

Capacitive Sensor Microcontrollers

CTSU Capacitive Touch Introduction Guide

Introduction

This application note is an introduction guide for customers who use the Capacitive Touch Sensor Unit (Capacitive Touch Sensing Unit: hereinafter referred to as CTSU) for the first time.

Target Device

RX Family, RA Family, and RL78 Family MCUs and Renesas Synergy™ embedding the CTSU (CTSU indicates CTSU2, CTSU2L, CTSU2SL, etc.)

Contents

1. Outline	3
1.1 Application Product Development Process for Capacitive Touch	4
2. Capacitance Detection.....	5
2.1 Capacitance Detection Overview.....	5
2.2 Self-capacitance Method	6
2.2.1 Detection Principle	7
2.2.2 Measurement Range.....	8
2.3 Mutual-capacitance Method	9
2.3.1 Detection Principle	10
2.3.2 Measurement Range.....	11
2.4 Drift Correction Processing and Touch Detection.....	12
3. Capacitive Touch Sensor Type.....	13
3.1 Differences in Capacitive Touch Sensor	13
3.2 CTSU1.....	14
3.2.1 Random Pulse Frequency Measurement.....	14
3.3 CTSU2.....	15
3.3.1 Multi-clock Measurement	15
3.3.2 Active Shield	17
3.3.3 Automatic Judgment	18
3.3.4 Automatic Correction.....	18
3.3.5 Multi-Clock Automatic Correction	18
3.3.6 Multiple Electrode Connection (MEC).....	19
3.3.7 Mutual capacitance parallel measurement.....	19
4. Capacitive Sensor MCU	20
4.1 MCU Lineup.....	20

4.2	Points for MCU Selection	21
4.2.1	Interface.....	21
4.2.2	Features of Detection Method	22
4.3	MCUs Embedding CTSU1.....	24
4.4	MCUs Embedding CTSU2.....	28
5.	Hardware.....	32
5.1	Electrode design guide.....	32
6.	Software	33
6.1	Software Configuration.....	33
6.1.1	CTSU Module	34
6.1.2	TOUCH Module	34
6.1.3	User Application.....	34
6.2	Functional Safety	35
6.3	Low Power Guide.....	35
6.4	Software Filter.....	35
7.	Tuning	36
7.1	Auto Tuning with QE for Capacitive Touch	37
7.2	Manually Tuning with CapTouch Parameters.....	38
7.3	Tuning with advanced mode.....	41
8.	Development Environment.....	42
8.1	Capacitive Touch Sensor Development Support Tool “QE for Capacitive Touch”	42
8.2	Stand-alone Version of QE for Capacitive Touch.....	43
8.3	Measurement of Standard Deviation and SNR	44
9.	Evaluation kit.....	45
9.1	Capacitive Touch Evaluation System.....	45
10.	Other	46
10.1	Term.....	46
10.2	Q&A's	46
10.3	Technical inquiries	47
	Revision History.....	48

1. Outline

This application note is an introduction guide for customers who use the capacitive touch sensor for the first time.

This chapter introduces the principles and features of capacitive touch detection, MCU lineup, hardware, software, development environment, and evaluation kits (Capacitive Touch Evaluation System).

The first-generation capacitive touch sensor uses a third-party IP(SCU/TSCU) and is used in R8C family (newly adopted is not recommended). From the second-generation, products used Renesas' uniquely developed capacitive touch sensor CTSU were released from the RX family and are also being deployed in the Renesas Synergy™ and RA families. Today, the functions of CTSU2 are also evolving, as products embedding the third-generation CTSU2 are used in a variety of products, RA family, RL78 family, and RX family.

In the user's manuals of products. etc., the capacitance sensor of the second generation is written as CTSU/CTSUA/CTSUB, but it is written as "CTSU1" to distinguish it from CTSU2 of the third generation. This application note discusses the differences between CTSU1 and CTSU2, as well as the characteristics and types of CTSU2.

1.1 Application Product Development Process for Capacitive Touch

Figure 1.1 shows the application product development process for capacitive touch. Refer to the following for details.

- For understanding of capacitive touch, refer to “2.Capacitance Detection” and “3.Capacitive Touch Sensor Type”
- For MCU with Capacitive touch sensor, refer to “4.Capacitive Sensor MCU”
- For H/W, refer to “5.Hardware”
- For S/W, refer to “6.Software”
- For tuning, refer to “7.Tuning”
- For development environment, refer to “8.Development Environment”
- For evaluation kit, refer to “9.Evaluation kit”

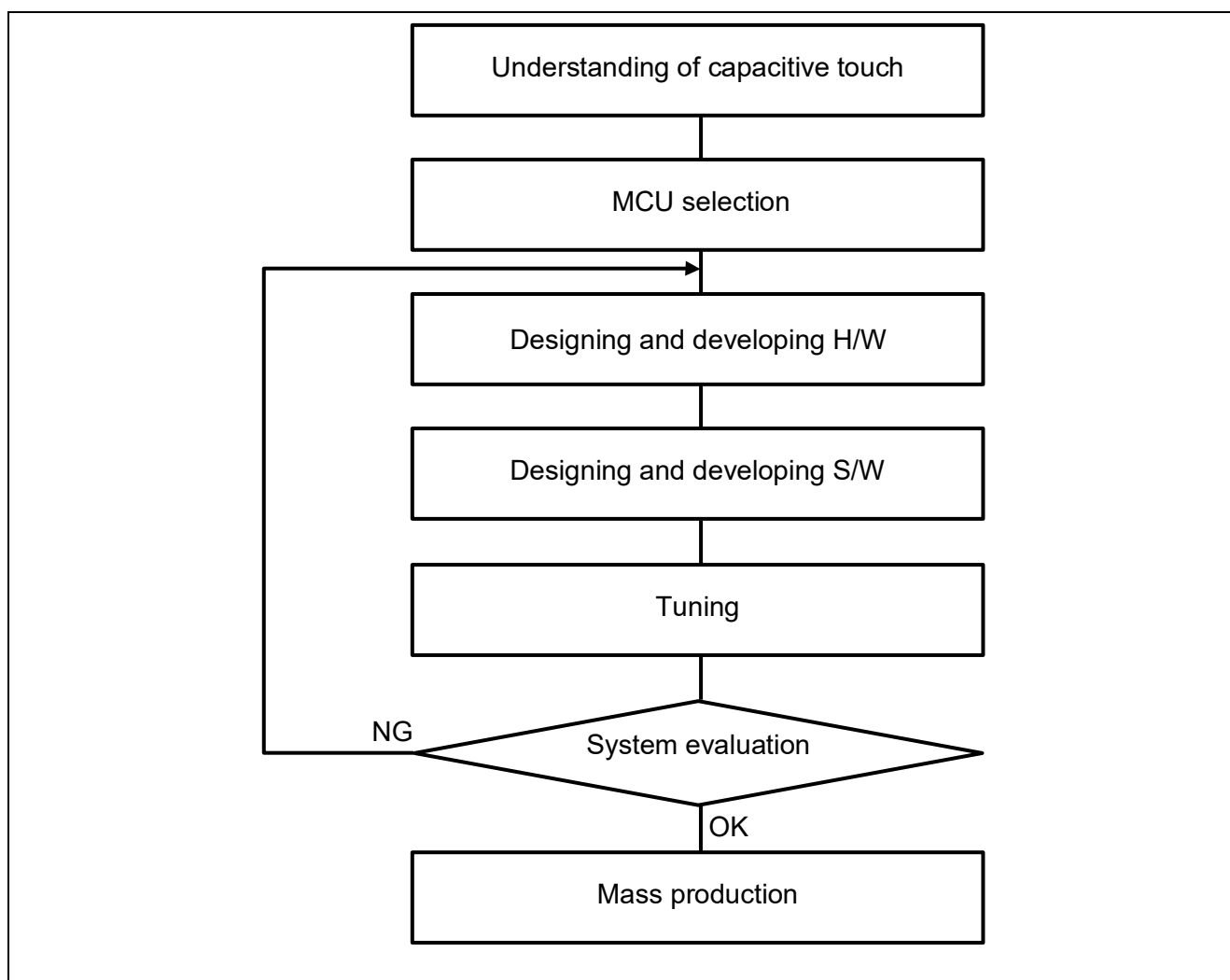


Figure 1.1 Application product development process for capacitive touch

2. Capacitance Detection

2.1 Capacitance Detection Overview

Unlike a typical switch with mechanical contacts, the Capacitance Touch Key captures minute changes in capacitance (a few pF or less) generated between the human body and the electrodes. The change is converted to ON/OFF similar to that of a mechanical switch.

For this reason, wiring and electrodes for capacitance detection must be designed correctly because they are affected by disturbances such as noise.

CTSU developed by Renesas converts capacitance into current and quantifies capacitance measurement using a switched capacitor circuit. The measured value is corrected, and the noise component is reduced by filtering. Finally, the judgment is made by comparing with the threshold value. Figure 2.1 shows the flow from capacitance measurement to judgment. In addition, CTSU supports the "self-capacitance method" and "mutual-capacitance method" detection methods.

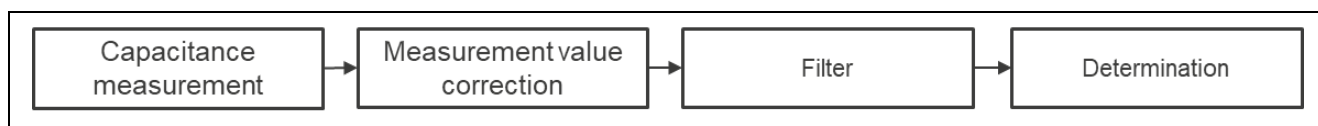


Figure 2.1 Flow from Capacitance Measurement to Touch Judgment

2.2 Self-capacitance Method

The "self-capacitance method" is a method that detects changes in capacitance that occur between the electrode and GND.

Figure 2.2 shows the touch detection principle of self-capacitance method. The left side of the figure shows the capacitance of only the electrode. The right side shows the capacitance where the capacitance has changed (increased) due to human body contact. Capacitance also exists in electrodes and wiring where capacitors are not connected in the circuit. This is called parasitic capacitance. When a finger approaches, capacitance is generated between the electrode and the finger, and the capacitance between the electrode-GND is increased.

The total capacitance between the electrode and GND can be expressed by the following formula.

$$\text{Total Capacity} = C_p + C_f$$

C_p : Parasitic capacitance

C_f : Capacitance to fingers

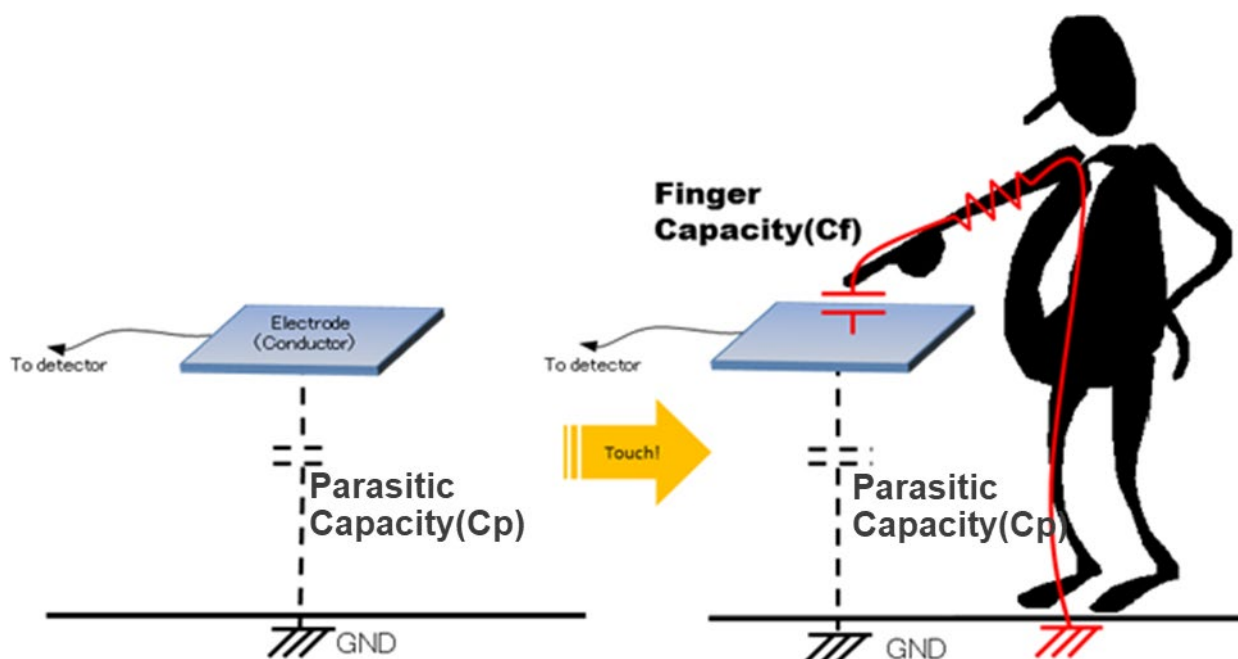


Figure 2.2 Touch detection principle of self-capacitance method

2.2.1 Detection Principle

Figure 2.3 shows the measurement circuit of CTSU.

