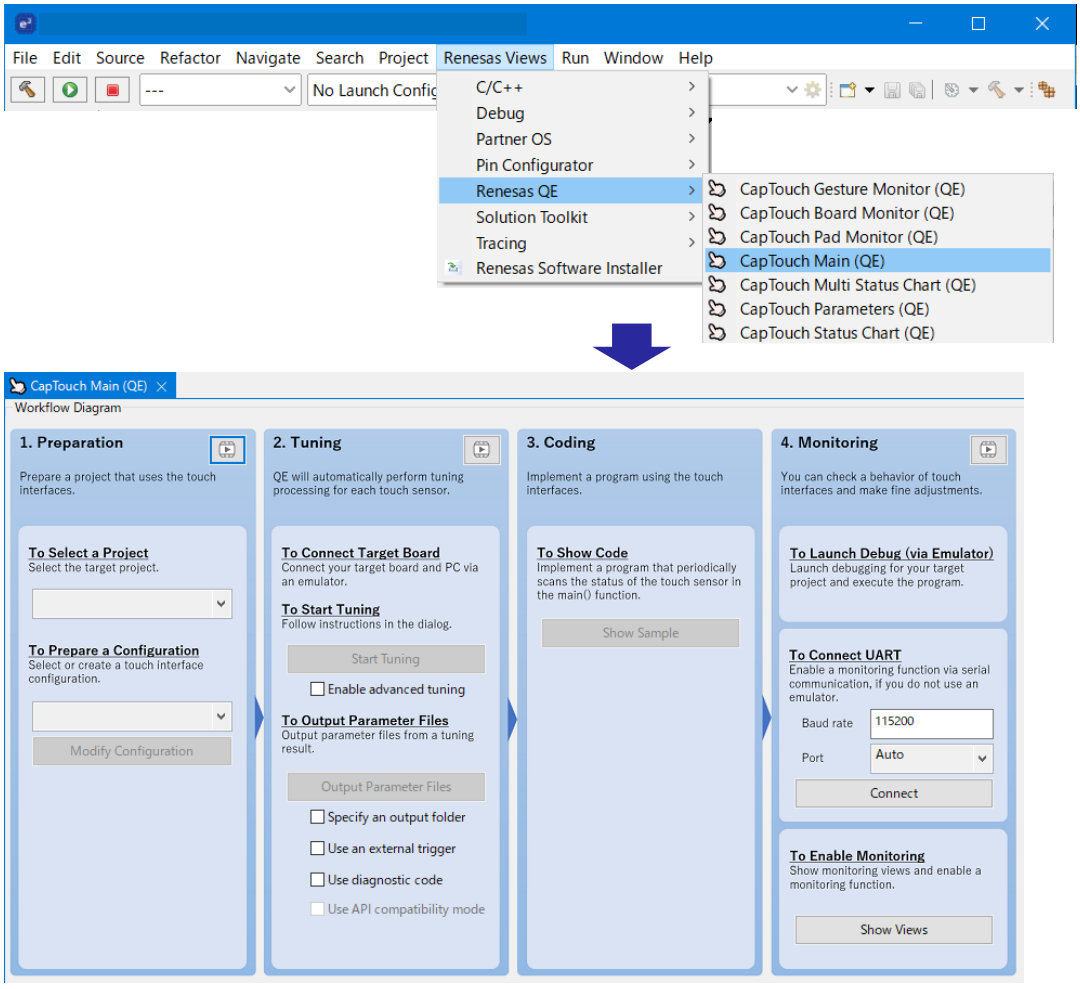


**Figure 4-2 Select CapTouch Main Window**

### 4.1.2 Selection of a Project

Create a new project in e² studio and select a project as shown below.

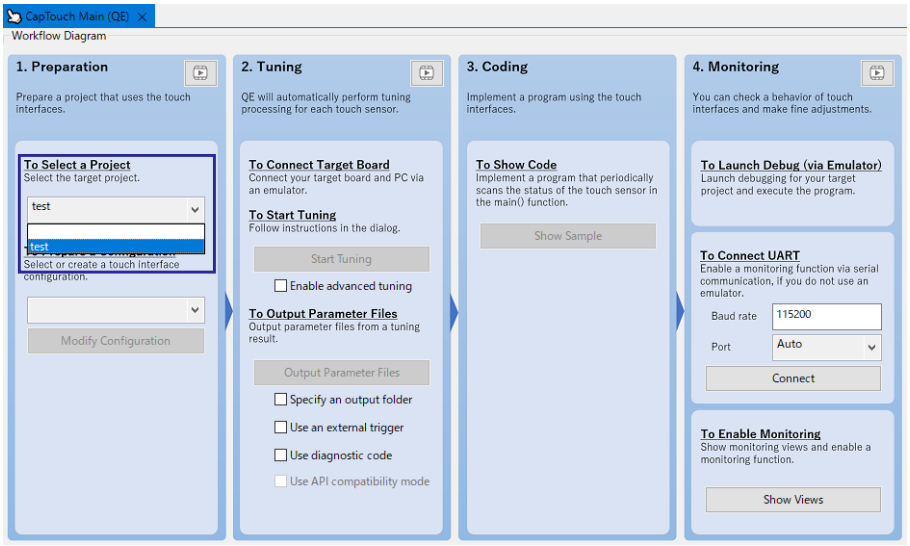


**Figure 4-3 Selection of a Project**

#### 4.1.2.1 Project Creation Example

In this chapter, Figure 4-4 to Figure 4-8 show examples of creating a new RA2L1 project.

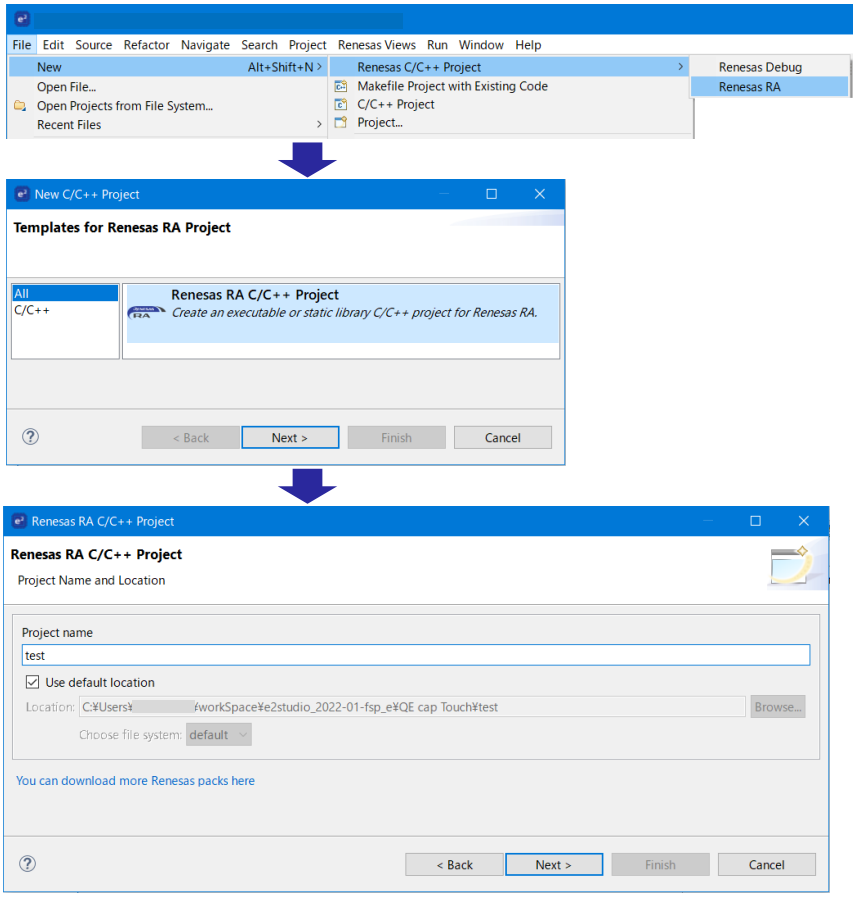


**Figure 4-4 Creating a New Project**

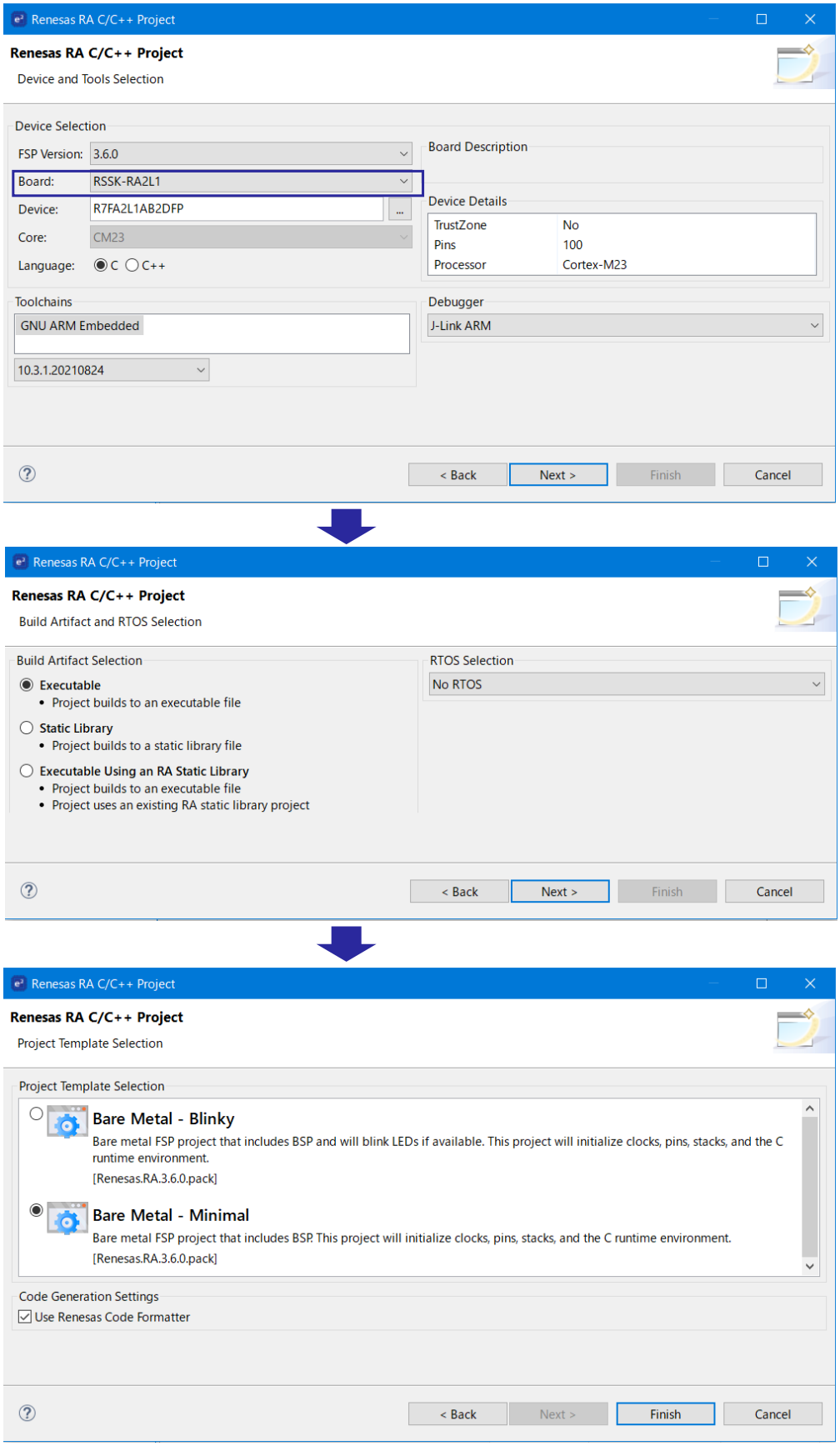


**Figure 4-5 Setup the Project**

Change the Heap size of the BSP.

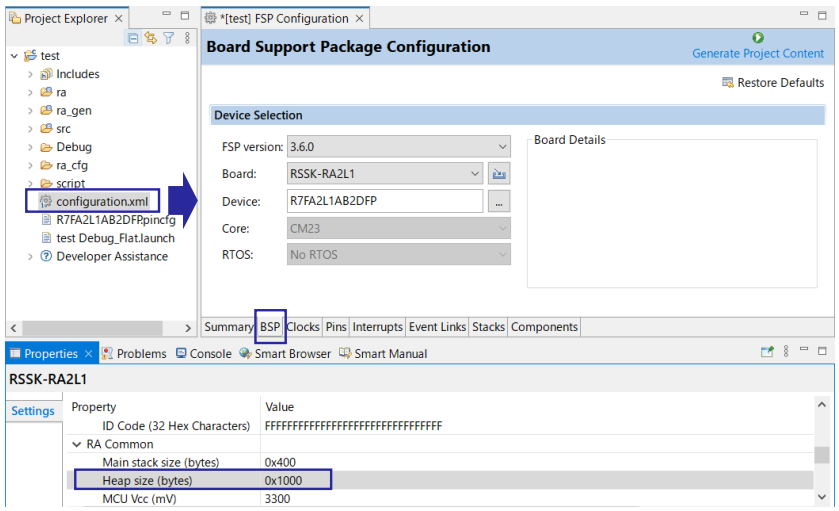


**Figure 4-6 Changing the Heap Size**

Add Touch middleware for 3D gesture recognition.

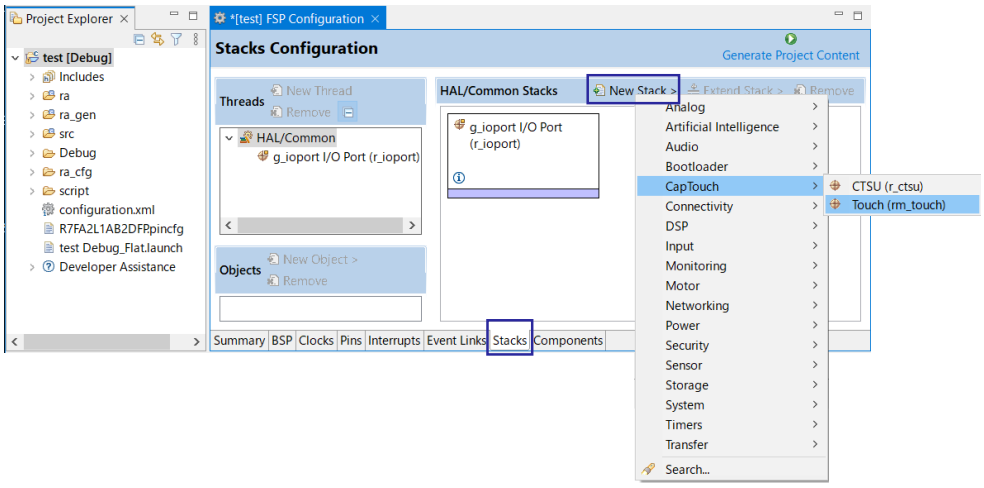


**Figure 4-7 Adding Touch Middleware**

Generate a code with FSP.

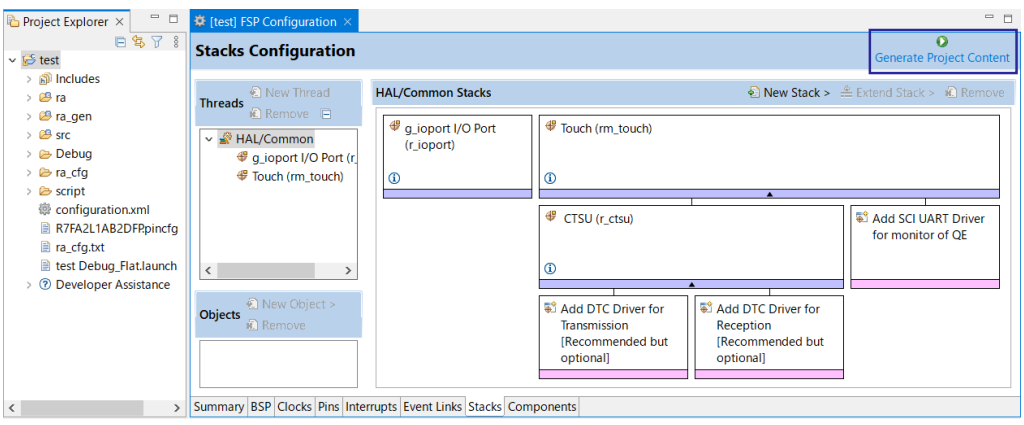


**Figure 4-8 Generate a Code**

### 4.1.3 Preparation of a Configuration

Click “Create a new configuration” as shown below to display the Create Configuration of Touch Interfaces window. After placing the 3D Gesture Interface in the configuration of touch interface, the “Gesture Setting” item will be displayed in the CapTouch Main window.