CTSUS measures the capacitance by outputting a sensor drive pulse from TS terminal and measuring the charge/discharge current. The following equation is established when the electrode-side current: I , sensor drive pulse frequency F , and parasitic capacitance are C_p , finger capacitance C_f , and sensor drive pulse voltage V .

$$I = F (C_p + C_f) V$$

The current I is the sum of the current I_1 supplied from VDC (Voltage Down Converter) for measurement and the current I_2 supplied from the offset current (Offset DAC).

For the current I_1 supplied from the measurement VDC, a current I_{OUT} proportional to the current-controlled oscillator (Current Controlled Oscillator: hereinafter CCO) is applied via the current mirror. CCO outputs a pulse-frequency proportional to I_{OUT} . The current I_{OUT} is measured for a fixed period according to the sensor drive pulse output. It is stored in the sensor counter register. CCO has oscillation variation. The result obtained by correcting the value of the sensor counter register (CCO correction) is used to reduce this variation. For details 3.3.4 Automatic Correction.

The amount of current (measured value) differs when only the parasitic capacitance C_p is used and when a C_f is applied due to finger-contact. The change of capacitance is detected from the change of this measured value.

The TSCAP capacitor is used to stabilize the internal voltage of CTSUS.

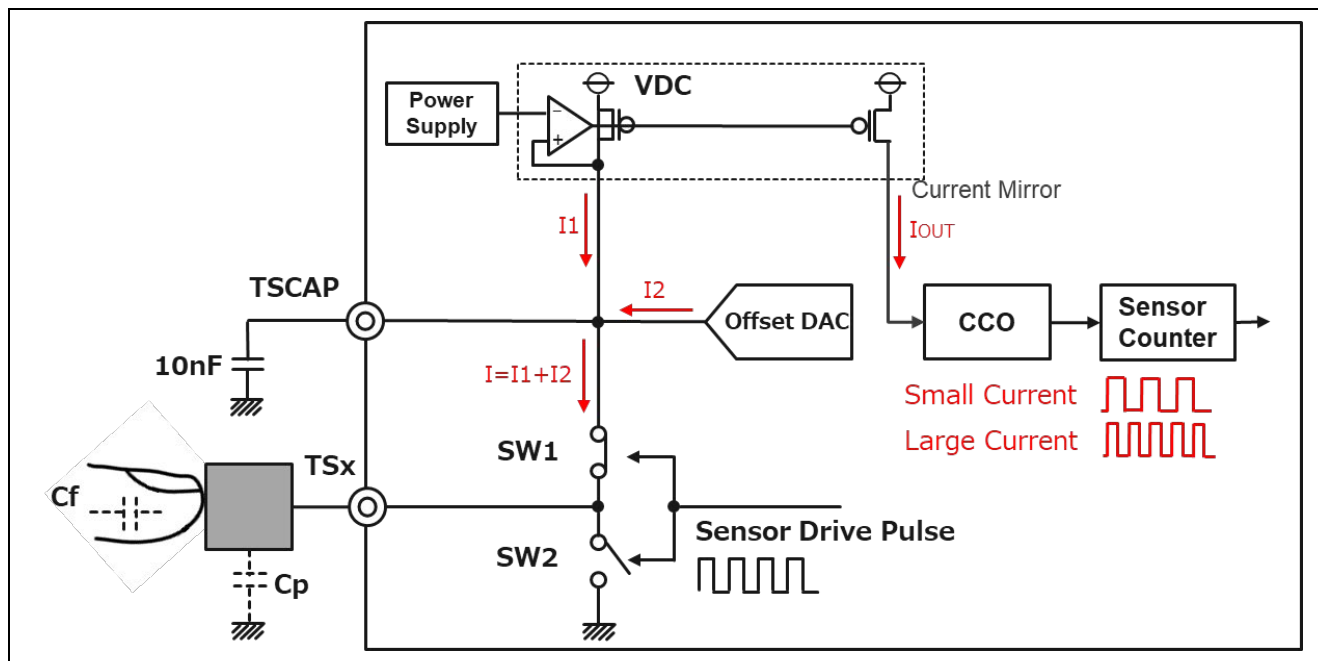


Figure 2.3 Measurement circuit of CTSU

2.2.2 Measurement Range

The sensor counter register must be used within the valid measurement range (upper limit $\leq 100\%$), even though the 16-bit register itself spans 0–65535. The CTSU includes a sensor offset adjustment register that tunes the offset current to compensate for parasitic capacitance components. Adjusting this offset current to bring the measured count closer to the target value within the valid range is referred to as the offset tuning process. This is done by software-processing after resetting MCU to adjust individual differences in hardware.

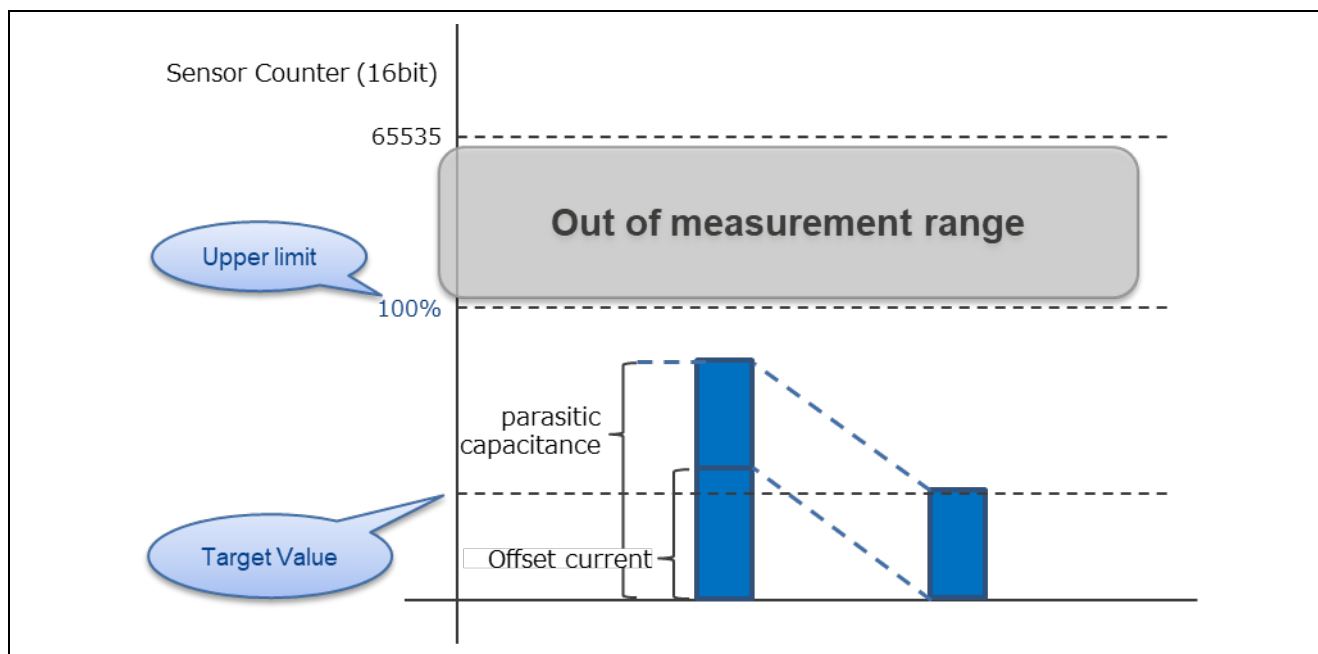


Figure 2.4 Offset tuning process of self-capacitance method

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_ ctsu\)](#)

2.3 Mutual-capacitance Method

The "mutual-capacitance method" is a method that detects changes in capacitance that occur between electrodes.

In the mutual-capacitance method, the electrostatic capacitance (hereinafter referred to as "inter-electrode capacitance") generated between two electrodes, the transmitting electrode TX (hereinafter referred to as "TX electrode") and the receiving electrode RX (hereinafter referred to as "RX electrode") is measured. When a finger comes close to the electrode, a portion of the electromagnetic field moves to the human body, which is the conductor, causing the capacitance between the electrodes to decrease. By periodically observing the capacitance change between these electrodes, the approaching of the human body is detected.