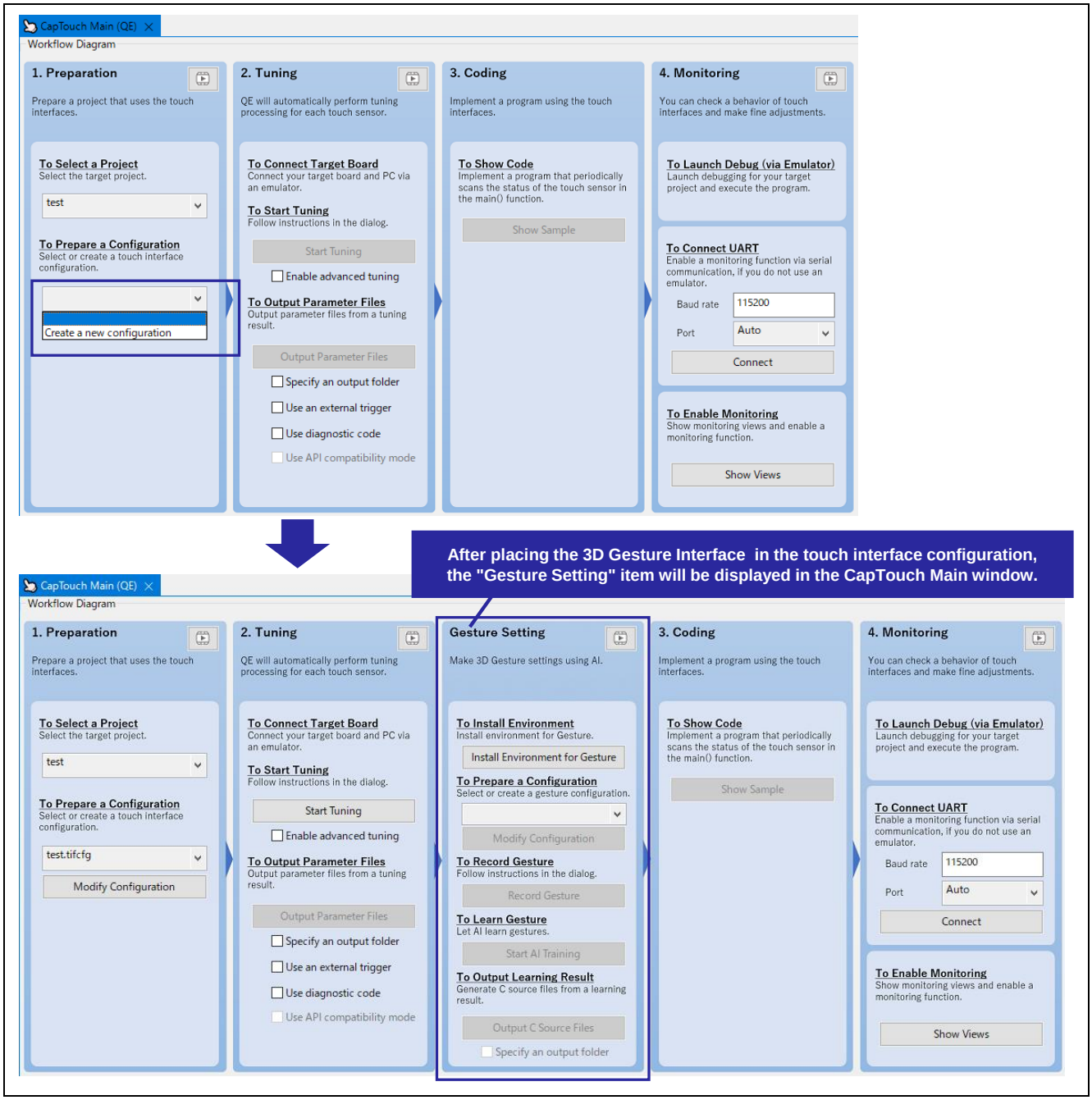


**Figure 4-9 Creation a New Configuration**

When the Create Configuration of Touch Interfaces window appears, place the 3D Gesture Interface as follows:

- Select “Mutual Capacitance” of the Capacitance Type – (1)
- Click “3D Gesture (AI)” – (2)
- Place 3D Gesture Interface – (3)
- Click “3D Gesture (AI)” again, cancel placement mode – (4)
- Click on the placed 3D Gesture Interface – (5)
- Click “Setup Touch I/F” to display the Setup Touch Interface window. – (6)

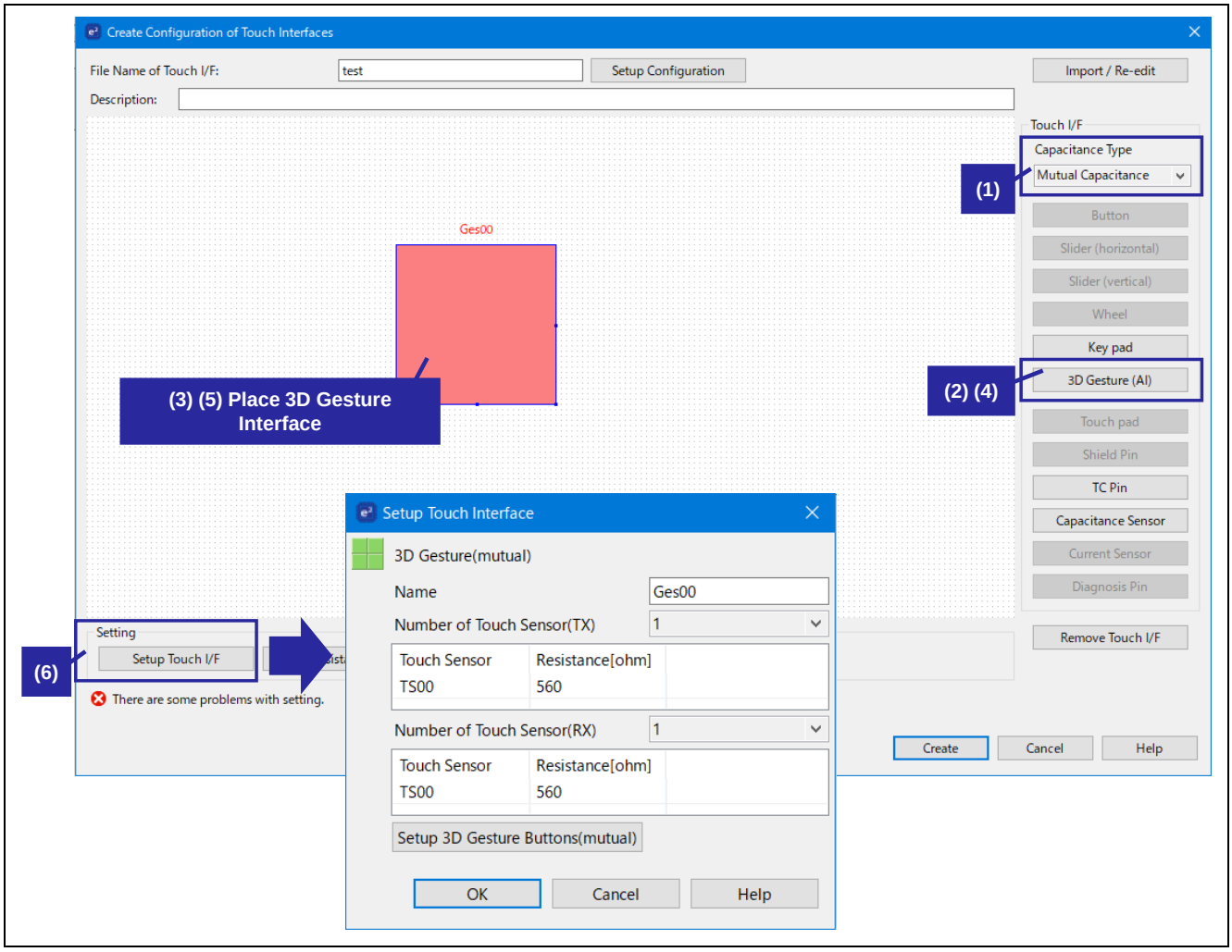


**Figure 4-10 Placement 3D Gesture Interface**

### 4.1.3.1 Setup Touch Interface

When the Setup Touch Interface window is displayed, follow the steps below to set the sensor terminals. If the touch interface is set correctly, the target interface will turn green.

The sensor terminals must be set according to the specifications of the sensor board to be used.

- Set the number of touch sensors for transmission to “4” – (1)
- Set TS02, TS08, TS09 and TS10 to the touch sensor terminal for transmission – (2)
- Set the number of touch sensors for receiving to “1” – (3)
- Set TS18 to the touch sensor terminal for receiving – (4)
- Click “Setup 3D Gesture Buttons(mutual)” – (5)
- Set the button name – (6) – optional

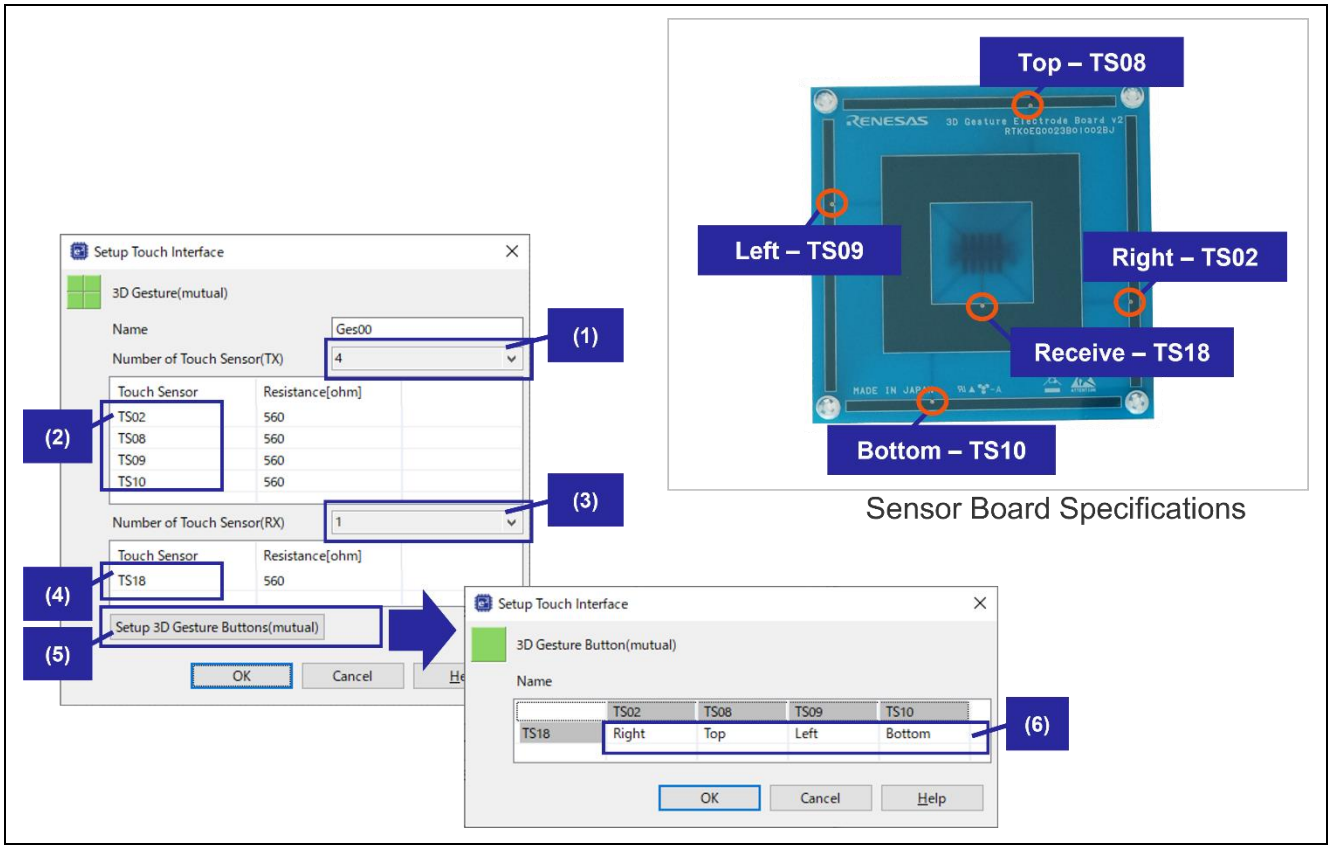


**Figure 4-11 Setup Touch Interface**

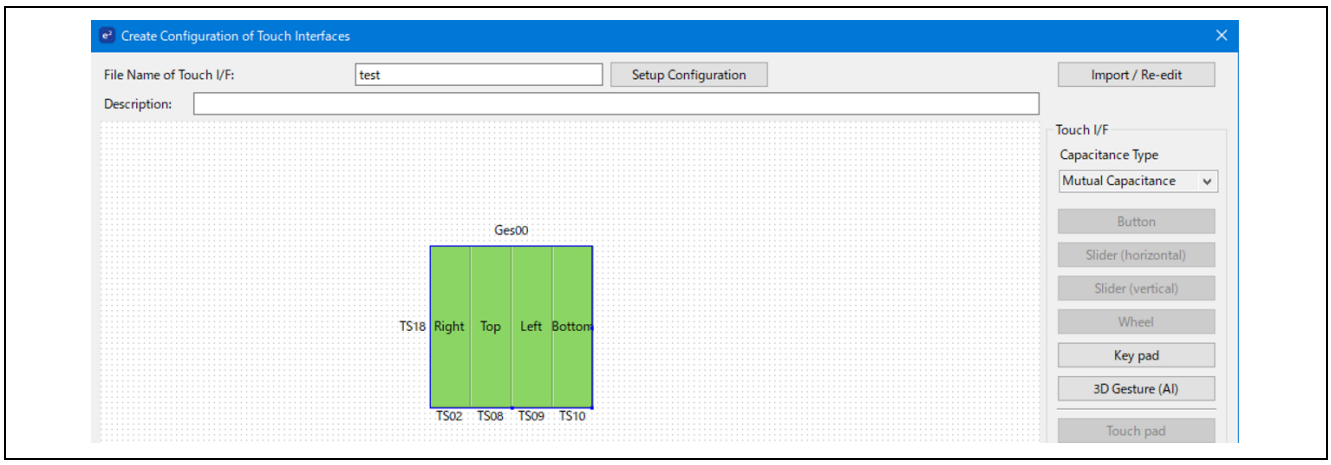


**Figure 4-12 Success in “Setup Touch Interface”**

## 4.2 Tuning

When adjusting the electrode sensitivity of the sensor, It is necessary to connect the PC and the evaluation board. See Chapter 2.2 for connection methods.

- Precautions when sensitivity adjustment
  - Place the evaluation board on an insulator, etc.
  - If you place it directly on conductors such as iron plates, you will not be able to measure correctly.
  - Do not hold anything over the sensor until instructed.

### 4.2.1 Start Tuning

Follow the procedure below to adjust the sensitivity. If the absolute threshold value of the adjustment result is extremely large, the sensor terminal settings made in Chapter 4.1.3.1 may not be correct.

- Click “Start Tuning”
- When the message in Figure 4-14 is displayed, hold your hand over the specified sensor and click the key.
- Click “Continue the Tuning Process” when the message in Figure 4-15 is displayed.

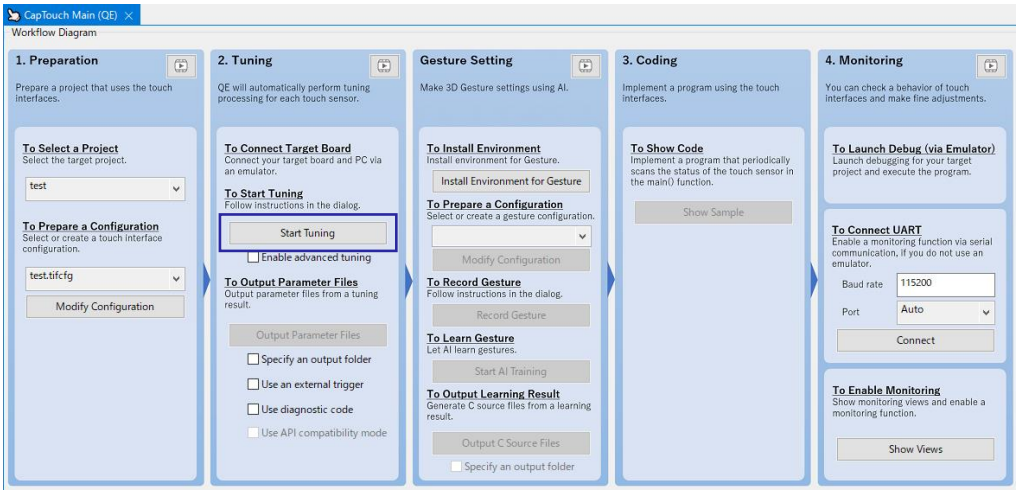


**Figure 4-13 Start Tuning**

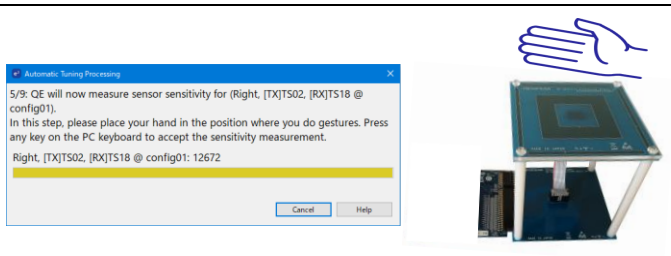


**Figure 4-14 During Sensitivity Adjustment**

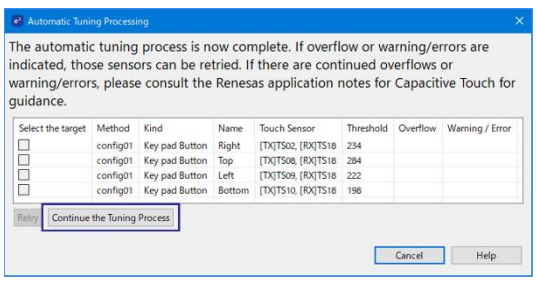


**Figure 4-15 Results of Sensitivity Adjustment**

### 4.2.2 Output Parameter Files

Click “Output Parameter Files” as shown in the figure below to output the C source files of the adjustment result. The files output in this chapter are stored in the qe_gen folder and are used in the coding process.

For the C source files output by QE for Capacitive Touch, refer to Chapter 5.

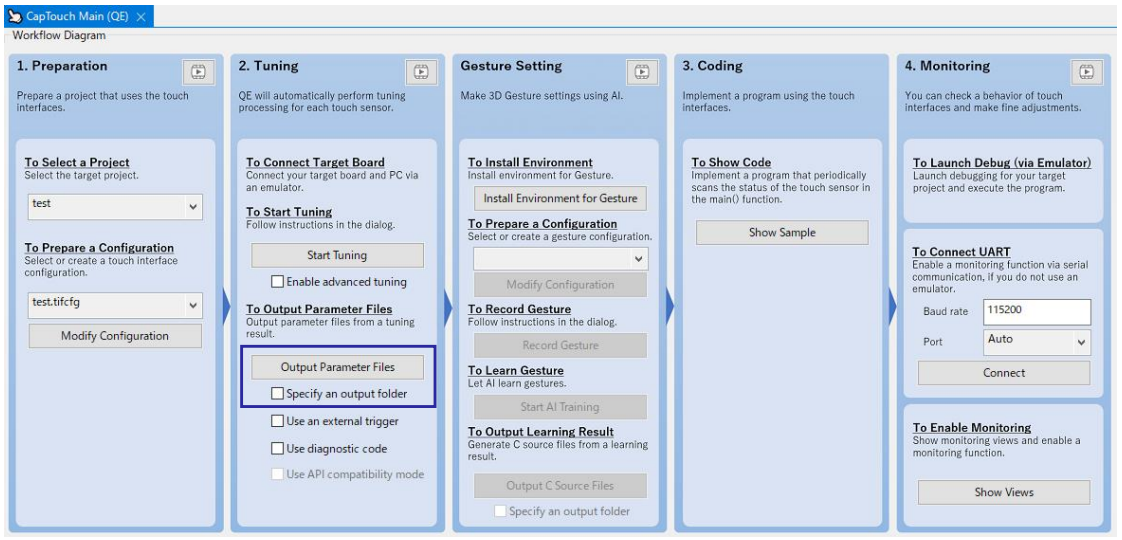


**Figure 4-16 Output Parameter Files**

## 4.3 Setup Gesture

### 4.3.1 Install Environment for Gesture

If the Environment for Gesture has not been installed or the license needs to be renewed, an error mark will be displayed.

Click “Install Environment for Gesture” as shown below to install the gesture environment. Once you have introduced the Environment for Gesture, you only need to renew your e-AI license on a regular basis.

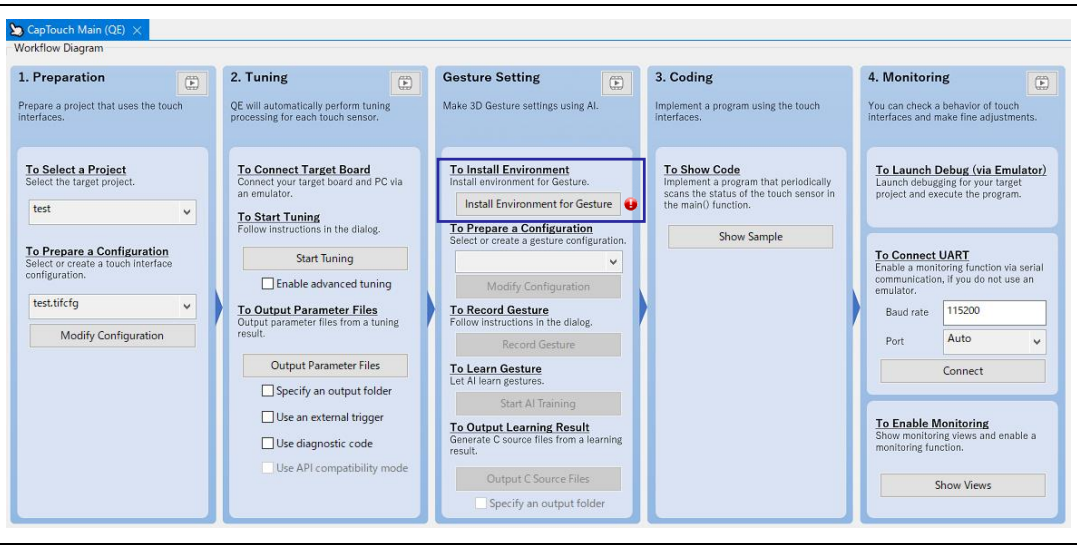


**Figure 4-17 Install Environment for Gesture**

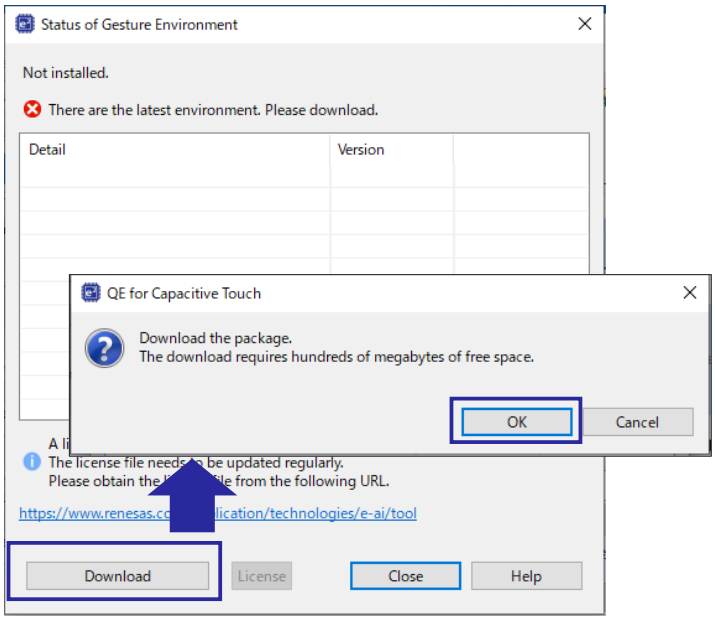


**Figure 4-18 Download of Gesture Environment**

When the installation of the Gesture Environment is completed, a list of installed tools will be displayed.

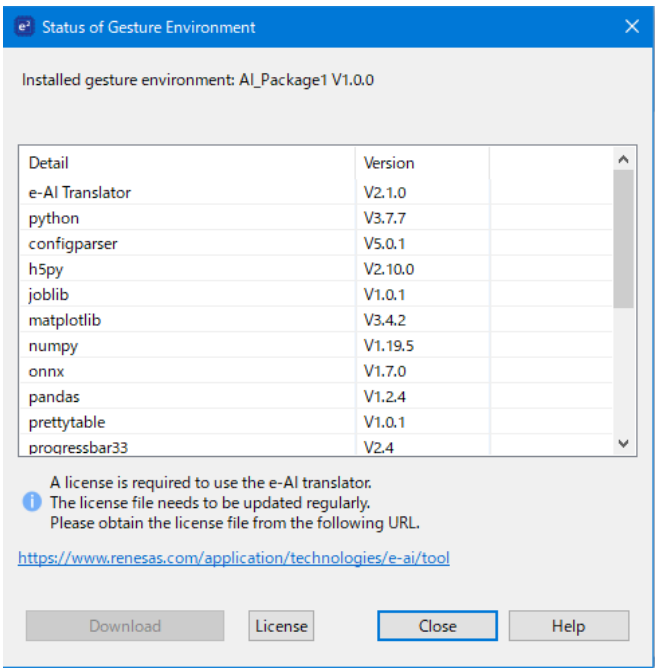


**Figure 4-19 List of Installed Tools**

### 4.3.1.1 Update of the e-AI License File

The e-AI license requires regular renewal. Obtain the license file for “e-AI Translator License Extension QE for Capacitive Touch version” from the Renesas website. Place the license file in the following location in the download area of your support folder.

<Support folder download area>¥QE¥AI_Package1¥Tools¥eAITranslator_<version> ¥bin

If you have not changed the support folder from the default settings, place it in the following location.

C:¥Users¥<user>¥.eclipse¥com.renesas.platform_download¥QE¥AI_Package1¥Tools¥eAITranslator_<version> ¥bin

The location of the support folder download area can be found in “e² studio download area” of “Support Folders” from the help of e² studio as shown in Figure 4-20. The support folder can also be changed as shown in Figure 4-21.

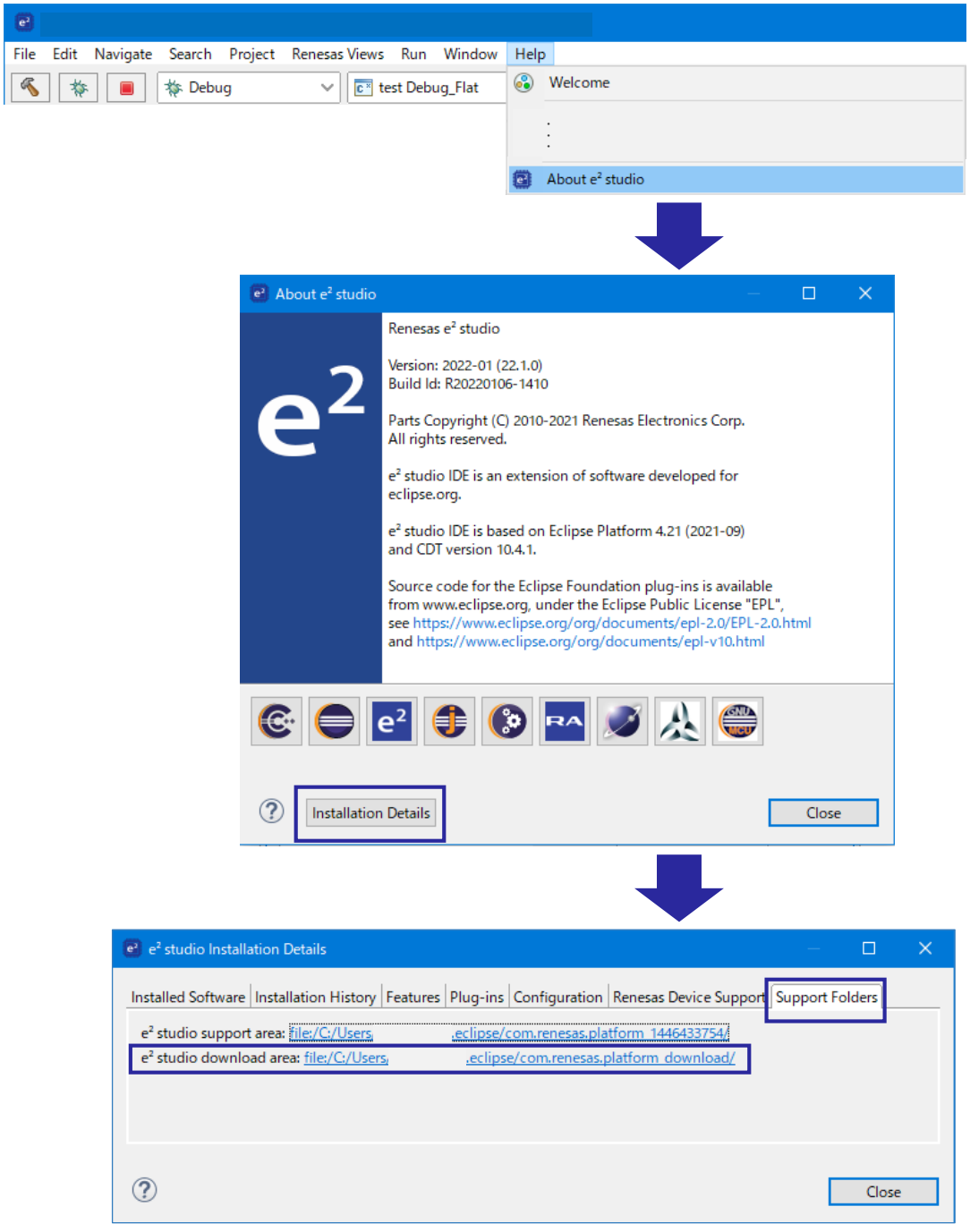


**Figure 4-20 Download Area of the Support Folder**

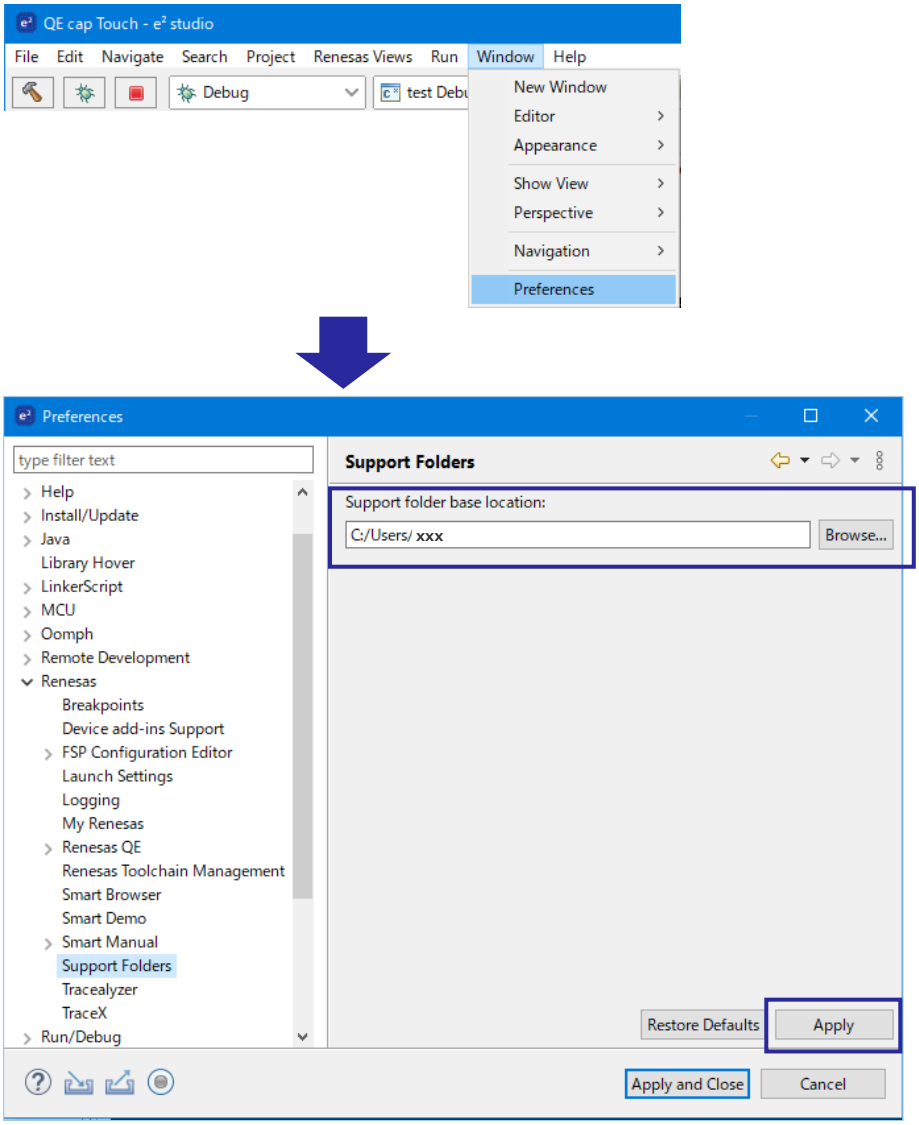


**Figure 4-21 Change the Support Folder**

### 4.3.2 Preparation of a Configuration

Follow the steps below to create a gesture configuration. Since 4 types of gestures are registered in the initial setting, unnecessary gestures are removed and necessary gestures are registered from the gesture list.

- Select “Create a new configuration” to display the Create Gesture Configuration window.
- Check the gestures you do not want to use in the gesture configuration and click “Remove > >”.
- Check the gesture you want to use in the gesture list and click “< < Add”.