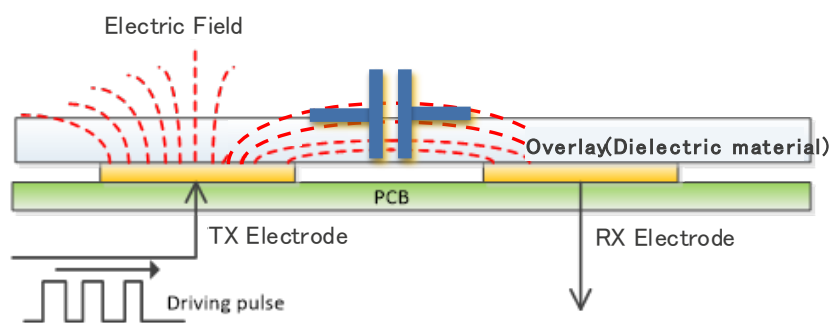


Figure 2.5 The human body is NOT approached

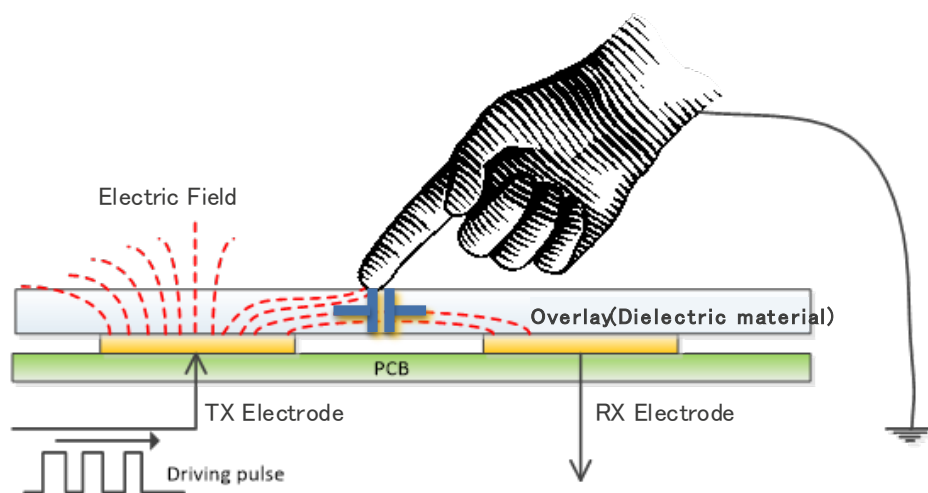


Figure 2.6 Approaching the human body

2.3.1 Detection Principle

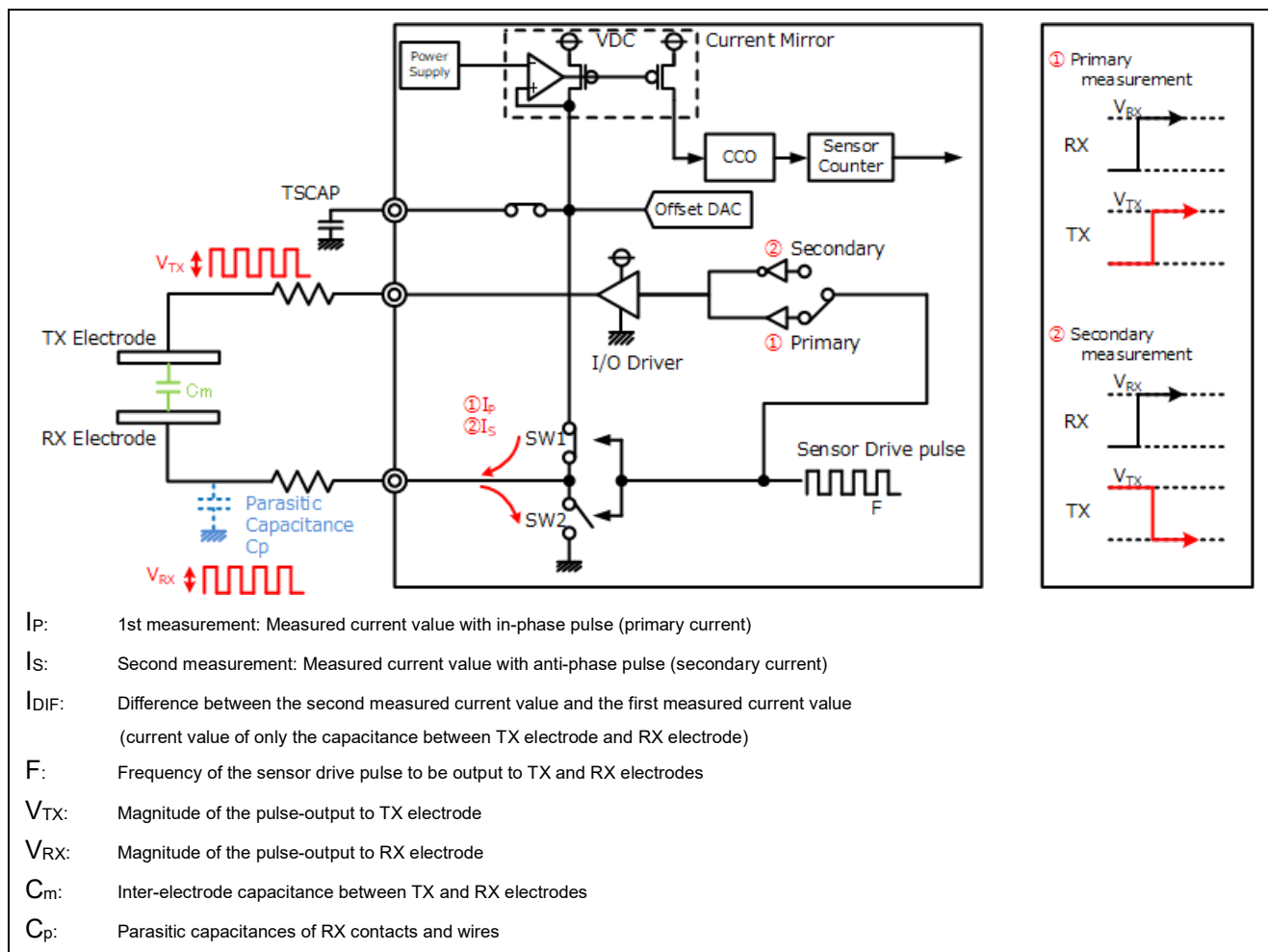


Figure 2.7 Outline of mutual-capacitance method operation

Figure 2.7 shows the configuration of CTSU inside the mutual-capacitance method.

If the frequency F of the sensor drive pulse, the voltage V_{TX} of the TX electrode side, the voltage V_{RX} of the RX electrode side, the parasitic capacitance is C_p , and the capacitance between electrodes is C_m ,

The primary current I_P when in-phase pulse-outputting is

$$I_P = F (C_p V_{RX} + C_m (V_{RX} - V_{TX}))$$

For the secondary current I_S when a anti-phase pulse is output

$$I_S = F (C_p V_{RX} + C_m (V_{RX} + V_{TX}))$$

When the difference between the measurement results is calculated,

$$I_{DIF} = I_S - I_P = F C_m 2V_{TX}$$

The current of the capacitance between the electrodes can be measured.

In this mutual-capacitance method, the parasitic capacitance C_p can be canceled by measuring the same-phase pulse and anti-phase pulse, and calculating the difference between the two measurements. Only the inter-electrodes capacitance C_m to be detected can be measured.

V_{TX} is supplied from the mains voltage. Therefore, the measured value varies depending on the power supply voltage of MCU.

The TSCAP capacitor is used to stabilize the internal voltage of CTSU.

2.3.2 Measurement Range

The sensor counter register must be used within the valid measurement range (upper limit $\leq 100\%$), even though the 16-bit register itself spans 0–65535. The CTSU includes a sensor offset adjustment register that tunes the offset current to compensate for parasitic capacitance components. Adjusting this offset current to bring the measured count closer to the target value within the valid range is referred to as the offset tuning process. This is done by software-processing after resetting MCU in order to adjust individual differences in hardware. Figure shows the offset tuning process of the mutual-capacitance method. The measured value of the current I_P measured at the time of in-phase pulse output is called the primary counter, and the measured value of the current I_S measured at the time of reverse-phase pulse output is called the secondary counter.

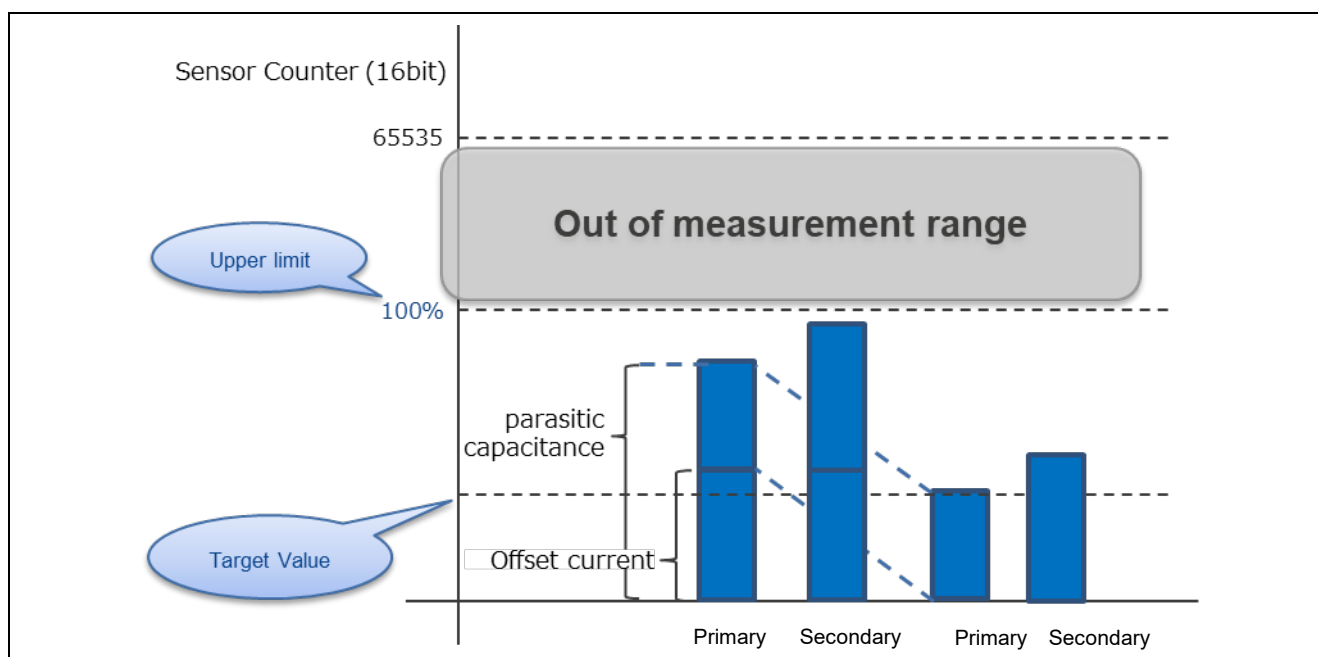


Figure 2.8 Offset tuning process of mutual-capacitance method

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_cts\)](#)

2.4 Drift Correction Processing and Touch Detection

Drift correction is a process that follows changes in capacitance that change when a person is not touching the touch sensor. An example of such changes is those imparted due to the changes in the humidity. Touch detection is the process of detecting the capacitance that changes when a person touches a touch sensor.

Figure 2.9 shows an example of a series of operations until a change in the measured value of self-capacitance method and the baseline and a touch ON are detected.

(1) Drift correction process

Capacitance changes when a finger comes into contact with an electrode placed nearby, or due to temperature, humidity, aging, etc. To reduce these effects, the baseline is updated periodically through drift correction processing.

(2) Touch detection

The capacitance changes when the human body contacts the overlay on the electrode, and the touch ON is confirmed when the touch threshold is exceeded. Since the mutual-capacitance button reduces the inter-electrode capacitance when touching, the touch threshold is set in the decreasing direction of the measured value to judge the touch ON/OFF.

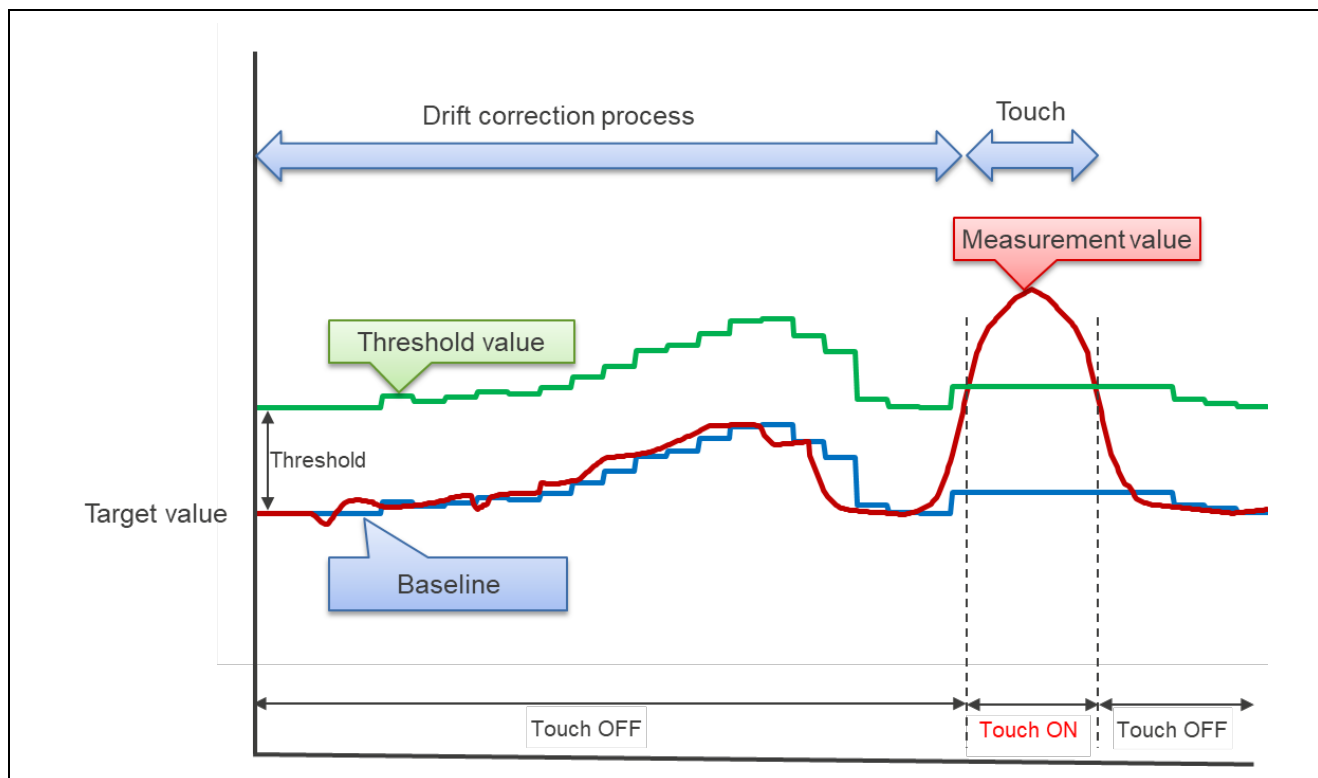


Figure 2.9 Drift correction processing and touch detection of self-capacitance method

For details of operation, refer to the following application note about the TOUCH module.