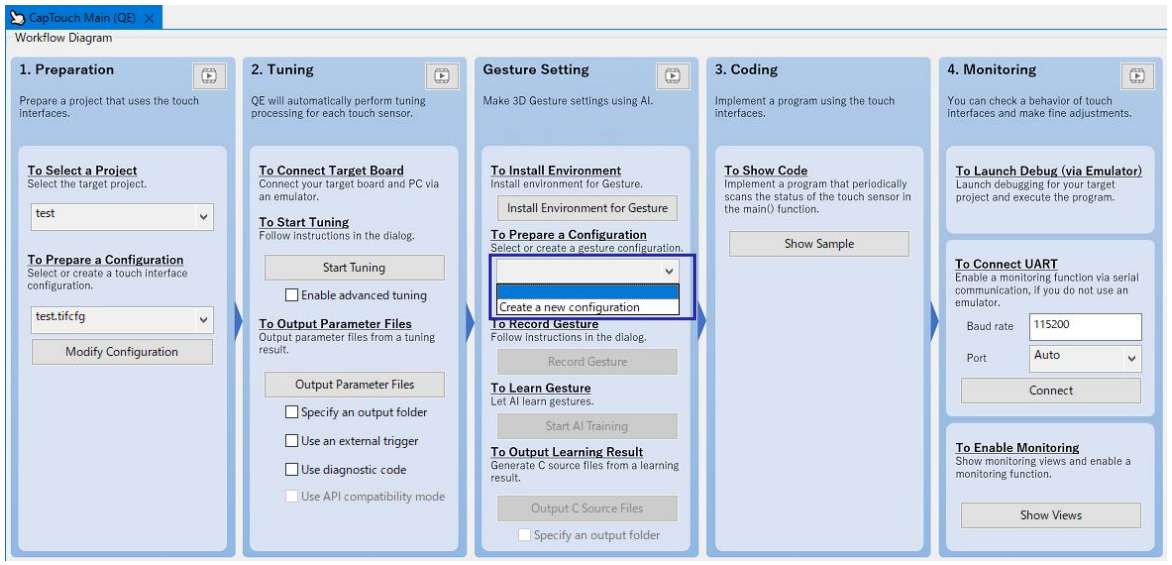


**Figure 4-22 Create a New Configuration for Gesture**

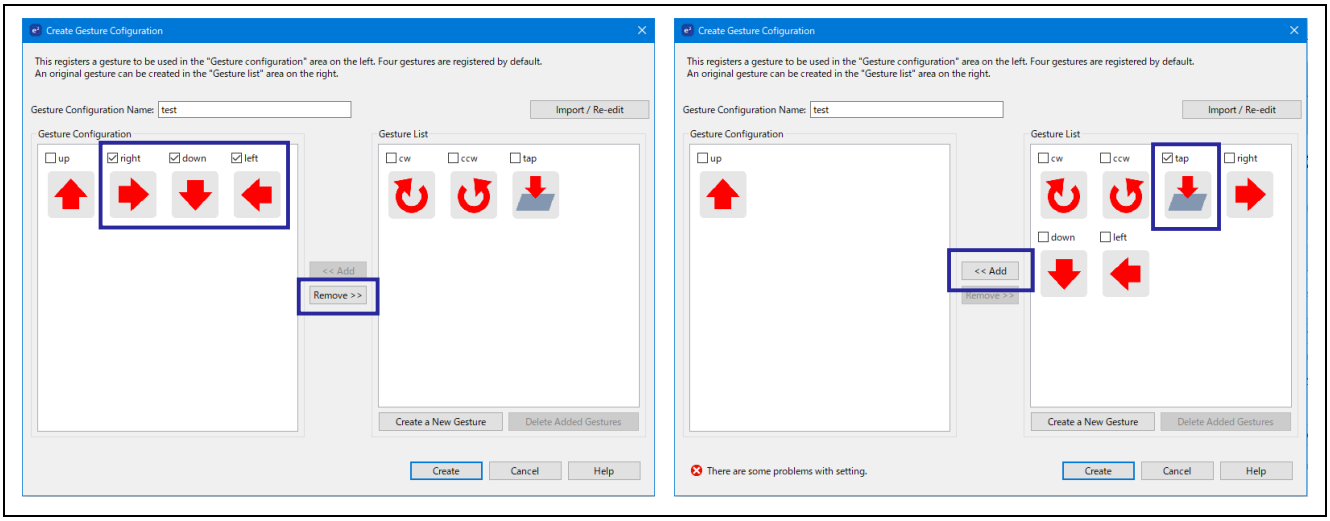


**Figure 4-23 Removing and Adding for Gestures**

### 4.3.3 Recording Gesture

Follow the steps below to register the gestures you made on the sensor. It is recommended to register 50 or more gesture data for one type of gesture. By changing the angle of the hand or registering with the right and left hands, the gesture recognition accuracy will increase.

- Click “Record Gesture” to display the Create Data List window.
- Click “Setting” and check “Notify by sound when gesture registration is completed” -optional
- From “Recording Gesture”, select one “Gesture Name” you want to register.
- Click “Start Recording Gesture” to display the Recording Your Gesture window.
- Make a gesture on the sensor.

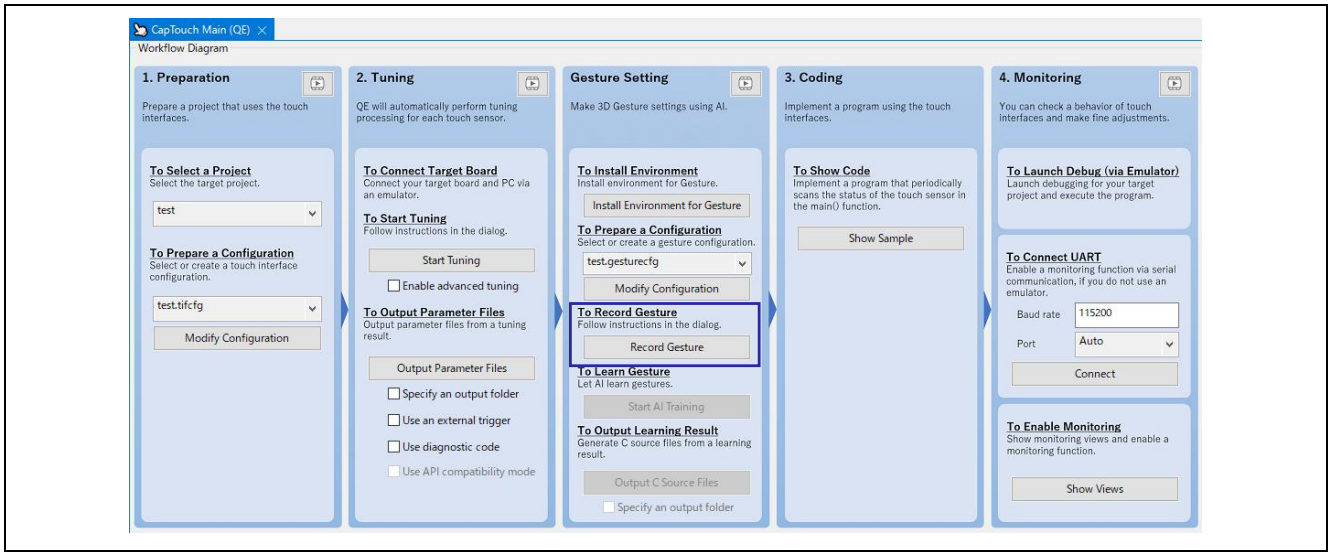


**Figure 4-24 Recording Gesture**

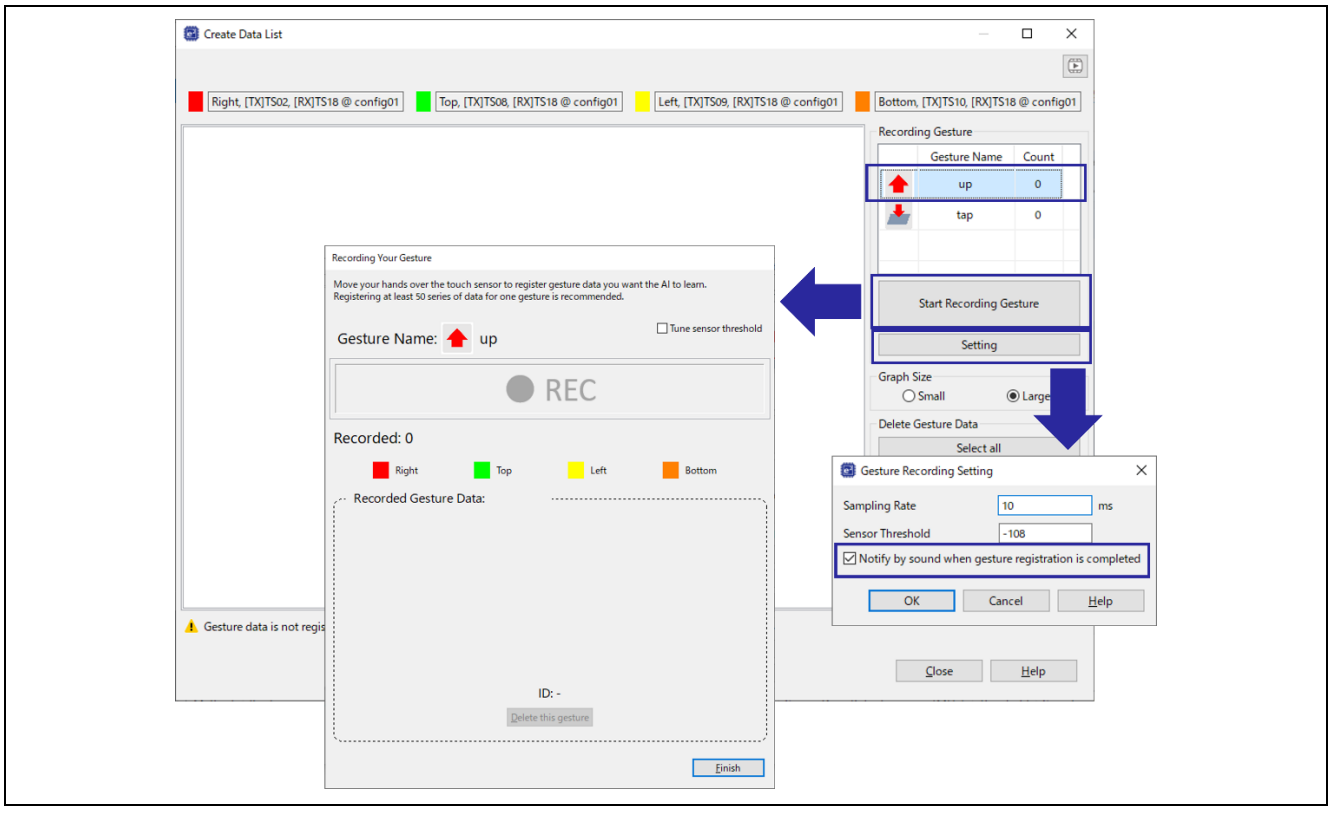


**Figure 4-25 Start Recording Gesture**

When a gesture is detected, REC turns red and the registered gesture data is displayed.

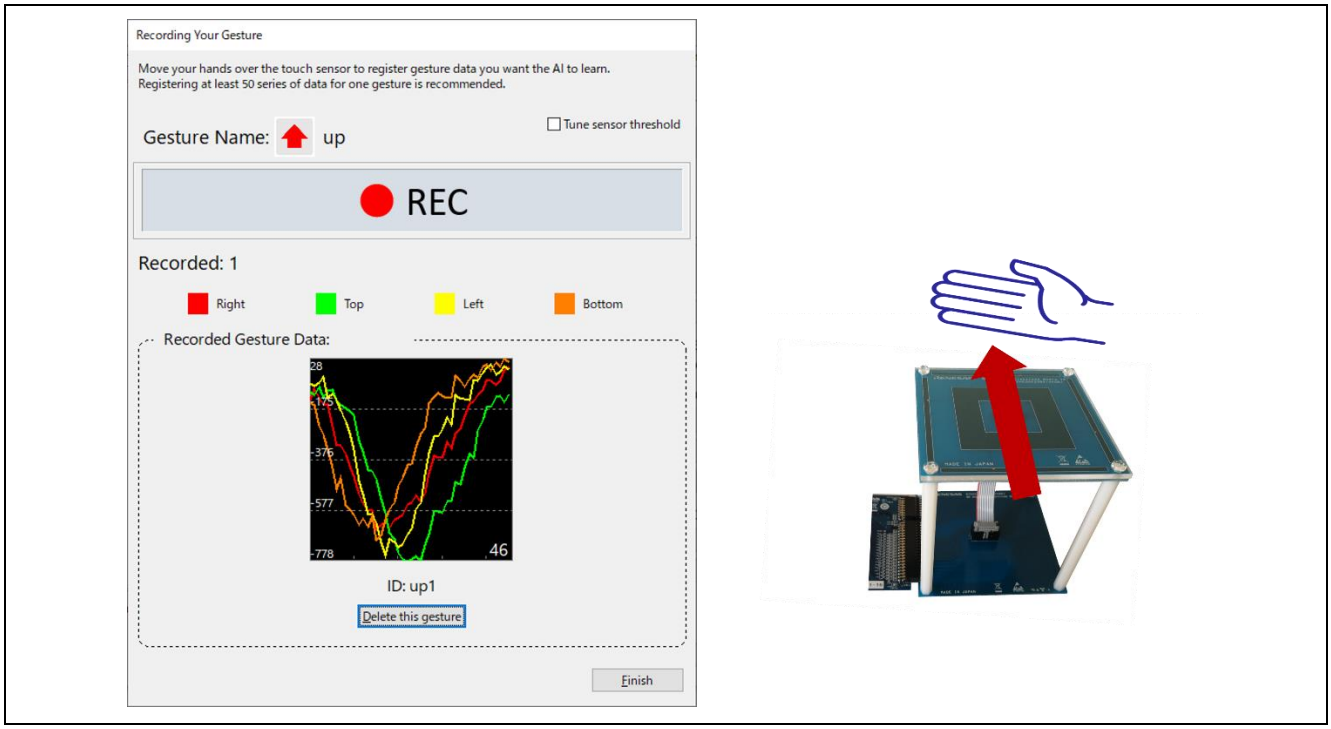


**Figure 4-26 Registration the Gesture Data**

When you finish the gesture registration, a list of registered gesture data will be displayed. Unnecessary data can be checked and deleted.

You can export the registered gesture data list in CSV file format. You can also import gesture data registered by multiple people, or you can import gesture data registered in another project.

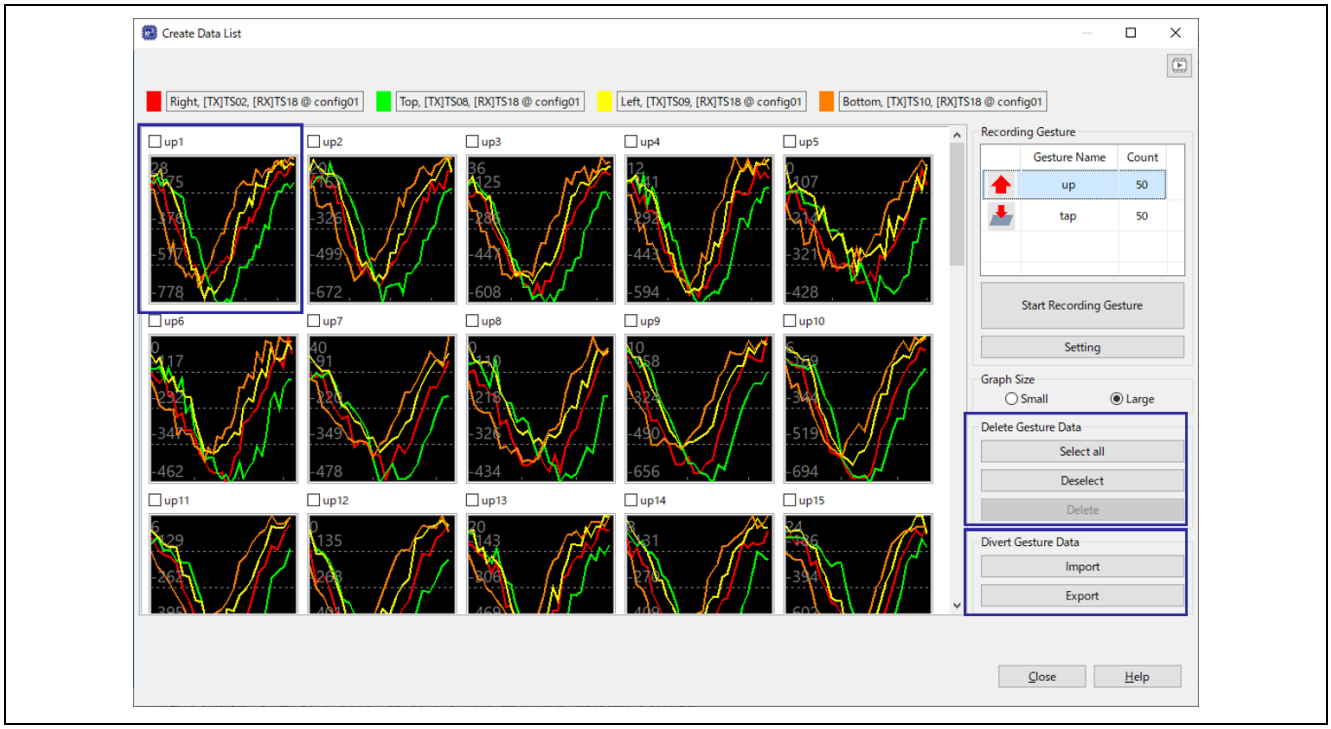


**Figure 4-27 List of Registered Gesture Data**

The window shown in Figure 4-28 may appear when you click “Record Gesture”. The frequency of the peripheral module clock can be checked in FSP Configuration as shown in Figure 4-29.

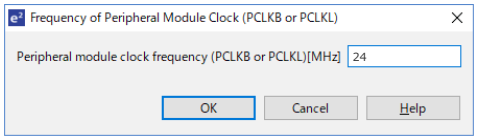


**Figure 4-28 Setting of the Clock Frequency**

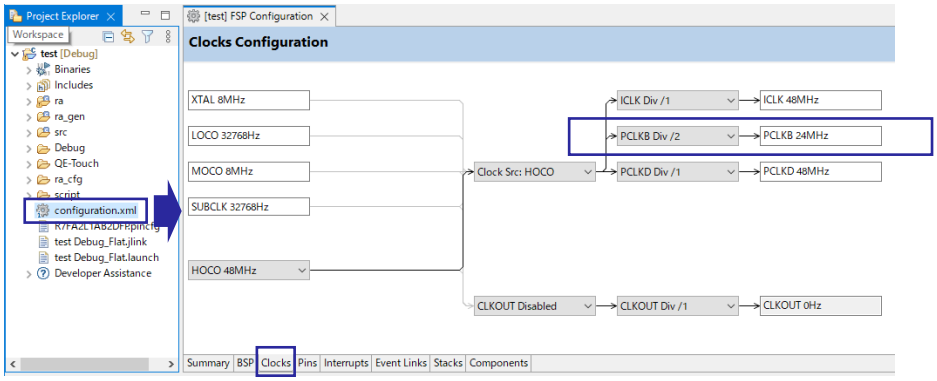


**Figure 4-29 Confirmation of the Clock Frequency**

### 4.3.3.1 Tuning the Sensor Threshold

If the REC does not turn red, the height at which you make the gesture may not suitable. If you check “Tune sensor threshold”, you can check the sensitivity while changing the height of your hand. The height at which the REC turns red when you bring your hand close to the sensor is the height at which you can register the gesture. After checking the height, click “Finish”.

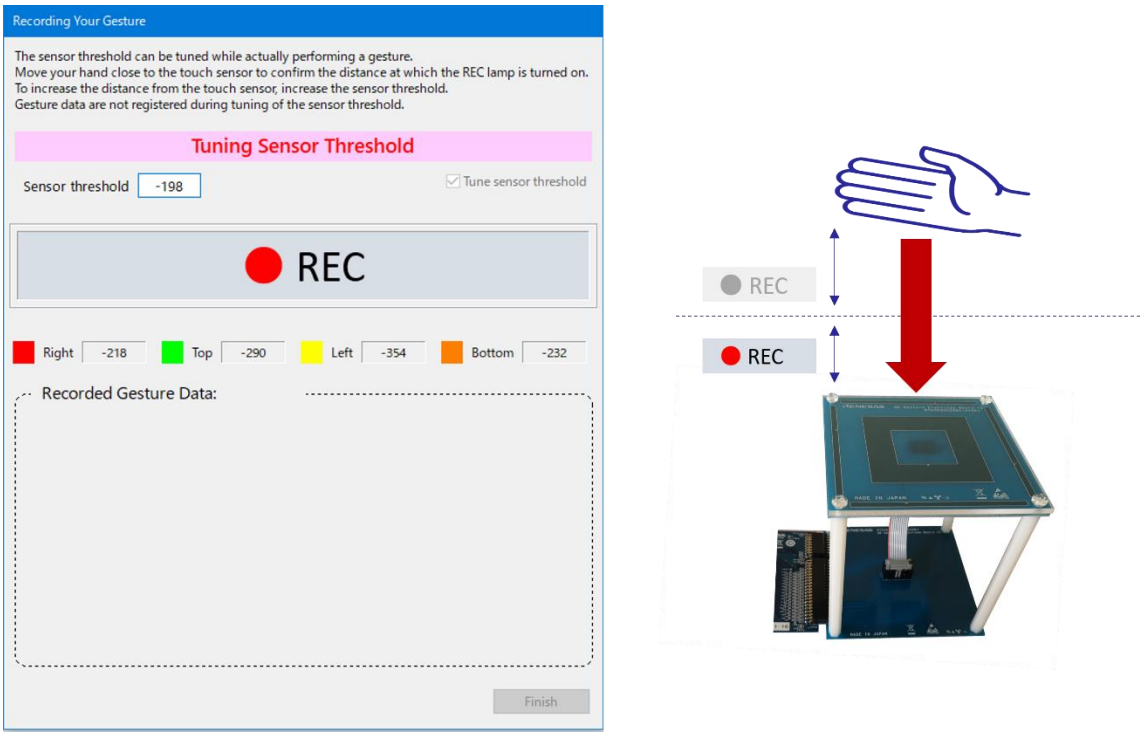


**Figure 4-30 Tuning Sensor Threshold**

### 4.3.4 Learning Gesture

Follow the steps below to learn AI for registered gestures. Make sure AI learning results are 90% or more accurate. You can improve the accuracy of AI learning by increasing the learning data and deleting the failure data. If the accuracy does not exceed 90%, review the registered gesture data.

- Click “Start AI Training” to display the Gesture Training window
- In case to set AI training, click “Setting” Note — Optional
- Click “Start Training”

Note: The neural network size affects the memory size required for AI learning. For some devices of RL78, only the neural network size [Small] can be used due to the limitation of the built-in ROM / RAM size.

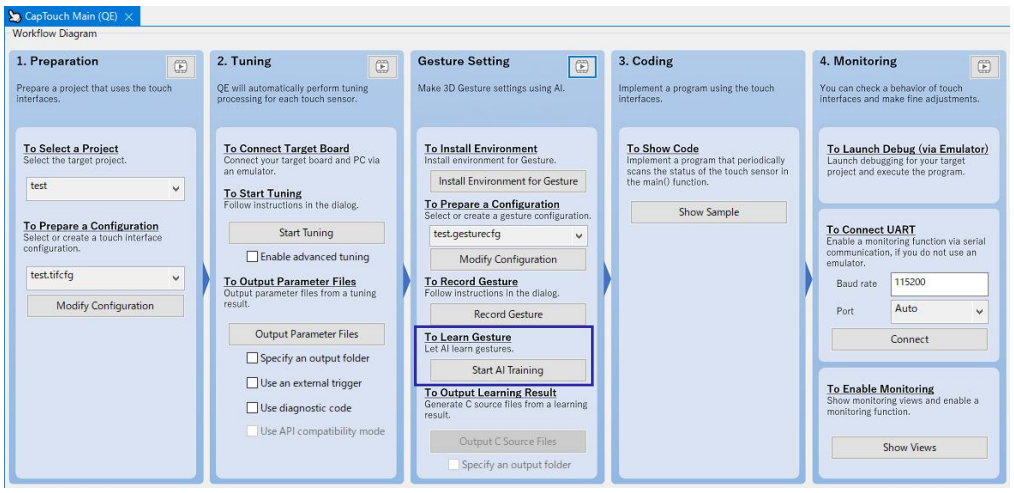


**Figure 4-31 Start AI Training**

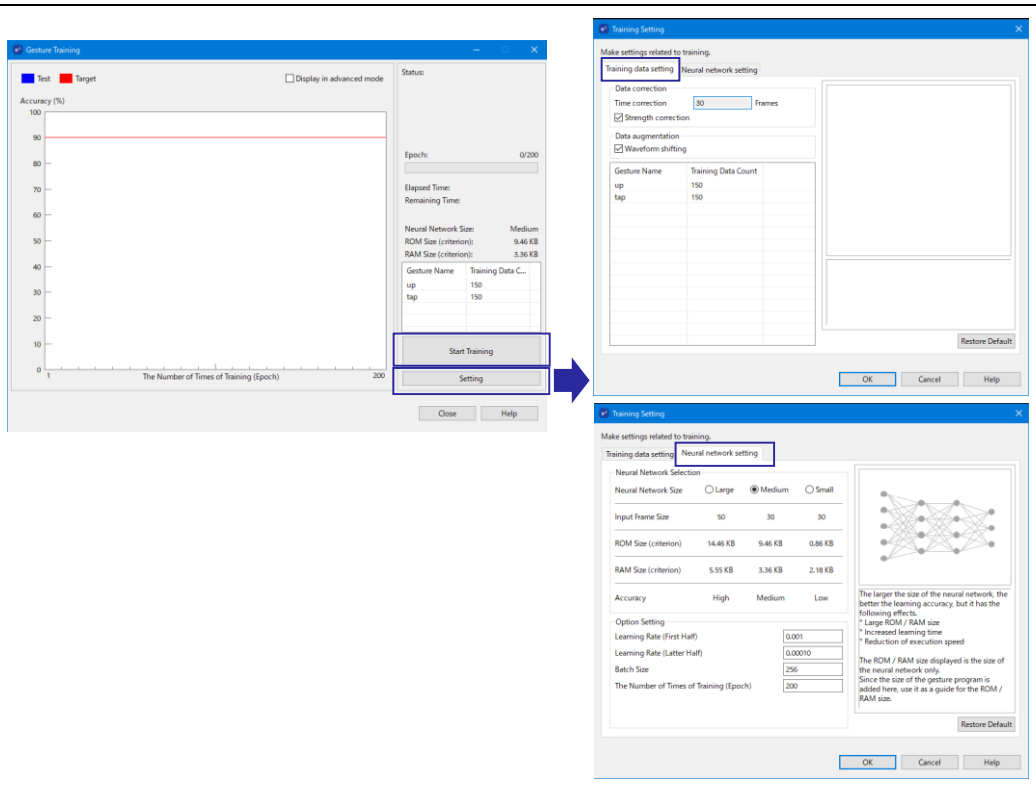


**Figure 4-32 AI Training**

When AI training is finished, the learning accuracy will be displayed in a graph. If the accuracy does not exceed 90%, review the registered gesture data and execute AI training again.

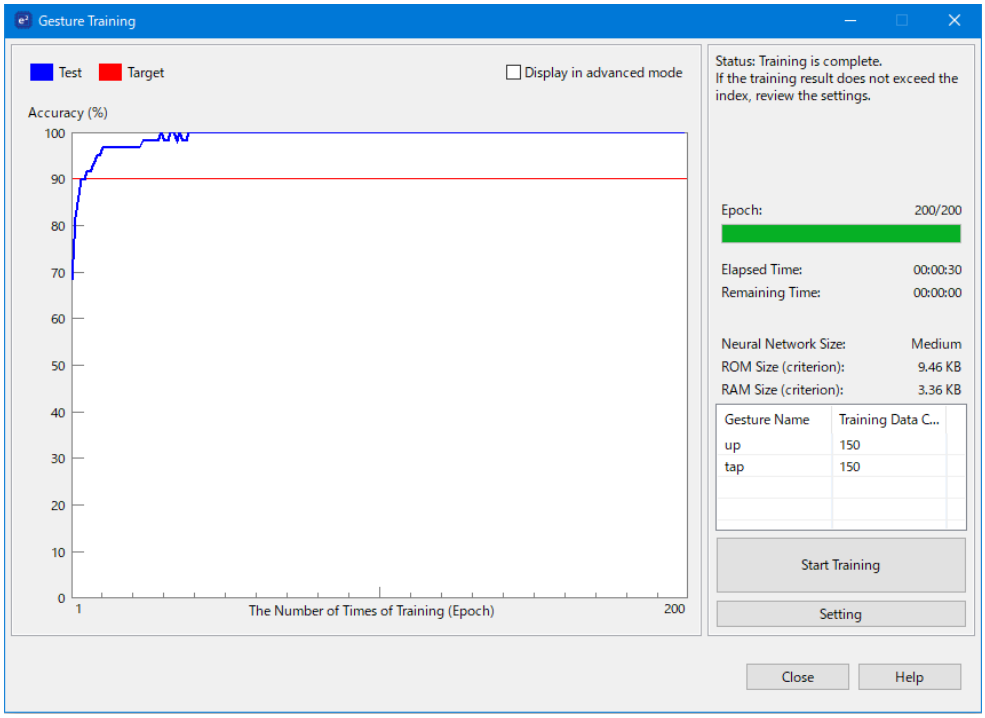


**Figure 4-33 AI Learning Result**

### 4.3.5 Output Learning Result

Click “Output C Source Files” as shown below to output the AI learning result to the C source file. The files output in this chapter are stored in the qe_gen folder and are used in the coding process.

For the C source files output by QE for Capacitive Touch, refer to Chapter 5.

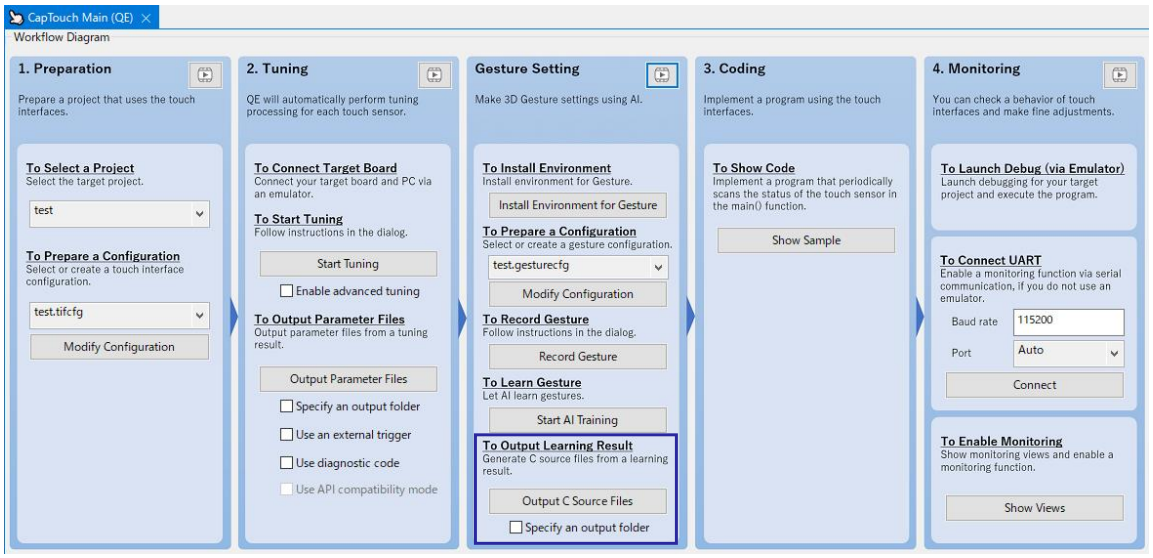


**Figure 4-34 Create C Source Files of AI Learning Results**

When using CC-RX compiler or CC-RL compiler made by Renesas, the changes of settings may be required. An Example of CC-RX compiler settings is shown in Figure 4-35, and an example of CC-RL compiler settings is shown in Figure 4-36.

- Addition of standard library
  Build error may occur because the default setting of standard library “math.h” is disable. In such case, change the standard library “math.h” setting to enable.
- Change setting to “C99”
  Build error may occur because the default setting of C standard is “C89” or “C90”. In such case, change the setting to “C99”.