[RL78 Family TOUCH Module Software Integration System \(R11AN0485\)](#)

[RX Family QE Touch module Firmware Integration Technology \(R01AN4470\)](#)

[RA Flexible Software Package Documentation: Touch \(rm_touch\)](#)

Refer to "7.Tuning" for details such as touch ON/OFF judgment and drift-correction.

3. Capacitive Touch Sensor Type

This chapter describes the types of capacitive touch sensors and the terminology that comes up when using capacitive touch sensors.

3.1 Differences in Capacitive Touch Sensor

Compared with CTSU1, CTSU2 has the following main added function and enhanced features:

- Enhanced noise immunity
 - Synchronous noise avoidance function by multiple sensor drive pulse frequency measurement
- Improved water resistance
 - Improvement of water resistance by the self-capacitance method using an active shield electrode
- Enhanced scan speed
 - Addition of " Current Frequency Converter " (CFC) that supports Mutual capacitance parallel measurement
- Reduction of software processing load
 - Addition of automatic judgment function that judges the touch of a button by hardware without activating CPU.
 - Addition of automatic correction function that corrects sensor CCO (ICO) with hardware without activating CPU.

Table 3.1 compares the capacitive touch sensor functions installed in the 2nd and 3rd generations.

Table 3.1 Comparison of the capacitive touch sensor functions

Function	CTSU/ CTSUa	CTSUB	CTSU2	CTSU2L	CTSU2La	CTSU2SL	CTSU2SLa
Self-capacitance method	✓	✓	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	△	△	△	△	△
Multi-clock measurement			✓	✓	✓	✓	✓
Active shield			✓	✓	✓	✓	✓
Automatic judgment				✓*1 *2	✓*1 *2	✓ *2	✓*3
Automatic correction				✓*1	✓*1	✓	✓
Multi-Clock Automatic Correction							✓
Multiple Electrode Connection (MEC) *4		✓			✓	✓	✓
Mutual capacitance parallel measurement			✓				

✓ : Supported

△ : This function is not supported by the driver because it supports multi-clock measurement.

*1 : Achieved when used with SNOOZE-mode programmable controller (SMS). See Table 4-12 for supported products.

*2 : Automatic judgment is measured by JMM. For details, refer to 3.3.1(2) Judgement Majority Mode(JMM).

*3 : Automatic judgment is measured by VMM or JMM.

For details, refer to 3.3.1(1) Value Majority Mode(VMM).

For details, refer to 3.3.1(2) Judgement Majority Mode(JMM).

*4 : Supports self-capacitance methods.

3.2 CTSU1

3.2.1 Random Pulse Frequency Measurement

"Random pulse frequency measurement" is a method of reducing the effect of noise when measuring capacitance, so that the noise and the sensor drive pulse are not synchronized.

Capacitance is measured by outputting a sensor drive pulse. When noise is mixed in this sensor drive pulse, the measurement result fluctuates. For this reason, CTSU1 reduces the effect of noise by outputting random pulses that are not synchronized with noise by spread spectrum or phase shifting. Figure 3.1 shows the sensor drive pulse output waveform of CTSU1.

[RL78 Family TOUCH Module Software Integration System \(R11AN0485\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_ ctsu\)](#)

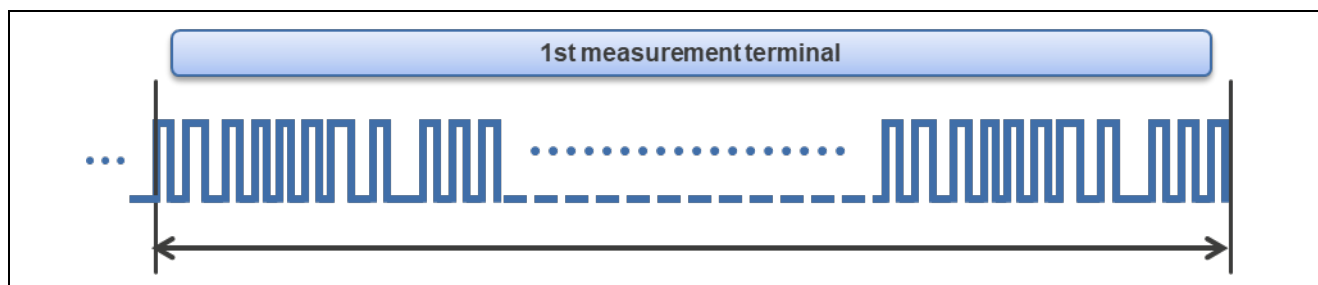


Figure 3.1 Sensor drive pulse output waveform of CTSU1

3.3 CTS2

3.3.1 Multi-clock Measurement

"Multi-clock measurement" is a method of measuring by using multiple different frequencies for the sensor drive pulse to reduce the effect of noise when measuring capacitance.

Capacitance is measured by outputting a sensor drive pulse. When noise is mixed in this sensor drive pulse, the measurement result fluctuates. For this reason, CTS2 is equipped with a multi-clock measurement function that enables measurement with several clocks with differing frequencies. Figure 3.2 shows an example of the sensor drive pulse waveform when measured at 3 frequencies. Noise immunity is improved by excluding measurement results that may be affected by noise.

Refer to the following for details.

[RL78 Family CTS2 Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTS2 module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTS2 \(r_ctsu\)](#)

[Capacitive Sensor MCU QE for Capacitive Touch Advanced mode Parameter Guide \(R30AN0428\)](#)

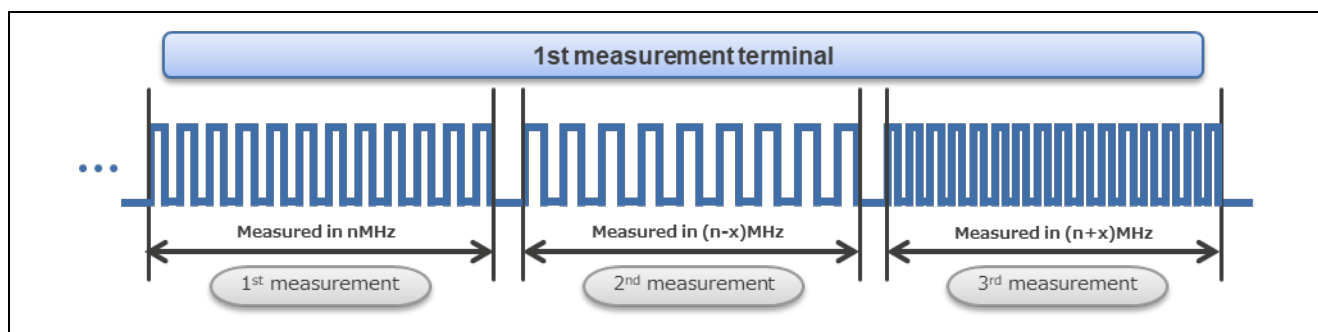


Figure 3.2 Example of the sensor drive pulse waveform when measured at 3 frequencies

There are two measurement methods for multi-clock measurement touch judgment: "Value Majority Mode" and "Judgement Majority Mode".

In multi-clock measurement, the measurement starts after executing offset tuning for each of three frequencies.

(1) Value Majority Mode(VMM)

Value Majority Mode (VMM) is a method to determine the value by adding two measured values that are close to each other from the measurement results of three frequencies.

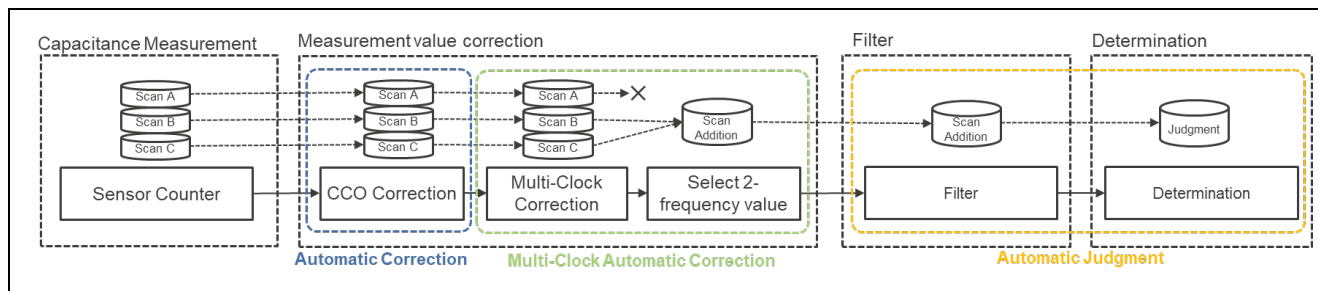


Figure 3.3 VMM signal flow

After performing CCO correction on the measurement results of each of the three frequencies, the values of Scan B and Scan C are corrected based on the measurement frequency of Scan A (multi-clock correction), since the measured value increases or decreases depending on the measurement frequency. Select two measured values with close multi-clock correction results and add them together. Perform filter processing to reduce noise in the measured values to determine touch. Figure 3.3 shows the signal flow of the VMM.

VMM by software supports button judgment, slider/wheel position detection, etc., VMM by while automatic judgment supports only button judgment.

The automatic judgment function is realized by a combination of automatic correction and multi-clock automatic correction. Automatic judgment of VMM is supported in QE for Capacitive Touch Ver. 4.1. 0 or later.

(2) Judgement Majority Mode(JMM)

Judgement Majority Mode(JMM) is a method in which each of the three frequency measurements is judged by a majority vote.

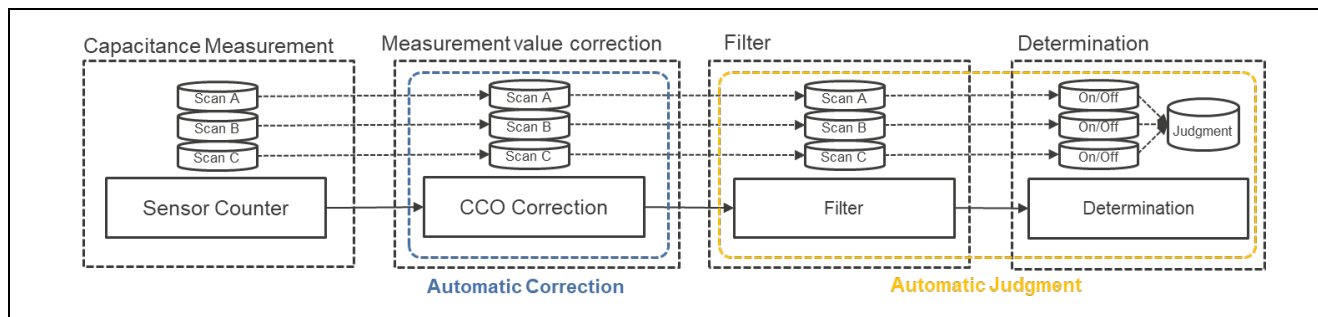


Figure 3.4 JMM signal flow

CCO correction is applied to the measurement results of each of the three frequencies, and filter processing is performed to reduce noise. An On/Off decision is made for each of the measurement frequencies, and the majority vote is used to make the decision. Figure 3.4 shows the JMM signal flow.