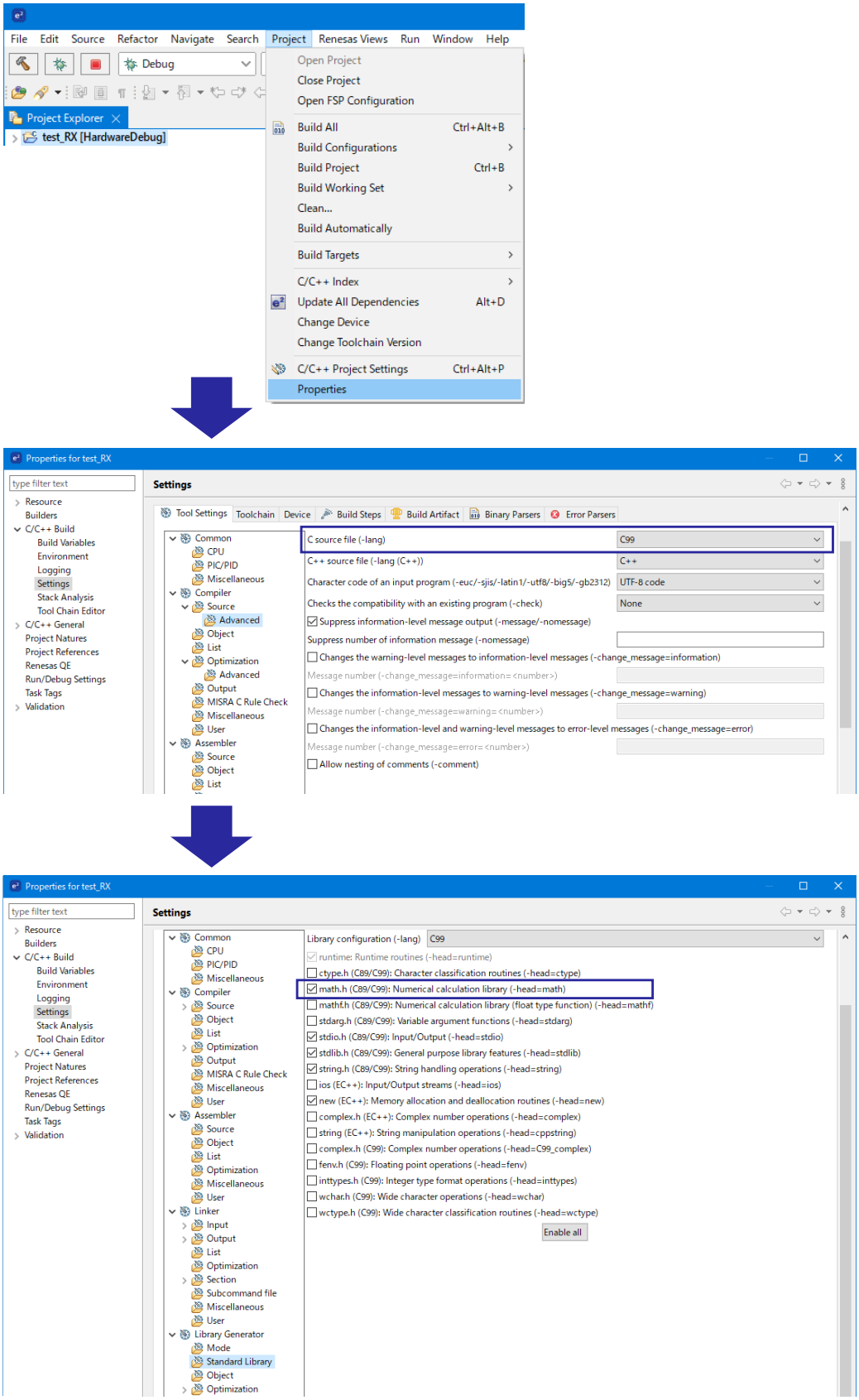


**Figure 4-35 Example of CC-RX compiler settings.**

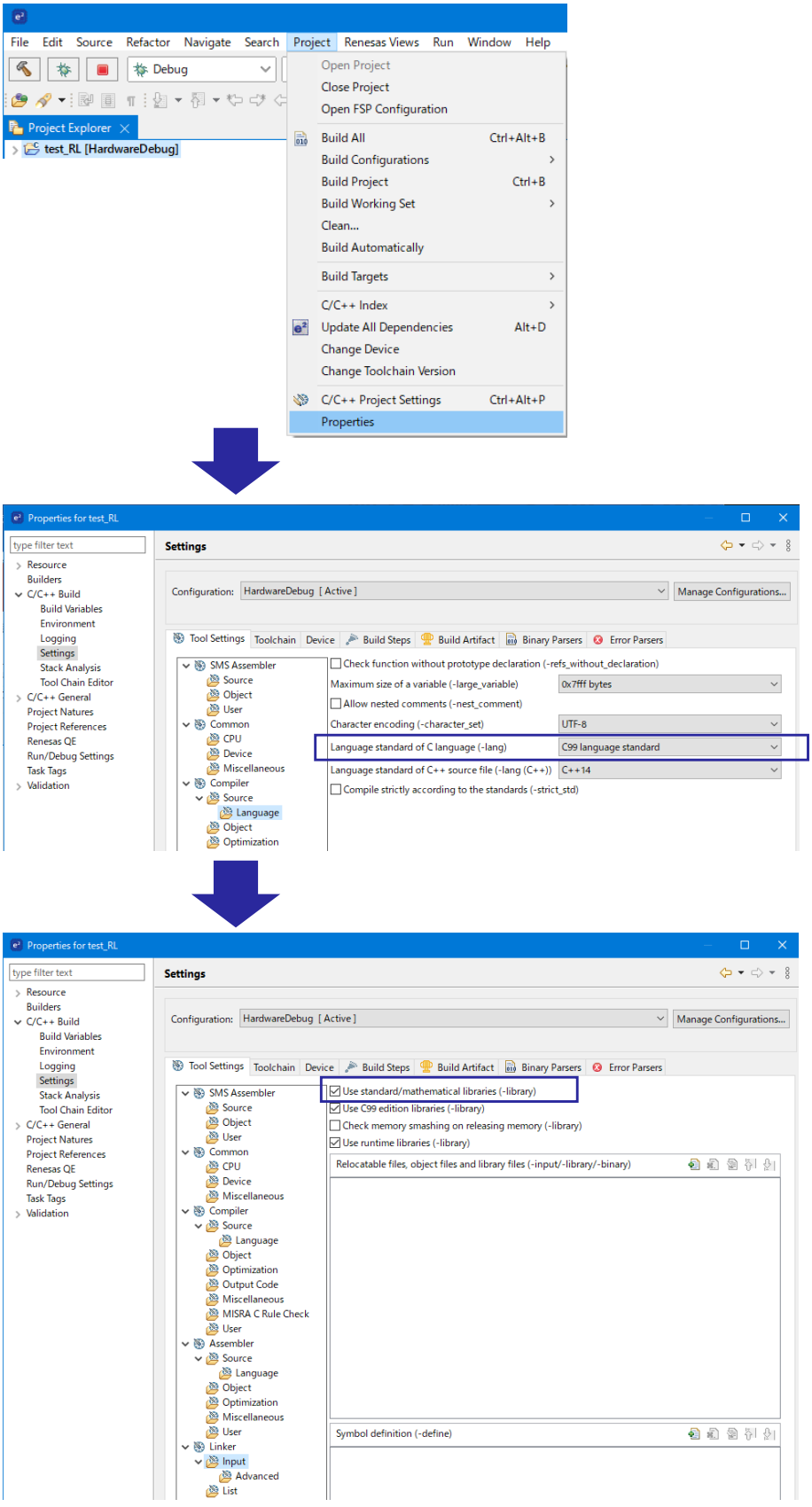


**Figure 4-36 Example of CC-RL compiler settings**

## 4.4 Coding

Implement the C source code output by QE for Capacitive Touch into your project.

### 4.4.1 Showing the Code Example

Click "Show Sample" and "Output to a File" as shown below to output the main function of gesture recognition to the C source file. The files output in this chapter are stored in the qe_gen folder.

For the C source files output by QE for Capacitive Touch, refer to Chapter 5.

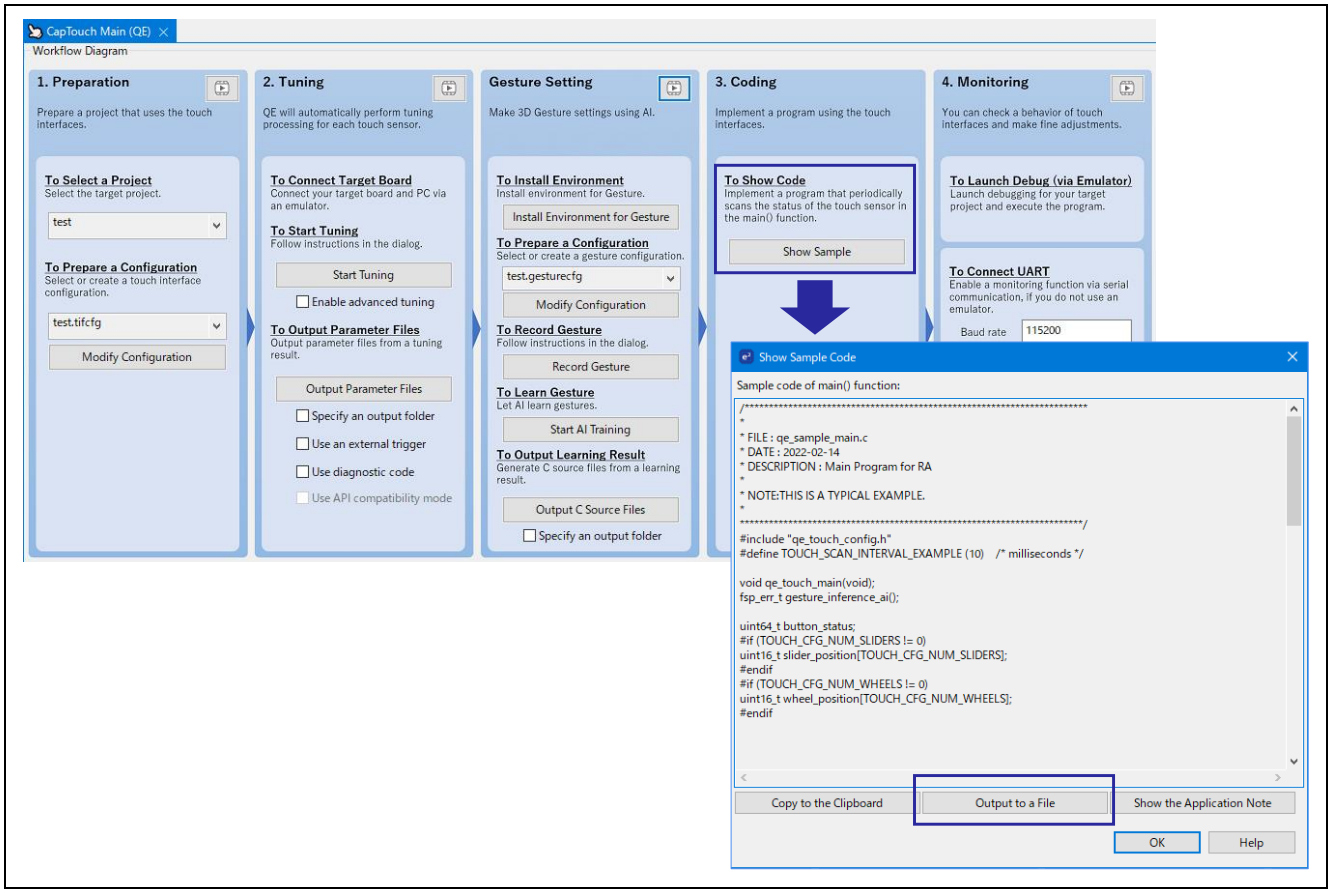


**Figure 4-37 Output to a File of the Code Example**

#### 4.4.1.1 Calling Gesture Main Function

Follow the procedure below to implement the call to the gesture recognition main function generated by QE for Capacitive Touch in the C source file.

- Open the hal_entry.c file in your project's src folder.
- Add the call to the qe_touch_main () function to the hal_entry () function in the hal_entry.c file.
- Build the project.


**Figure 4-38 Calling the QE Generation Function**

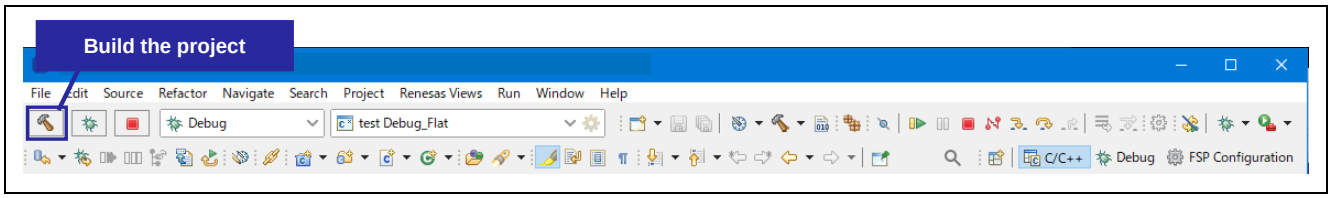


**Figure 4-39 Build the Project**

## 4.5 Monitoring

### 4.5.1 Launching Debugging

Start debugging the program implemented by the procedure below.

- Start debugging – (1)
- Start the program – (2)

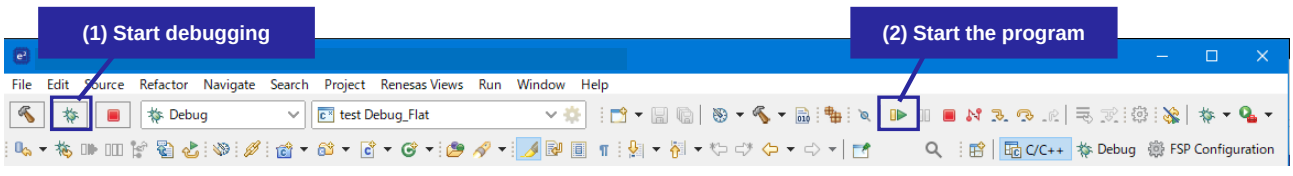


**Figure 4-40 Starting Debugging and Program**

### 4.5.2 Enable Monitoring

Follow the steps below to display the view for monitoring.

- Show the CapTouch Main window – (1)
- Click “Show Views” to display the CapTouch Status Chart window.

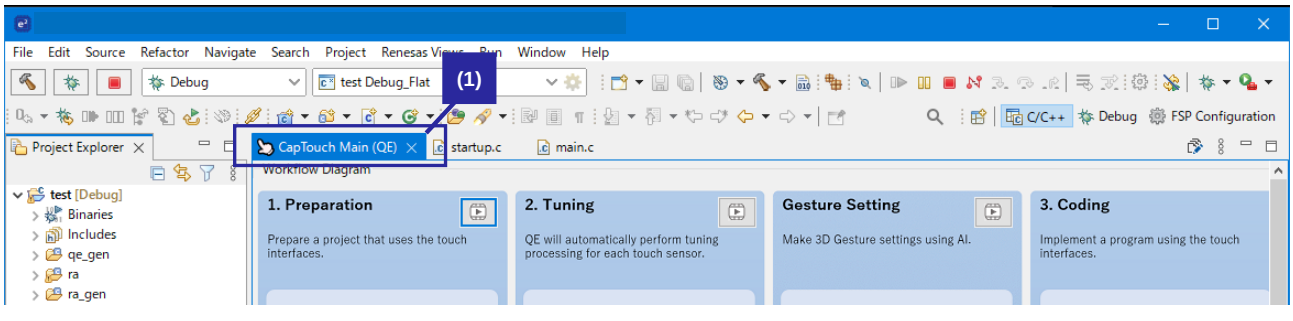


**Figure 4-41 Showing CapTouch Main**

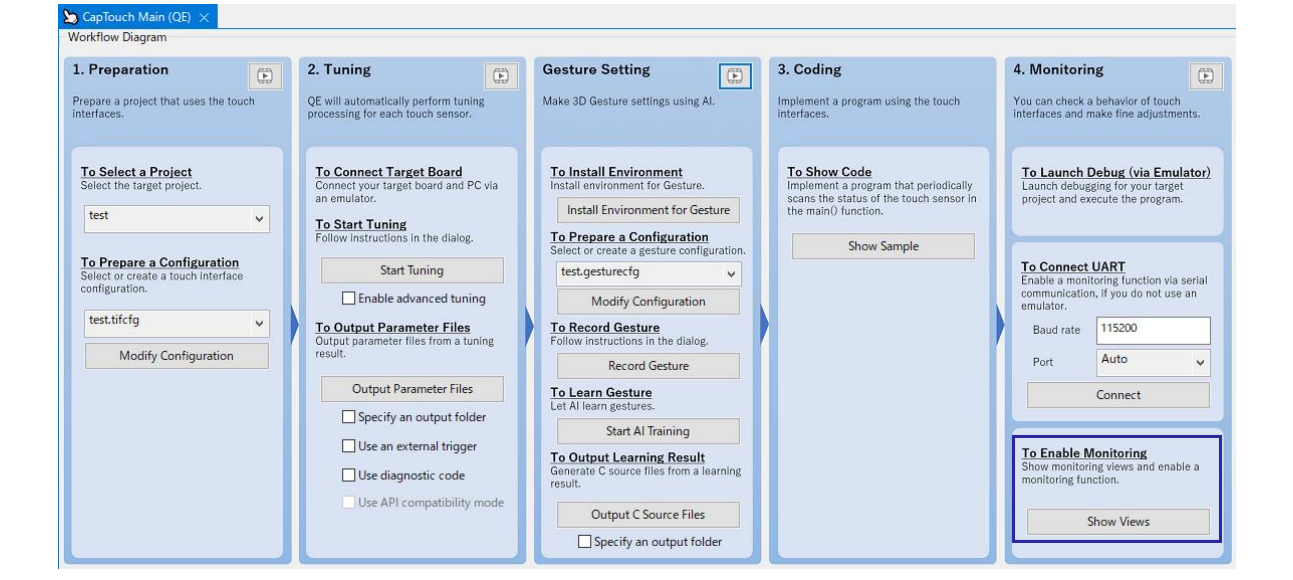


**Figure 4-42 Enable Monitoring**

### 4.5.2.1 Recognition Gesture

Follow the steps below to monitor the gesture recognition status.