JMM by software and JMM by automatic judgment only support button judgment of self-capacitance and mutual capacitance methods. Slider/wheel and mutual capacitance parallel measurement are not supported by JMM. The automatic judgment function is realized by a combination of automatic correction.

Software JMM is supported in QE for Capacitive Touch Ver.4.0.0 or later.

3.3.2 Active Shield

"Active shield" is a method of noise suppression while reducing the effect of capacitive coupling between the electrode and the shield pattern. As an operation, the shield pattern is driven by a signal of the same potential and phase as the electrode being measured.

This function drives the shield pattern with the same potential and phase as the electrode being measured. Using the active shield will reduce the effect of capacitance coupling between the electrode and shield pattern as well reduce noise interference.

GND shield shown in Figure 3.5, if the capacitance is increased by water drop while measuring one button, it can cause a malfunction. By using an active shield as shown in Figure 3.6, the influence of the capacitance generated in the other electrode can be reduced, thus preventing malfunction.

The active shield can only be used with the self-capacitance method. It cannot be used in the mutual-capacitance method.

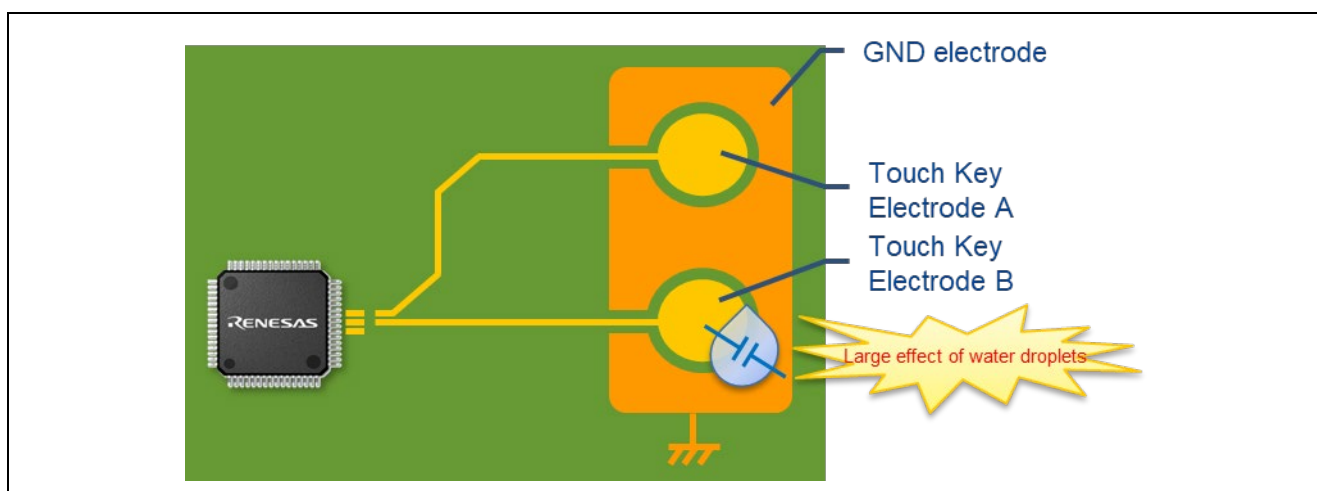


Figure 3.5 GND shield

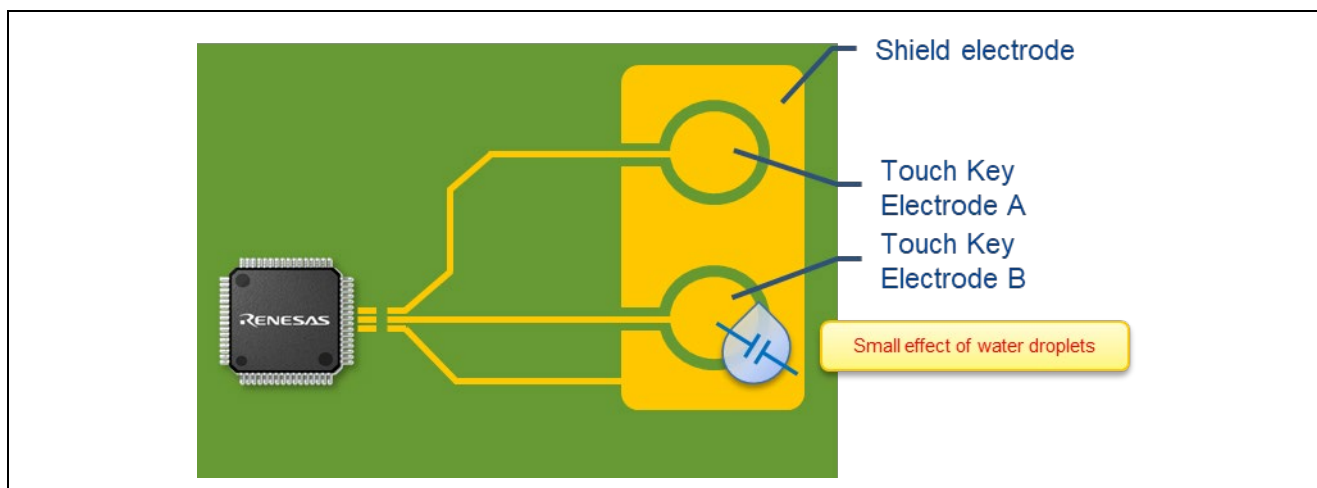


Figure 3.6 Active Shield

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_cts\)](#)

[Capacitive Sensor Microcontrollers CTSU Capacitive Touch Electrode Design Guide \(R30AN0389\)](#)

3.3.3 Automatic Judgment

Automatic judgement (Auto Judgement) functions to judge the touch of a button by hardware without activating CPU. Therefore, low power consumption operation can be realized.

To use this function, select the capacitance sensor equipped with the automatic judgment function.

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_atsu\)](#)

[Capacitive Sensor MCU QE for Capacitive Touch Advanced Mode Parameter Guide\(R30AN0428\)](#)

3.3.4 Automatic Correction

Automatic correction(Automatic Interpolation) function to correct sensor CCO(ICO) with hardware. It does not consume the main processor processing time because the hardware handles the correction calculation without using the software's correction calculation processing.

Refer to the following for details on CCO correction and automatic correction.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_atsu\)](#)

[Capacitive Sensor MCU QE for Capacitive Touch Advanced Mode Parameter Guide\(R30AN0428\)](#)

3.3.5 Multi-Clock Automatic Correction

This is a function that performs multi-clock correction after 3-frequency measurement in hardware.

It does not consume the main processor processing time because the hardware handles the correction calculation.

Multi-Clock Automatic Correction is supported in QE for Capacitive Touch Ver. 4.1. 0 or later.

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System\(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology\(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_atsu\)](#)

[Capacitive Sensor MCU QE for Capacitive Touch Advanced Mode Parameter Guide\(R30AN0428\)](#)

3.3.6 Multiple Electrode Connection (MEC)

Multiple Electrode Connection (MEC) is a function of connecting self-capacitance type electrodes inside CTSU to measure them as a single electrode. If you use MEC, you can use any key to return from standby mode. In addition, since the number of measurements can be shortened by making multiple measurements into one, the power consumption can be reduced. In addition, multiple electrodes can be used as electrodes for one large proximity sensor. Due to increased parasitic capacitance of the MEC when compared to a single electrode the CTSU2 settings and software settings are adjusted to compensate - an example of this is the scanning frequency being reduced to keep the operation stable across the dynamic range.

Figure 3.7 shows operation image of MEC. The diagram on the left shows MEC function disabled, and the diagram on the right shows MEC function enabled, which can be switched by software.

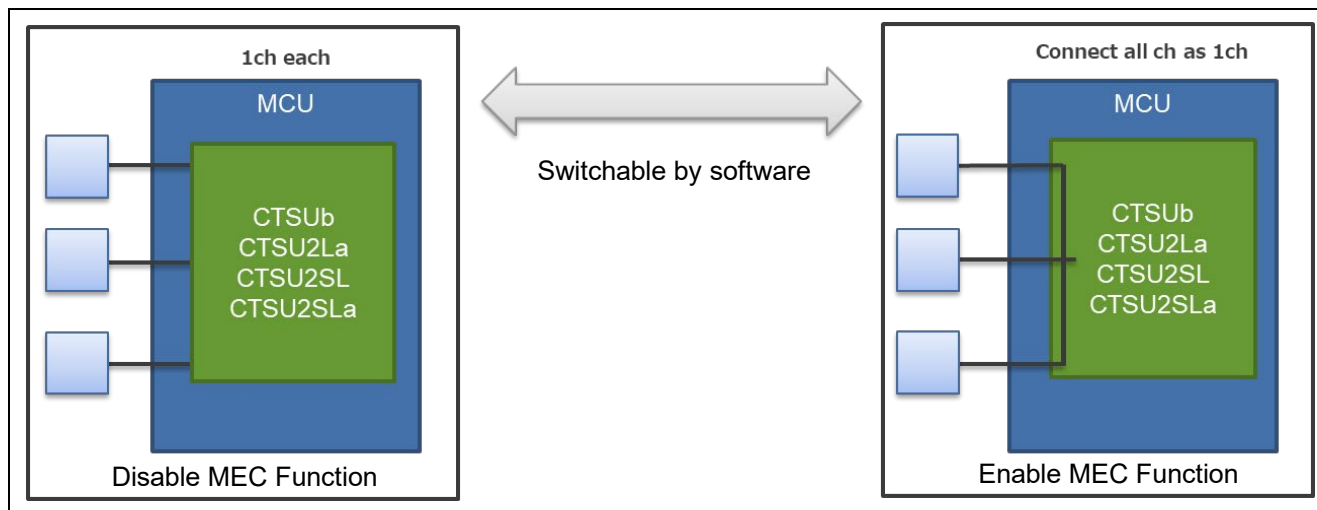


Figure 3.7 Operation image of MEC

To use this function, select a capacitance sensor equipped with a MEC function.

Refer to the following for details.

[RL78 Family CTSU Module Software Integration System \(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology \(R01AN4469\)](#)

[RX140 Group Smart Wakeup Solution \(R11AN0613\)](#)

3.3.7 Mutual capacitance parallel measurement

By using CFC pin as the receiving pin and switching to the mutual capacitance parallel measurement mode, it is possible to simultaneously measure several pins.

Refer to the following for details.

[RA Flexible Software Package Documentation: CTSU \(r_ctsu\)](#)

4. Capacitive Sensor MCU

4.1 MCU Lineup

Renesas offers a broad lineup of MCU with capacitive sensors for all of RA, RX, RL78 Family MCU, Renesas Synergy™.

Please refer to the following for details and the latest information of each of these families.