- Select the target interface from “Touch I/F” to display the CapTouch Gesture Monitor window. – (1)
- Enable monitoring. – (2)
- Make a gesture on the sensor.
- Recognized gestures are displayed.

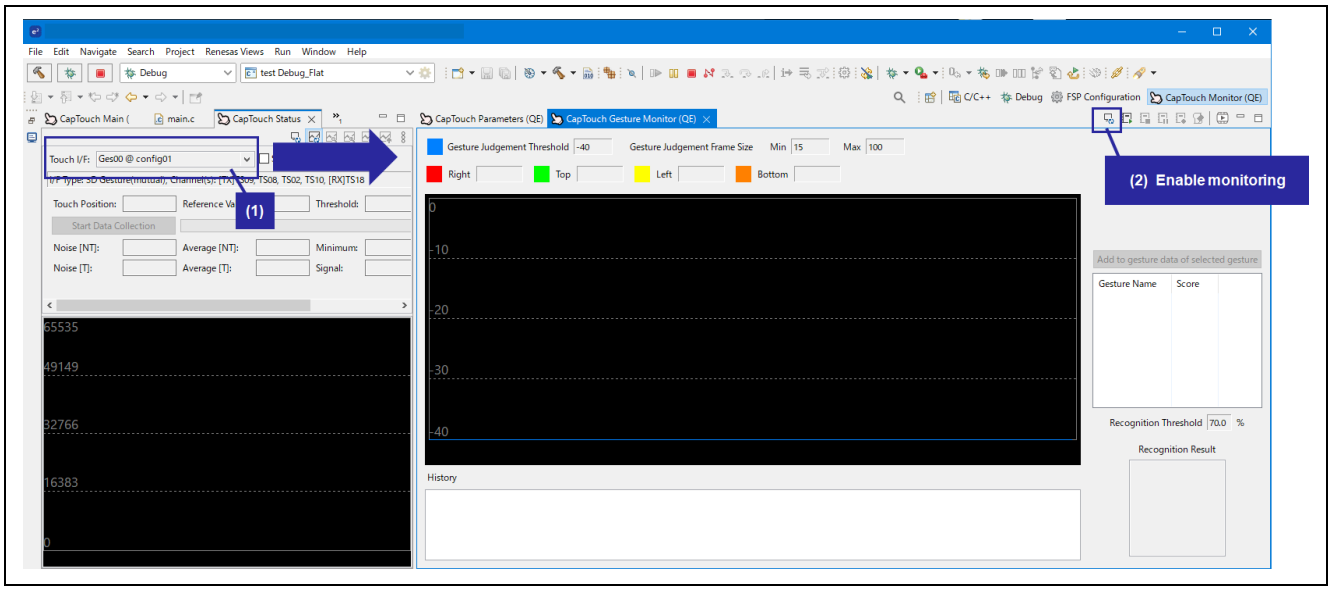


**Figure 4-43 Enable monitoring**

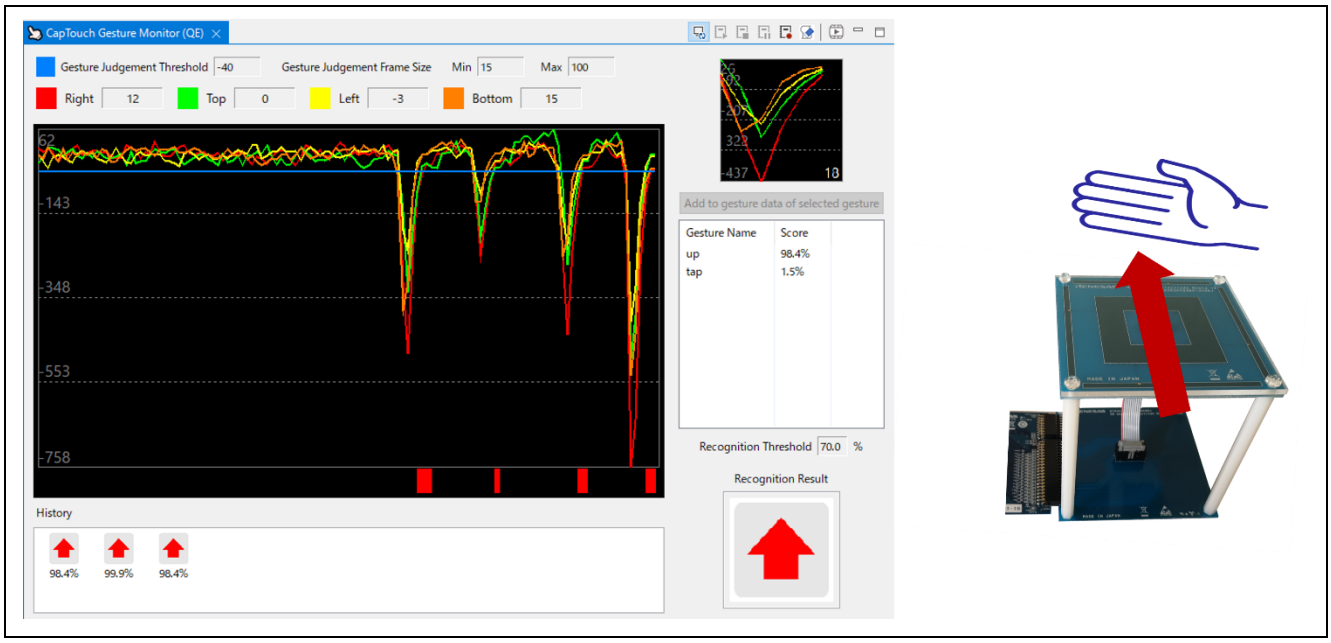


**Figure 4-44 Recognition Gestures**

### 4.5.2.2 Misrecognition Gesture Registration

If it is not recognized as expected, you can additionally register the gesture data that was misrecognized. The additionally registered gesture data will be reflected in the gesture data list in Chapter 4.3.3 To monitor again, start over from Chapter 4.3.4.

Gestures that have been misrecognized can be additionally registered by following the steps below.

- Disable monitoring – (1)
- Select a misrecognized gesture from the history – (2)
- Select the gesture name you want to register – (3)
- Click “Add to gesture data of selected gesture” – (4)
- Stop debugging and start over from Chapter 4.3.4. – (5)

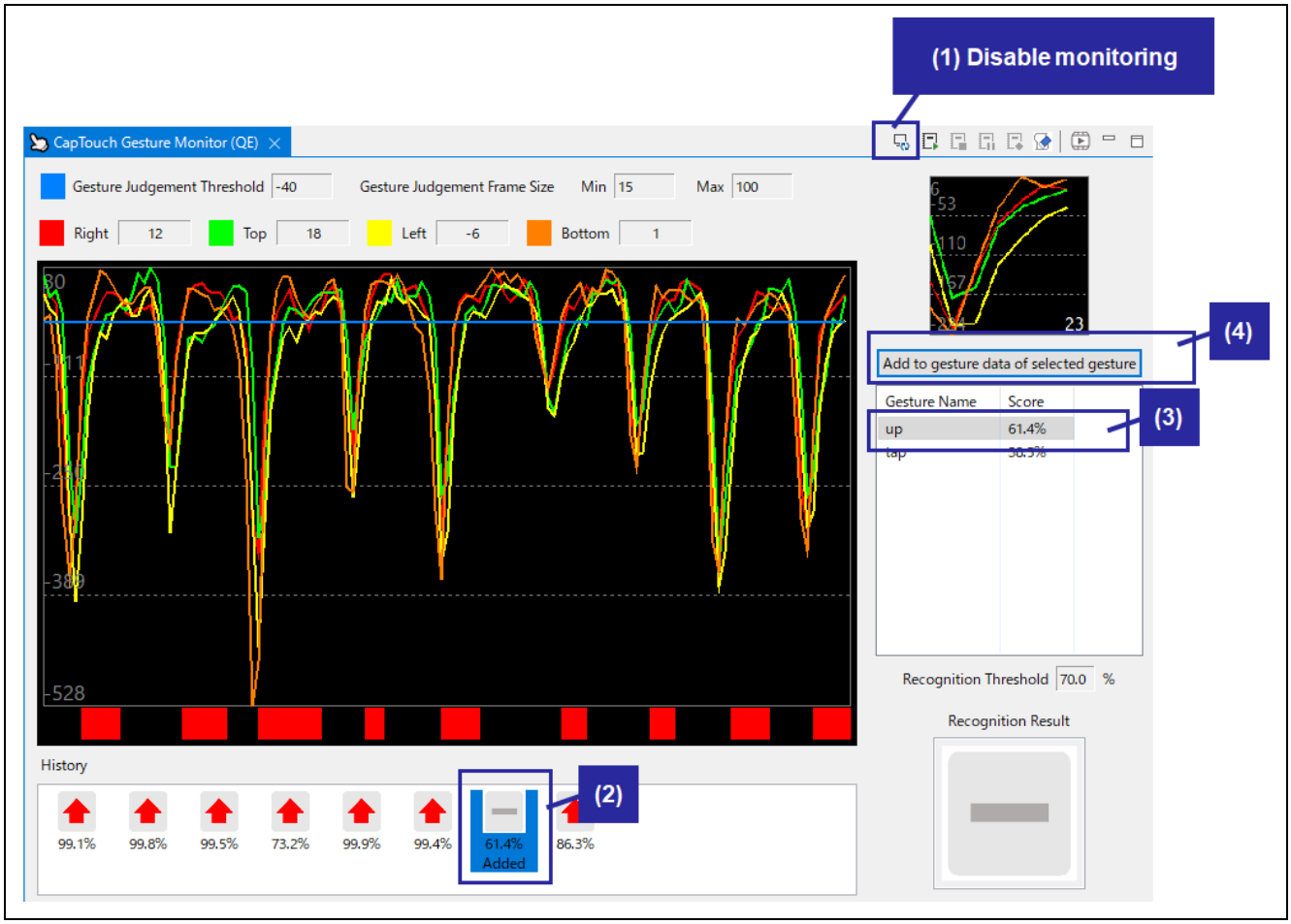


**Figure 4-45 Registration Misrecognized Gestures**

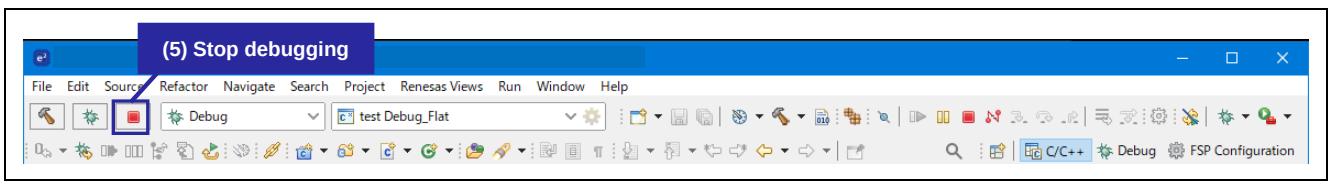


**Figure 4-46 Stop debugging**

### 4.5.2.3 Changing the Parameter

You can change the following parameters in the CapTouch Parameters window while monitoring the gesture recognition status in the CapTouch Gesture Monitor window. The CapTouch Parameters window can be displayed by selecting “CapTouch Parameters (QE)” from the e² studio menu.

- Gesture Judgement Threshold: The threshold of the sensor that initiates the gesture judgment – (1)
- Gesture Judgment Frame Size: Frame size for gesture judgment (sampling number) – (2)
- Recognition Threshold: the threshold at which the gesture is judged to have been performed – (3)

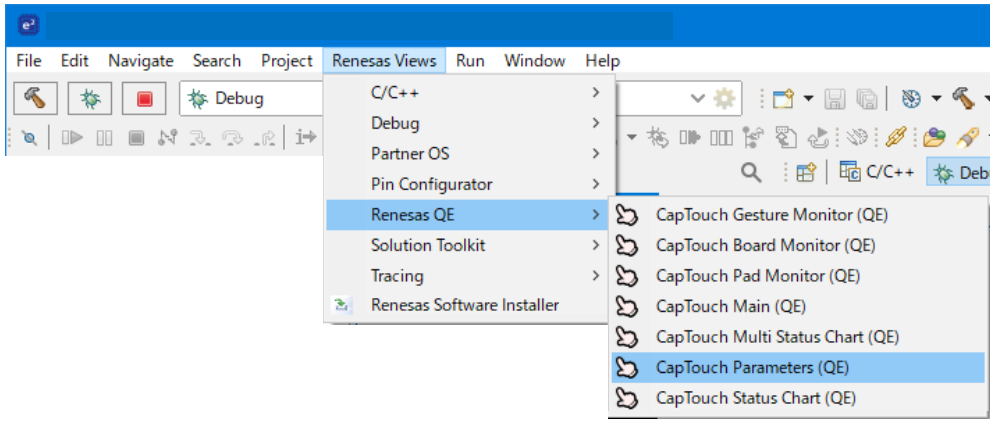


**Figure 4-47 Selection for CapTouch Parameter List**

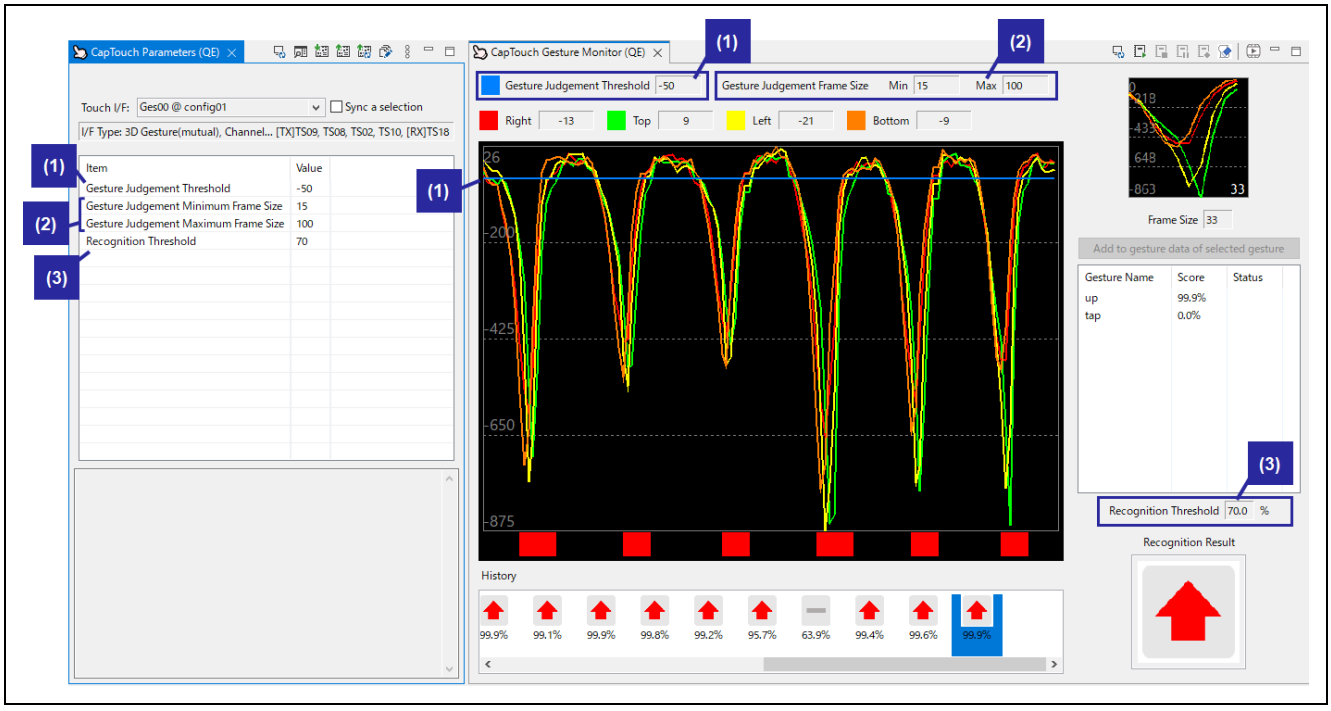


**Figure 4-48 Variable Parameters**

You can change the gesture judgement threshold by following the steps below. Other parameters can be changed in the same way.