- [RA Family of Arm® Cortex®-M based MCUs](#)
- [RX 32-Bit Performance / Efficiency MCUs](#)
- [RL78 Low Power 8 & 16-bit MCUs](#)
- [Renesas Synergy™ Platform MCUs](#)

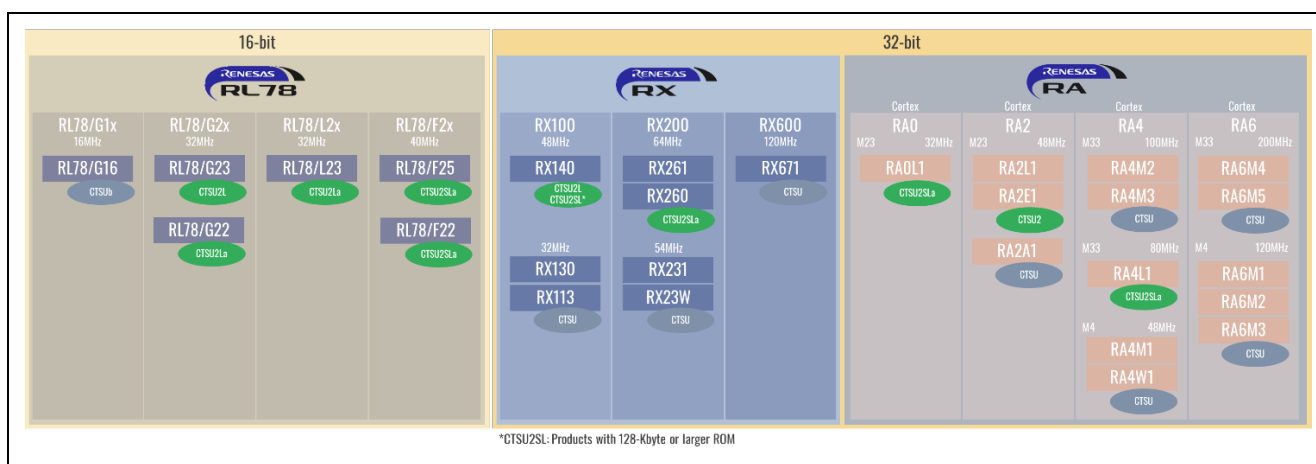


Figure 4.1 Lineup of capacitive sensor MCU

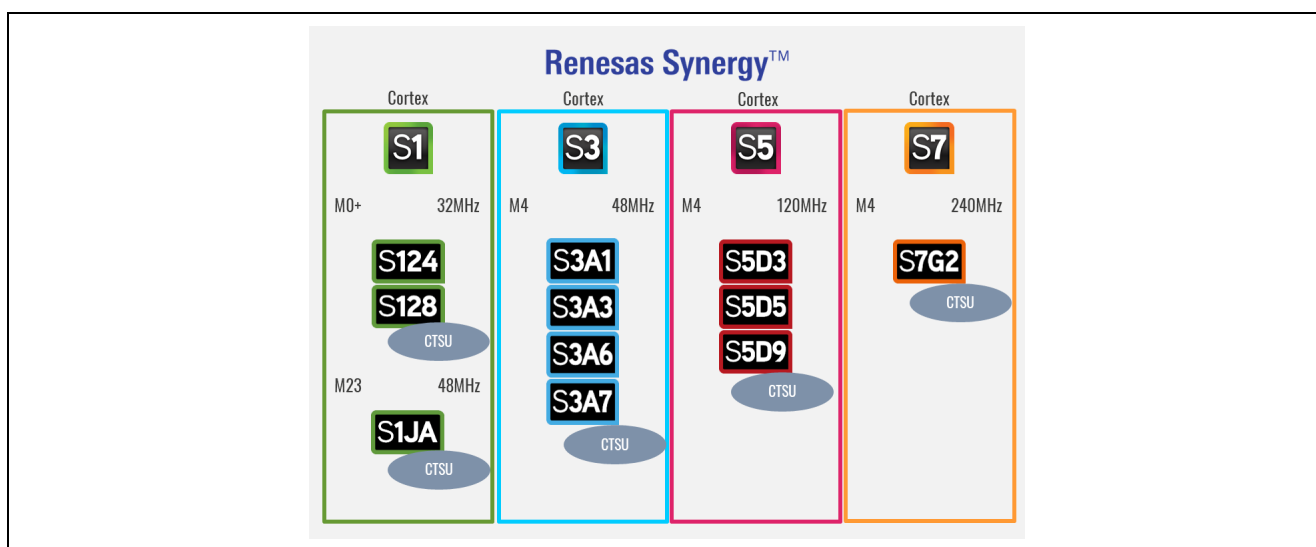


Figure 4.2 Lineup of capacitive sensor MCU (4Renesas Synergy™)

4.2 Points for MCU Selection

Selection of MCU with capacitive touch sensor has several points. First of all, it is important to find out what kind of interface applying the capacitive touch sensor should be used in your product. When the interface is determined, the number of TS terminals required for that interface is determined, and options can be narrowed down. Furthermore, by verifying MCU peripheral functions that are required in addition to the electrostatic capacitance touch sensor, the optimal MCU of mounting the electrostatic capacitance touch sensor for the customer is determined.

4.2.1 Interface

This section introduces the interface in the capacitive touch sensor solution provided by Renesas.

The interface includes a self-capacitance button, a self-capacitance slider, a self-capacitance wheel, and a mutual-capacitance button (keypad). Each feature is as follows.

(1) Self-capacitance button

This interface reacts when a finger is touched and is ideal for replacing mechanical contact switches.

Only the self-capacitance method can be used. It can be configured with one TS pin.

(2) Self-capacitance slider and self-capacitance wheel

The slider detects the position of the finger when it is moved up and down, left and right, and the wheel detects the angle when the finger is moved in a circle. These two interfaces differ from each other in that they do not have both ends of TS terminal (the wheel is applicable) or have both ends (the slider is applicable). Otherwise, they are almost the same interface.

Sliders and wheels can only be used with the self-capacitance method. Sliders can be configured with three to ten TS terminals and wheels with four or eight TS terminals.

(3) Mutual-capacitance button (Keypad)

This is the same as a button, but with a matrix of TS terminals, you can configure many buttons with fewer TS jacks, ranging from 2 buttons (1 x 2) to 64 buttons (8 x 8).

Only the mutual-capacitance method can be used, and two TS pins can be configured.

Table 4.1 Supported interfaces and required numbers of TS pins for each sensing method

Interface	TS number of terminal
Self-capacitance button	1
Self-capacitance wheel	4, 8
Self-capacitance slider	3~10
Mutual-capacitance button (Keypad)	2

Note: Indicates the minimum required TS pin count.

4.2.2 Features of Detection Method

The features of the detection method are shown below.

- Features of self-capacitance method

There are few items that must be considered when wiring on a printed circuit board, making layout easy. In addition, you can configure the electrodes of the sliders and wheels.

Water resistance is not very strong, so if water resistance is required, CTSU2 will make one terminal active-shielded, which will increase water resistance as with the mutual-capacitance method. (See Figure 4.3)

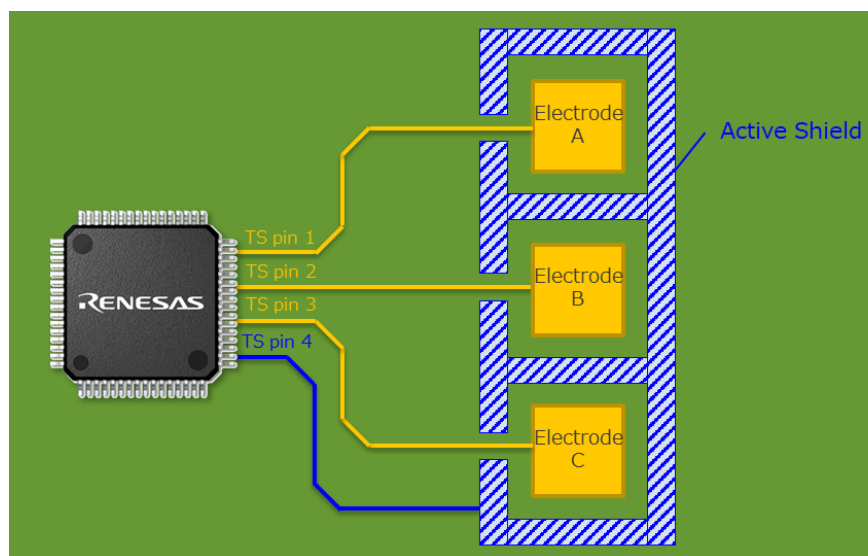


Figure 4.3 Electrode layout of self-capacitance method

- Features of mutual-capacitance method

The keypad can be configured with a matrix of buttons with a small number of terminals, and it is also water resistant. However, there are more items to consider when wiring, such as the limit of the wiring between the transmitting and receiving electrodes and the parasitic capacitance value, compared with the self-capacitance.

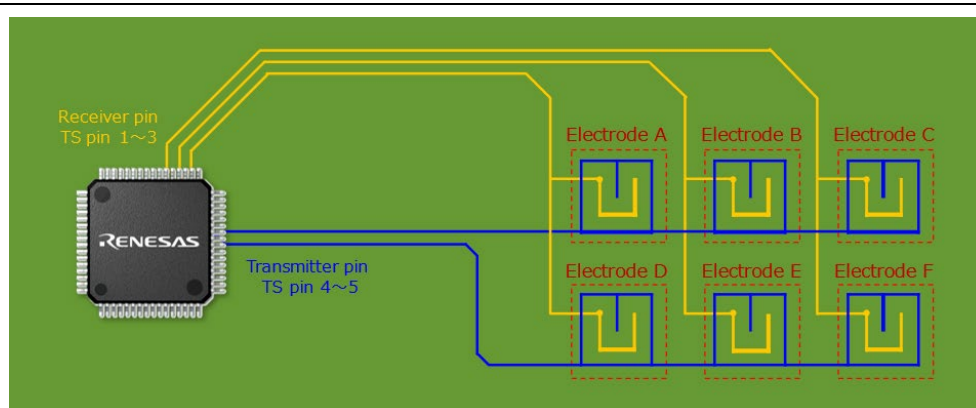


Figure 4.4 Electrode layout of mutual-capacitance method

Refer to the following for details of the electrode layout.

[Capacitive Sensor Microcontrollers CTSU Capacitive Touch Electrode Design Guide \(R30AN0389\)](#)

- Precautions on detection method and wiring

As shown in Figure 4.3, the self-capacitance method configures one electrode for one TS terminal. However, for the mutual-capacitance method, as shown in Figure 4.4, the mutual-capacitance method configures several electrodes with one TS terminal. Therefore, there are more restrictions on the wiring between the transmitting and receiving electrodes, parasitic capacitance values, etc. compared to the self-capacitance method.

Refer to the following for details.

[Capacitive Sensor Microcontrollers CTSU Capacitive Touch Electrode Design Guide \(R30AN0389\)](#)

4.3 MCUs Embedding CTSU1

Table 4.2 to Table 4.9 compare the functions of MCU embedding CTSU1(CTSU/CTSUa/CTSUB). Refer to each product introduction page for details of each device.

You can narrow down the product in more detail in the product selection field at the bottom of the product introduction page.

Table 4.2 RA family MCU embedding CTSU1 (1/2)

Name of family product	RA2A1	RA4M1	RA4M2	RA4M3	RA4W1
CPU cores	ARM CM23	ARM CM4	ARM CM33	ARM CM33	ARM CM4
Max. operating frequency (MHz)	48	48	100	100	48
Power supply voltage (V)*1	1.6 - 5.5	1.6 - 5.5	2.7 - 3.6	2.7 - 3.6	1.8 - 3.6
Program Memory (KB)	256	256	256, 384, 512	512, 768, 1024	512
Data Flash (KB)	8	8	8	8	8
RAM (KB)	32	32	128	128	96
Capacitance sensor type	CTSU				
TS number of terminal	26	27	12	20	11
Self-capacitance method	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.3 RA family MCU embedding CTSU1 (2/2)

Name of family product	RA6M1	RA6M2	RA6M3	RA6M4	RA6M5
CPU cores	ARM CM4	ARM CM4	ARM CM4	ARM CM33	ARM CM33
Max. operating frequency (MHz)	120	120	120	200	200
Power supply voltage (V)*1	2.7 - 3.6	2.7 - 3.6	2.7 - 3.6	2.7 - 3.6	2.7 - 3.6
Program Memory (KB)	512	512, 1024	1024, 2048	512, 768, 1024	1024, 1536, 2048
Data Flash (KB)	8	32	64	8	8
RAM (KB)	256	384	640, 384	256	512
Capacitance sensor type	CTSU				
TS number of terminal	19	18	18	20	12
Self-capacitance method	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.4 RX family MCU embedding CTSU1 (1/2)