- Select the target interface from the ”Touch I/F” – (1)
- Change the value of the “Gesture Judgement Threshold” – (2)
- Write to target board – (3)
  It is reflected in the “Gesture Judgment Threshold” of the CapTouch Gesture Monitor window.
  It is not reflected in the C source file, but you can monitor the behavior after changing the parameters.
- Once the parameters are finalized, generate a parameter file – (4)
  It will be reflected in the C source file in the qe_gen folder.
- Disable monitoring if you want to check the operation in the program after changing the parameters – (5)
  You can disable monitoring in either the CapTouch Gesture Monitor window or the CapTouch Parameters window.
- Stop debugging, build the project, and start over from Chapter 4.5.1. – (6)

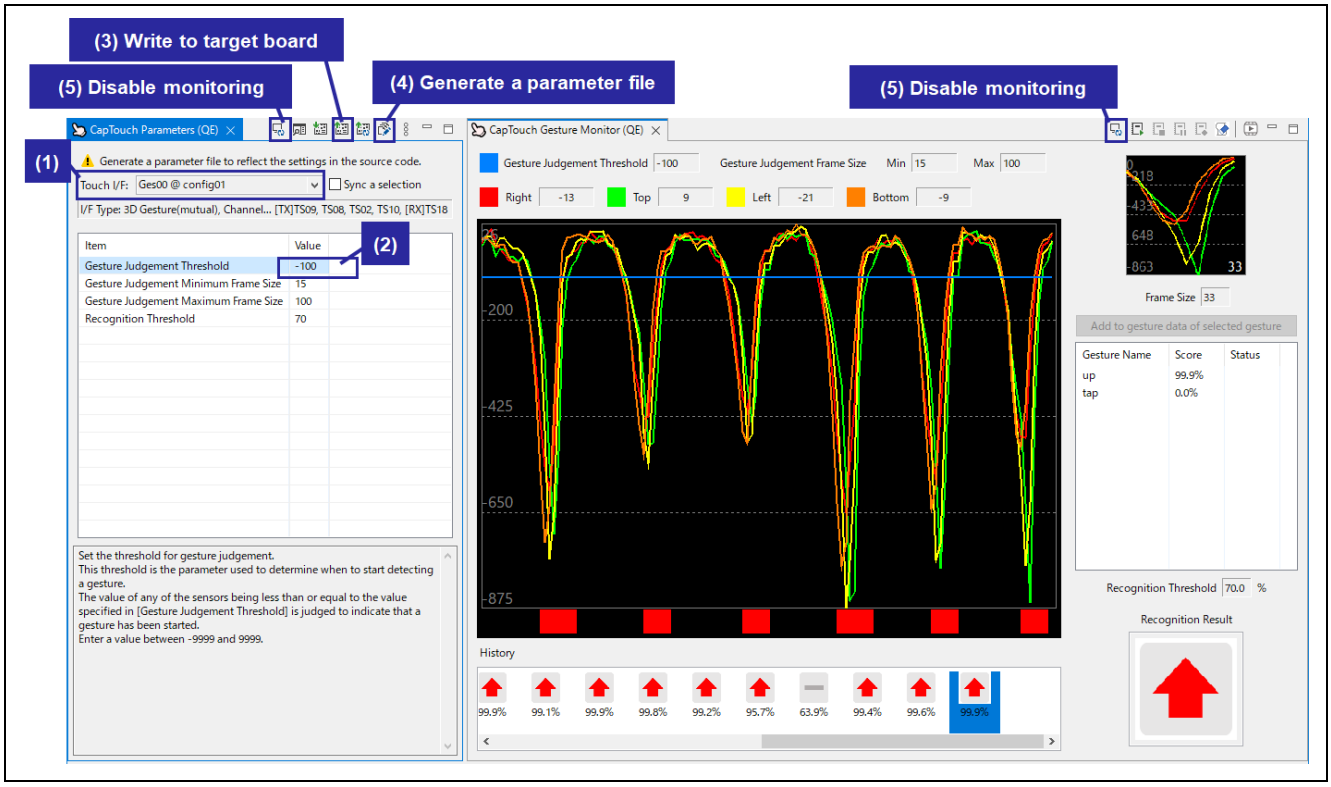


**Figure 4-49 Process for Change the Parameter**

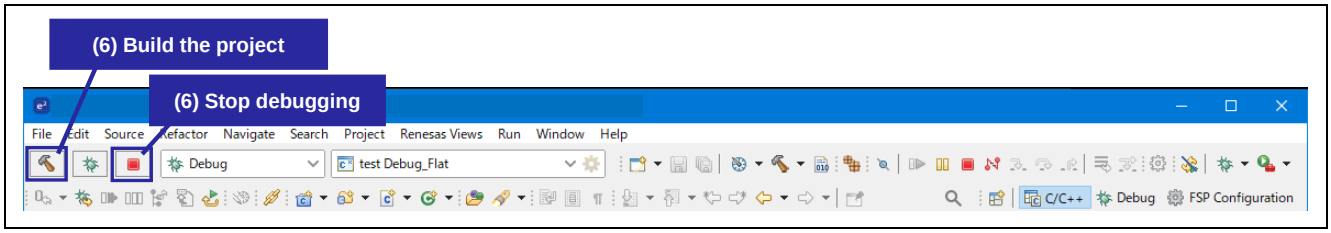


**Figure 4-50 Stop the Debugging and Build the Project**

### 4.5.2.4 Log Recording and Playback

In the CapTouch Gesture Monitor window, you can record and play the monitoring log by following the steps below.

- Enable monitoring – (1)
- Start recording monitoring log – (2)
- Make a gesture on the sensor.
- Stop logging – (3)
  Save the log file with the extension ".qegesmon".
- Disable monitoring – (4)
- Play monitoring log – (5)


**Figure 4-51 Recording Monitoring Log**

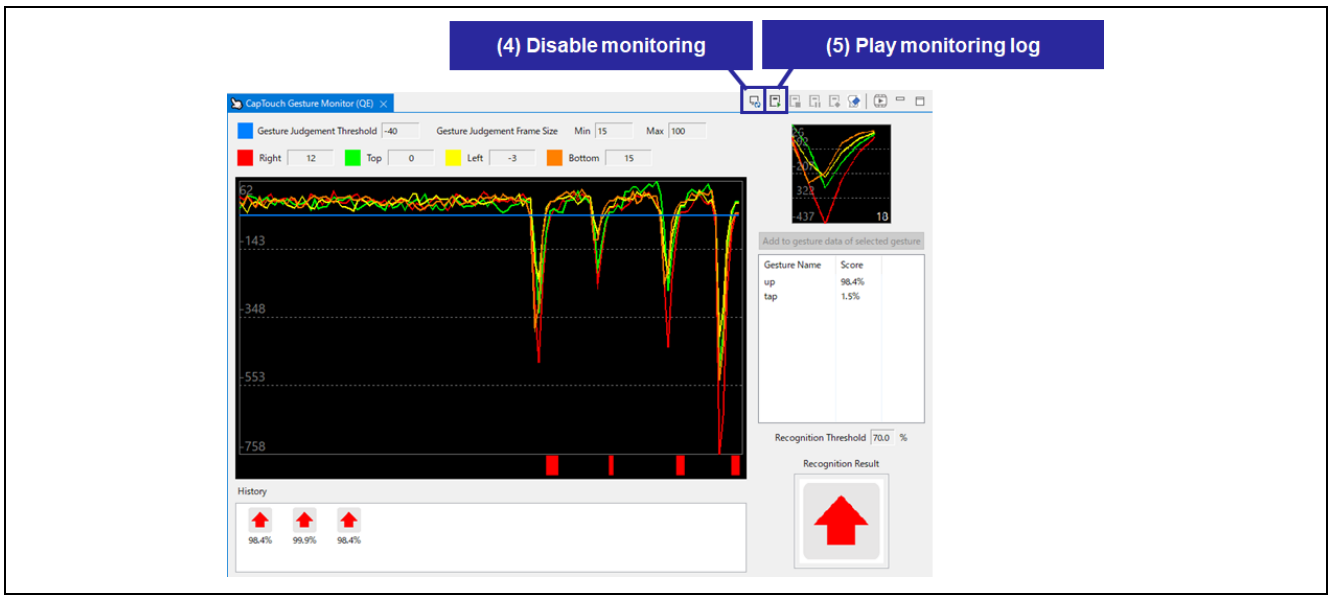


**Figure 4-52 Play Monitoring Log**

## 5. Output files of the QE for Capacitive Touch

The files output by QE for Capacitive Touch are as follows.

**Table 5-1 List of Output Files from the QE**

| Folders | File Name | Items Generated |
|---|---|---|
| qe_gen | qe_touch_sample.c | Display implementation example |
| | qe_gesture.c | Output of gesture adjustment result |
| | qe_gesture.h | |
| | qe_gesture_user.c | |
| | qe_touch_config.c | Output of touch electrode sensitivity adjustment result |
| | qe_touch_config.h | |
| | qe_touch_define.h | |
| qe_gen¥Translator | checker_log_output.txt | Output of learning result |
| | dnn_compute.c | |
| | input_image_0.h | |
| | layer_graph.h | |
| | layer_shapes.h | |
| | network.c | |
| | network_description.txt | |
| | Typedef.h | |
| | weights.h | |

## 5.1 User Program Implementation

This chapter describes how to add processing when each gesture is detected.

Follow the steps below to add the user program to the qe_gesture_user.c file generated by QE for Capacitive Touch.

- Open the “qe_gesture_user.c” file in the project's qe_gen folder.
- Add the user program to the function in the “qe_gesture_user.c” file.

The gesture judgment frame size is set in the argument “frame_size” of the “gesture_user ()” function. By using “frame_size”, the user can divide the processing even for the same gesture according to the difference in the speed of moving the hand.

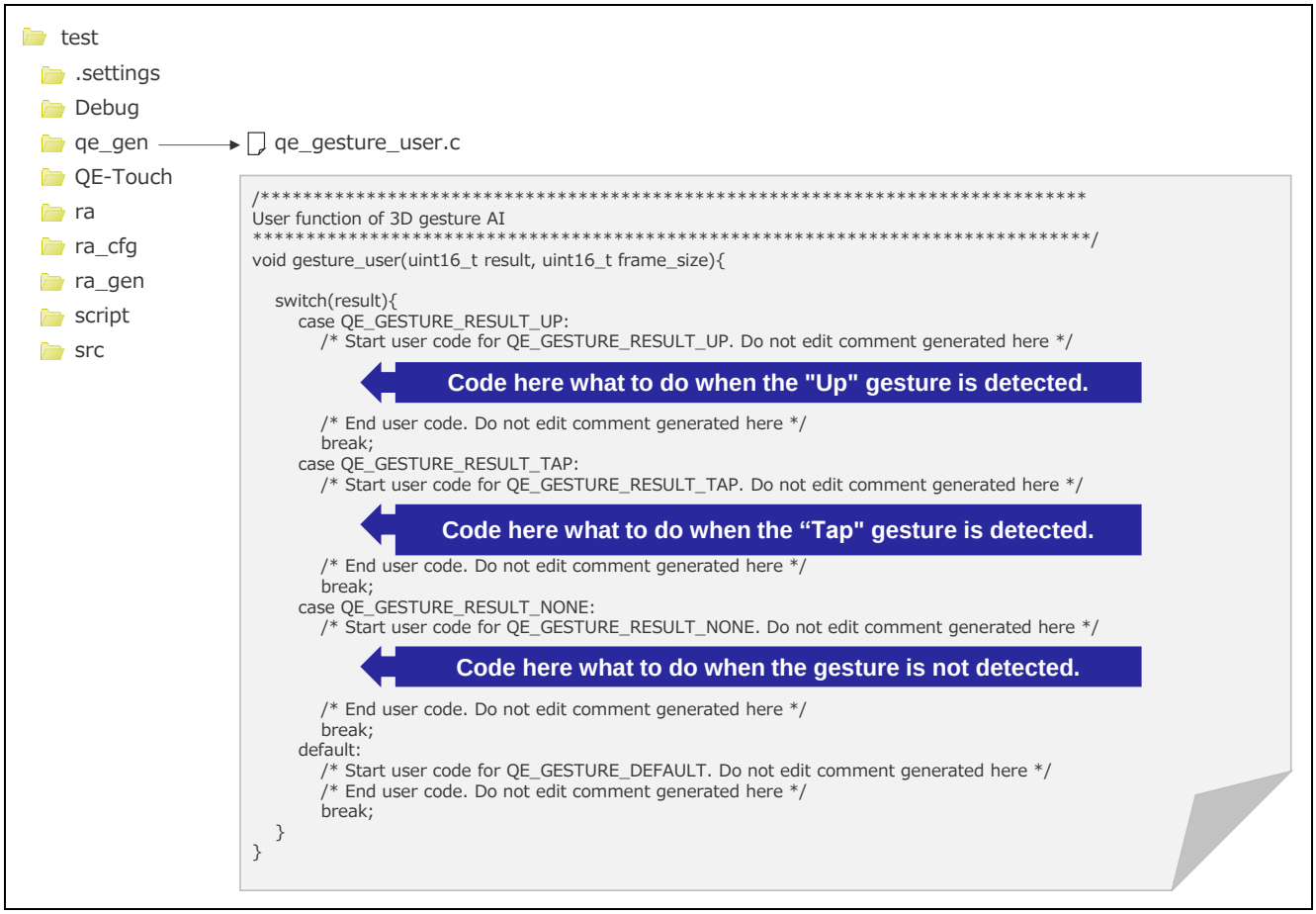


**Figure 5-1 User Program Implementation**

## 6. Help Function

You can check the details of the functions of “QE for Capacitive Touch” from the help of e² studio.

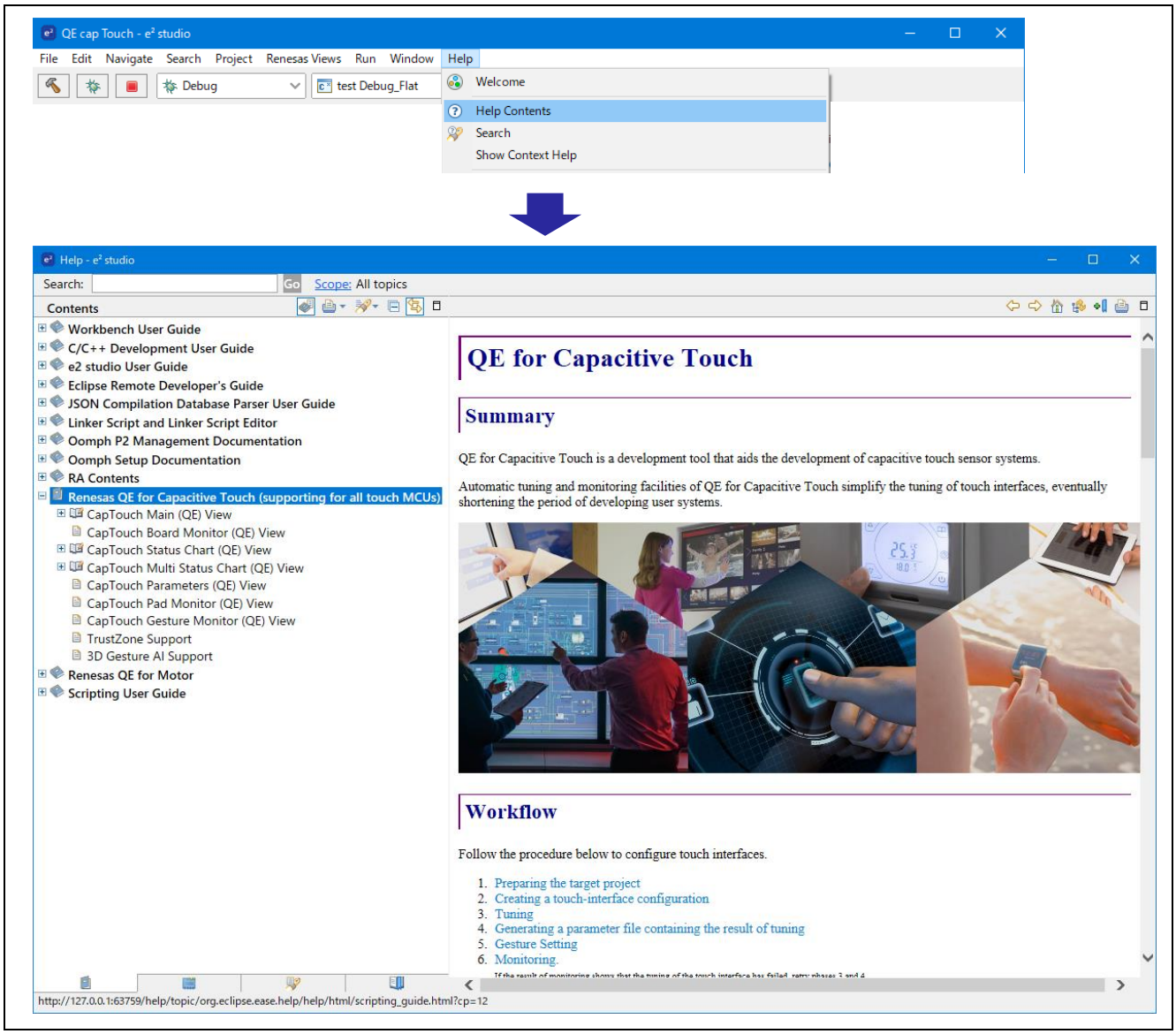


**Figure 6-1 Help Function**

## Revision History

| Rev. | Date | Description | |
|---|---|---|---|
| | | Page | Summary |
| 1.00 | Apr.22.22 | — | First edition |
| 1.10 | Oct.7.22 | 16-17, 24-26, 35-39 | Change figure. |

# General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between $V_{IL}$ (Max.) and $V_{IH}$ (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between $V_{IL}$ (Max.) and $V_{IH}$ (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

# Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

## Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

## Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

## Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.