Name of family product	RX113	RX130	RX230	RX231	RX23W
CPU cores	RXv1	RXv1	RXv2	RXv2	RXv2
Max. operating frequency (MHz)	32	32	54	54	54
Power supply voltage (V)*1	1.8 - 3.6	1.8 - 5.5	1.8 - 5.5	1.8 - 5.5	1.8 - 3.6
Program Memory (KB)	128, 256, 384, 512	64, 128, 256, 384, 512	128, 256	128, 256, 384, 512	384, 512
Data Flash (KB)	8	8	8	8	8
RAM (KB)	64, 32	10, 16, 48, 32	32	32, 64	64
Capacitance sensor type	CTSU	CTSUa	CTSU		
TS number of terminal	12	36	24	24	12
Self-capacitance method	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.5 RX family MCU embedding CTSU1 (2/2)

Name of family product	RX671
CPU cores	RXv3
Max. operating frequency (MHz)	120
Power supply voltage (V)*1	2.7 - 3.6
Program Memory (KB)	1024, 1536, 2048
Data Flash (KB)	8
RAM (KB)	384
Capacitance sensor type	CTSUa
TS number of terminal	17
Self-capacitance method	✓
Mutual-capacitance method	✓
Random pulse frequency measurement	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.6 Renesas Synergy™ embedding CTSU1 (1/3)

Name of family product	S124	S128	S1JA	S3A1	S3A3
CPU cores	ARM CM0+	ARM CM0+	ARM CM23	ARM CM4	ARM CM4
Max. operating frequency (MHz)	32	32	48	48	48
Power supply voltage (V)*1	1.6 - 5.5	1.6 - 5.5	1.6 - 5.5	1.6 - 5.5	1.6 - 5.5
Program Memory (KB)	64, 128	256	256	1024	512
Data Flash (KB)	4	4	8	8	8
RAM (KB)	16	24	32	192	96
Capacitance sensor type	CTSU				
TS number of terminal	31	28	26	27	27
Self-capacitance method	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.7 Renesas Synergy™ embedding CTSU1 (2/3)

Name of family product	S3A6	S3A7	S5D3	S5D5	S5D9
CPU cores	ARM CM4	ARM CM4	ARM CM4	ARM CM4	ARM CM4
Max. operating frequency (MHz)	48	48	120	120	120
Power supply voltage (V)*1	1.6 - 5.5	1.6 - 5.5	2.7 - 3.6	2.7 - 3.6	2.7 - 3.6
Program Memory (KB)	256	1024	512	512, 1024	1024, 2048
Data Flash (KB)	8	16	8	32	64
RAM (KB)	32	192	256	384	640
Capacitance sensor type	CTSU				
TS number of terminal	27	31	12	18	18
Self-capacitance method	✓	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓	✓
Random pulse frequency measurement	✓	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.8 Renesas Synergy™ embedding CTSU1 (3/3)

Name of family product	S7G2
CPU cores	ARM CM4
Max. operating frequency (MHz)	240
Power supply voltage (V)*1	2.7 - 3.6
Program Memory (KB)	3072, 4096
Data Flash (KB)	64
RAM (KB)	640
Capacitance sensor type	CTSUS
TS number of terminal	18
Self-capacitance method	✓
Mutual-capacitance method	✓
Random pulse frequency measurement	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

Table 4.9 RL78 family MCU embedding CTSU1

Name of family product	RL78/G16
CPU cores	RL78-S2
Max. operating frequency (MHz)	16
Power supply voltage (V)*1	2.4 - 5.5
Program Memory (KB)	16, 32
Data Flash (KB)	1
RAM (KB)	2
Capacitance sensor type	CTSUB
TS number of terminal	15
Self-capacitance method	✓
Mutual-capacitance method	✓
Random pulse frequency measurement	✓
Multiple Electrode Connection: MEC *2	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

*2 : Supports self-capacitance methods.

4.4 MCUs Embedding CTSU2

CTSUS2(CTSUS2, CTSUS2L, CTSUS2La, CTSUS2SL) Table 4.10 to Table 4.13 compare the function outline of the installed MCU. The power supply voltage (V) in this table may differ from the operating voltage of the capacitive touch sensor.

Refer to each product introduction page for details of each device.

You can narrow down the product in more detail in the product selection field at the bottom of the product introduction page.

Table 4.10 RA family MCU embedding CTSU2

Name of family product	RA0L1	RA2E1	RA2L1	RA4L1
CPU cores	ARM CM23	ARM CM23	ARM CM23	ARM CM33
Max. operating frequency (MHz)	32	48	48	80
Power supply voltage (V)*1	1.6 - 5.5	1.6 - 5.5	1.6 - 5.5	1.6 - 3.6
Program Memory(KB)	32, 64	32, 64, 128	128, 256	256, 512
Data Flash(KB)	1	4	8	8
RAM(KB)	16	16	32	64
Capacitance sensor type	CTSUS2SLa	CTSUS2	CTSUS2	CTSUS2SLa
TS number of terminal	24	30	32	12
Self-capacitance method	✓	✓	✓	✓
Mutual-capacitance method	✓	✓	✓	✓
Multi-clock measurement t	✓	✓	✓	✓
Active shield *2	✓	✓	✓	✓
Automatic judgment *3	✓			✓
Automatic correction	✓			✓
Multi-Clock Automatic Correction *4	✓			✓
Multiple Electrode Connection: MEC *5	✓			✓
Mutual capacitance parallel measurement *2		✓	✓	
Measurement voltage setting *6	✓	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

*2 : Measurement voltage setting does not support low voltage (ATUNE0=1).

*3 : Automatic judgment is measured by VMM or JMM.

QE for Capacitive Touch Ver. 4.1.0 or later supports the automatic judgment of VMM.

*4 : Supported by QE for Capacitive Touch Ver. 4.1.0 or later.

*5 : Supports self-capacitance methods.

*6 : QE for Capacitive Touch Ver. 4.0.0 or later supports low measurement voltage (ATUNE0=1).

However, switching the measurement voltage during microcontroller operation is not supported.

Table 4.11 RX family MCU embedding CTSU2

Name of family product	RX140	RX260	RX261
CPU cores	RXv2	RXv3	RXv3
Max. operating frequency (MHz)	48	64	64
Power supply voltage (V)*1	1.8 - 5.5	1.6 - 5.5	1.6 - 5.5
Program Memory(KB)	64	128, 256	256,384,512
Data Flash(KB)	4	8	8
RAM(KB)	16	32, 64	128
Capacitance sensor type	CTSU2L	CTSU2SL	CTSU2SLa
TS number of terminal	12	36	34
Self-capacitance method	✓	✓	✓
Mutual-capacitance method	✓	✓	✓
Multi-clock measurement	✓	✓	✓
Active shield *2	✓	✓	✓
Automatic judgment		✓*3	✓*4
Automatic correction		✓	✓
Multi-Clock Automatic Correction *5			✓
Multiple Electrode Connection: MEC *6		✓	✓
Mutual capacitance parallel measurement			
Measurement voltage setting *7	✓	✓	✓

✓ : Supported

*1 : The power supply voltage (V) may differ from the operating voltage of the capacitive touch sensor.

*2 : Measurement voltage setting does not support low voltage (ATUNE0=1).

*3 : Automatic judgment is measured by JMM.

*4 : Automatic judgment is measured by VMM or JMM.

QE for Capacitive Touch Ver. 4.1.0 or later supports the automatic judgment of VMM.

*5 : Supported by QE for Capacitive Touch Ver. 4.1.0 or later.

*6 : Supports self-capacitance methods.

*7 : QE for Capacitive Touch Ver. 4.0.0 or later supports low measurement voltage (ATUNE0=1).

However, switching the measurement voltage during microcontroller operation is not supported.

Table 4.12 RL78 family MCU embedding CTSU2 (1/2)

Name of family product	RL78/G23	RL78/G22	RL78/L23
CPU cores	RL78-S3	RL78-S3	RL78-S3
Max. operating frequency (MHz)	32	32	32
Power supply voltage (V)*1	1.6 - 5.5	1.6 - 5.5	1.6 - 5.5
Program Memory(KB)	96, 128, 192, 256, 384, 512, 768	32, 64	64, 128, 256, 512
Data Flash(KB)	8	2	8
RAM(KB)	16, 12, 48, 32, 24, 20	4	16, 32
Capacitance sensor type	CTSU2L	CTSU2La	CTSU2La
TS number of terminal	32	29	36
Self-capacitance method	✓	✓	✓
Mutual-capacitance method	✓	✓	✓
Multi-clock measurement	✓	✓	✓
Active shield *2	✓	✓	✓
Automatic judgment *3 *4	✓	✓	✓
Automatic correction *3	✓	✓	✓
Multi-Clock Automatic Correction			
Multiple Electrode Connection: MEC *5		✓	✓
Mutual capacitance parallel measurement			
Measurement voltage setting *6	✓	✓	✓

✓ : Supported

*1 : The operating voltage requirement for CTSU2L and CTSU2La is VDD = 1.8 to 5.5 V.

When using CTSU2L and CTSU2La, VDD = 1.8 to 5.5 V.

*2 : Measurement voltage setting does not support low voltage (ATUNE0=1).

*3 : Achieved when used with SNOOZE-mode programmable controller (SMS)

*4 : Automatic judgment is measured by JMM.

*5 : Supports self-capacitance methods.

*6 : QE for Capacitive Touch Ver. 4.0.0 or later supports low measurement voltage (ATUNE0=1).

However, switching the measurement voltage during microcontroller operation is not supported.

Table 4.13 RL78 family MCU embedding CTSU2 (2/2)

Name of family product	RL78/F25	RL78/F22
CPU cores	RL78/S3	RL78/S3
Max. operating frequency (MHz)	40	40
Power supply voltage (V)	2.7 - 5.5	2.7 - 5.5
Program Memory(KB)	512	128
Data Flash(KB)	16	8
RAM(KB)	40	12
Capacitance sensor type	CTSU2SLa	CTSU2SLa
TS number of terminal	31	12
Self-capacitance method	✓	✓
Mutual-capacitance method	✓	✓
Multi-clock measurement	✓	✓
Active shield	✓	✓
Automatic judgment *1	✓	✓
Automatic correction	✓	✓
Multi-Clock Automatic Correction	✓	✓
Multiple Electrode Connection: MEC *2	✓	✓
Mutual capacitance parallel measurement		
Measurement voltage setting		

✓ : Supported

*1 : Automatic judgment is measured by VMM or JMM.

*2 : Supports self-capacitance methods.

5. Hardware

5.1 Electrode design guide

Electrode layout patterns and characteristic data of the self-capacitance method and the mutual-capacitance method are listed. Please refer to the following application notes for MCU design and design examples.

[Capacitive Touch Electrode Design Guide \(R30AN0389\)](#)

6. Software

This chapter describes the software involved in capacitive touch keys.

6.1 Software Configuration

This chapter describes the software of programs applying capacitive touch sensors, including CTSU modules and TOUCH modules.

Renesas provides drivers (CTSU Modules) and middleware (TOUCH Modules) to MCU family for the construction of programs that apply capacitive touch sensors. Drivers and middleware are used in conjunction with QE for Capacitive Touch and other developers described below. The tuning result by QE for Capacitive Touch is incorporated into the program as a parameter file, and the definition of the parameter file is reflected in the program by referring to TOUCH module and CTSU module. Figure 6.1 shows the software configuration of a typical program including CTSU module, TOUCH module, and application.

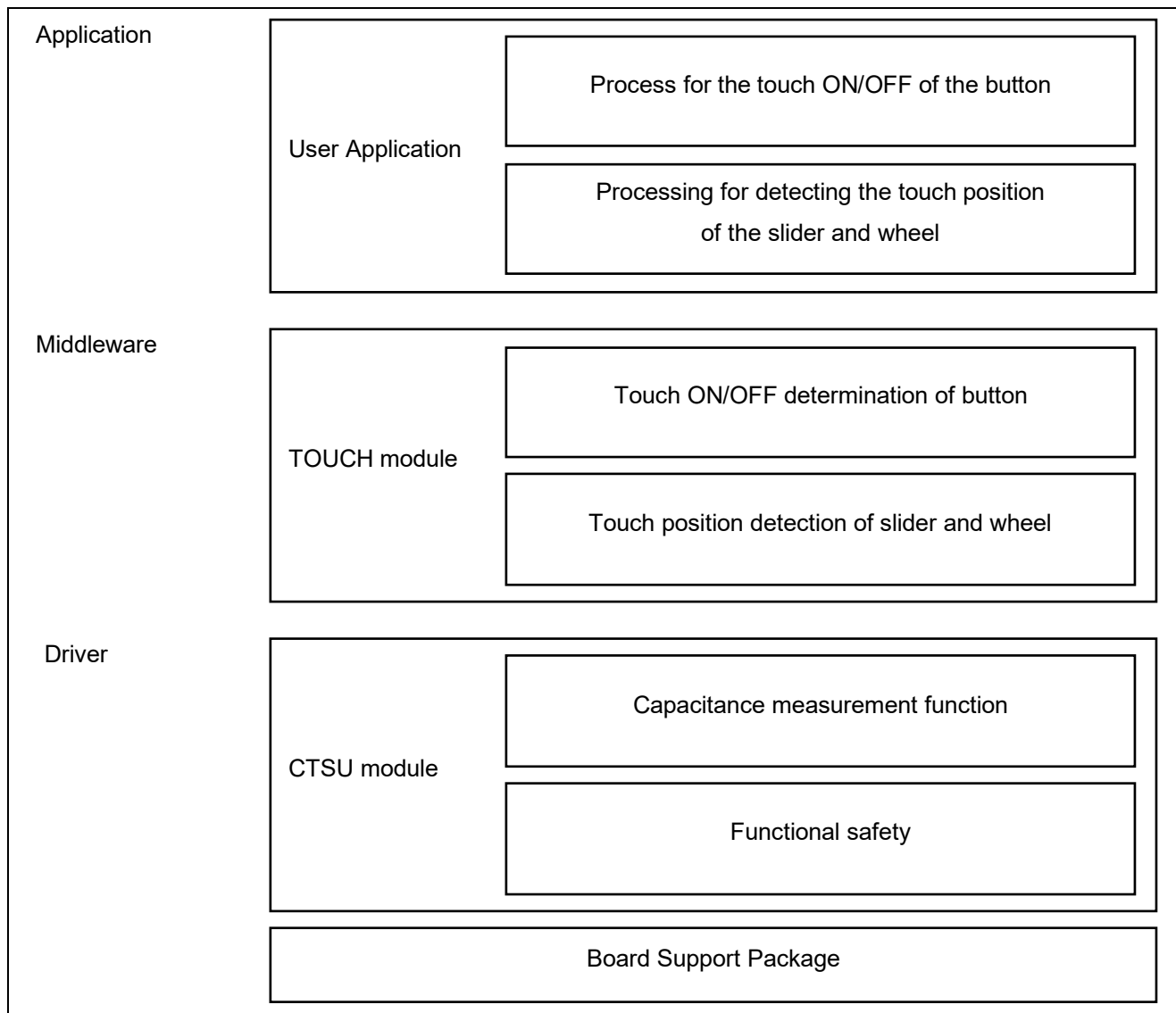


Figure 6.1 Typical software configuration

For an overview of the capacitive touch software, refer to the following application note.

[Capacitive Touch Sensor Microcontrollers Overview of Capacitive Touch Software \(R30AN0470\)](#)

6.1.1 CTSU Module

CTSU modules are drivers for TOUCH modules. CTSU register settings, etc. tuned by QE for Capacitive Touch are entered from the parameter file into the module and used.

The capacitance measurement function provided by this module is shown below.

- CCO compensation control
- Offset tuning
- Random pulse frequency measurement
- Multi-clock measurement
- Active shield control
- Automatic judgment control
- Automatic correction control
- Automatic multi-clock correction
- MEC control
- Mutual capacitance parallel measurement
- Functional safety

Refer to the following for more information on CTSU Modules.

[RL78 Family CTSU Module Software Integration System \(R11AN0484\)](#)

[RX Family QE CTSU module Firmware Integration Technology \(R01AN4469\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_ctsu\)](#)

[Renesas Synergy™ Software Package \(SSP\)](#)

6.1.2 TOUCH Module

TOUCH modules are middleware that use CTSU modules to provide capacitive touch-detection. Parameter setpoints tuned by QE for Capacitive Touch are entered from the parameter file to the module and used.

The main functions provided by this module are as follows.

- Touch ON/OFF determination of button
- Touch position detection of slider and wheel

Refer to the following for more information on TOUCH Modules.

[RL78 Family TOUCH Module Software Integration System \(R11AN0485\)](#)

[RX Family QE Touch module Firmware Integration Technology \(R01AN4470\)](#)

[RA Flexible Software Package Documentation: Touch \(rm_touch\)](#)

[Renesas Synergy™ Software Package \(SSP\)](#)

6.1.3 User Application

"User Application" is an application created by the Customer. Make buttons, sliders and wheel treatments according to your requirements. In the demo set, the processing of buttons is shown as an example.

6.2 Functional Safety

“Functional Safety” is the function to diagnose one's own internal circuitry using self-test software.

Renesas's capacitive touch keys (CTSU1 and CTSU2) allow the self-test software to diagnose its own circuitry. For details on functional safety, refer to APN-below.

[CTSU Self Test Software \(R11AN0512EJ0120\)](#)

6.3 Low Power Guide

An example of setting when using a capacitive touch for low-power applications is provided. Refer to APN below for the low-power guidelines using capacitive touch.

[RA2L1 Group Capacitive Touch Low Power Guide \(R01AN6266\)](#)

[RA6M2 Group Capacitive Touch Low Power Guide \(R01AN6473\)](#)

[RL78 Family RL78/G23 Capacitive Touch Low Power Guide \(SNOOZE function\) \(R01AN5886\)](#)

[RL78/G23 Group RL78/G23 Capacitive Touch Low Power Guide \(SMS function\) \(R01AN6670\)](#)

[RL78/G22 Capacitive Touch Low Power Guide \(SNOOZE function\) \(R01AN7413\)](#)

The application notes that use MEC function are as follows.

[RX140 Group Smart Wakeup Solution \(R11AN0613\)](#)

[RX261/260 Group Smart Wakeup Solution \(R11AN0941\)](#)

[RL78/G22 Capacitive Touch Low Power Guide \(SMS / MEC function\) \(R01AN6847\)](#)

[RL78/L23 Low power consumption Smart Wakeup Solution \(R11AN1044\)](#)

[RL78/F25 Low power consumption Smart Wakeup Solution \(R11AN1045\)](#)

[RA4L1 Group Smart Wakeup Solution \(R11AN1013\)](#)

6.4 Software Filter

“Software Filter” is a function that reduces the noise component and acquires the signal component for the acquired measured value.

We offer an API that enables you to replace the CTSU module's filter with your own filter or to use multiple filters with the built-in filter. Refer to the following application note for details on how to integrate filters into the project file, a software filter sample code, and the project file with usage samples.

[Capacitive Touch Software Filter Sample Program \(R30AN0427\)](#)

[Capacitive Sensor MCU Capacitive Touch Noise Immunity Guide \(R30AN0426\)](#)

7. Tuning

Tuning is the process of determining the value by measuring using an actual machine when determining the parameters to be controlled by software.

Capacitive touch sensors use QE for Capacitive Touch to generate software parameters to accurately capture minute changes in capacitance.

The basic CapTouch parameters are generated by “Auto Tuning” using QE for Capacitive Touch.

If the evaluation using this parameter does not meet the required specifications, perform “7.2 Manually Tuning with CapTouch Parameters”.

If you need to make further adjustments, refer to “7.3 Tuning with advanced mode”.

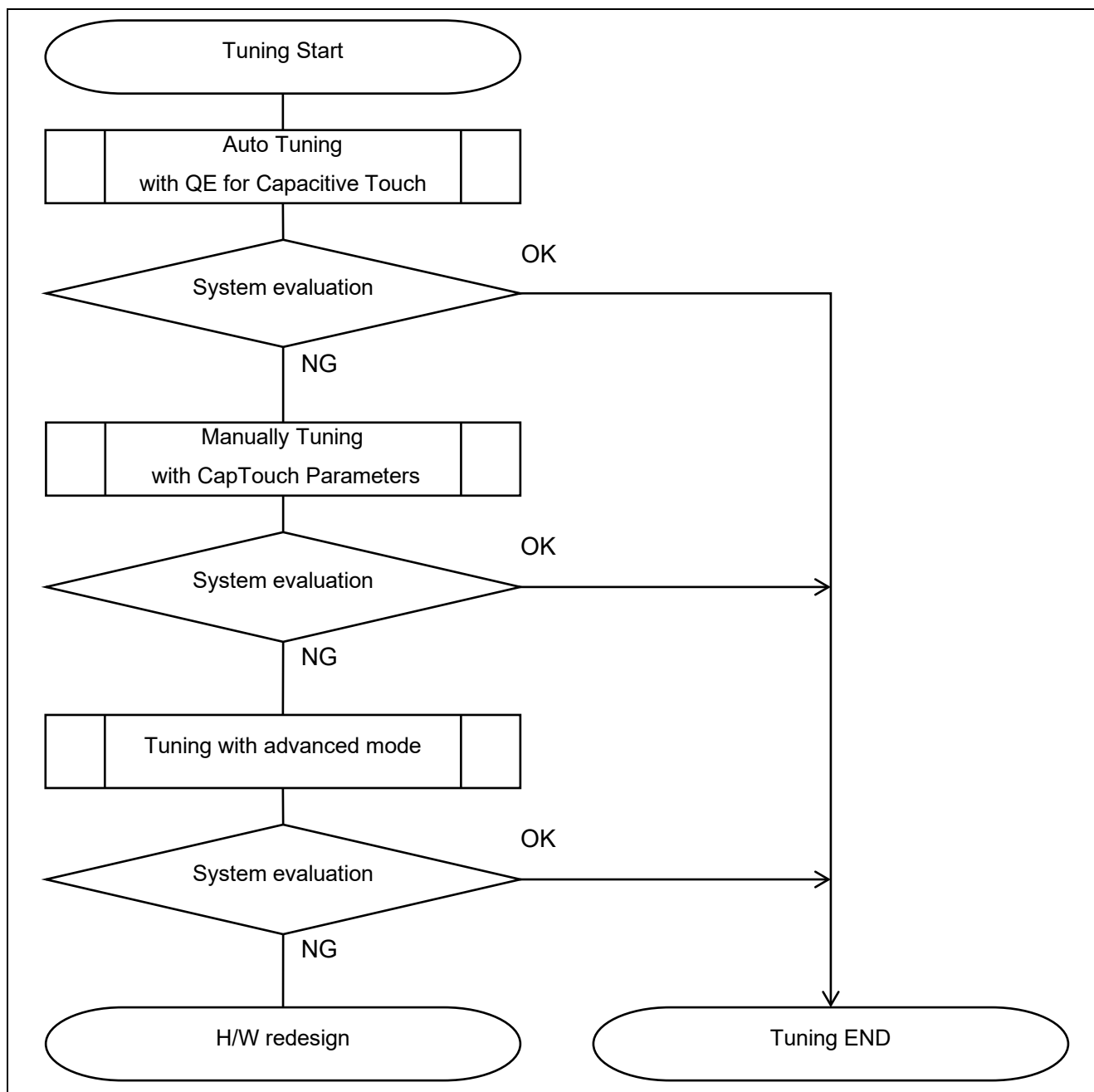


Figure 7.1 Tuning flow chart

7.1 Auto Tuning with QE for Capacitive Touch

Figure 7.2 shows the auto tuning using QE for Capacitive Touch, and the tuning parameters are created by the following process. After resetting MCU, the operation starts based on this parameter.

- Parasitic capacitance measurement

Determine the sensor drive pulse frequency that can be charged and discharged from the parasitic capacitance and resistance value.

- Offset adjustment

By tuning the offset current, the measured value is brought close to the target value within the measurement range.

- Sensitivity measurement (Not touched)

The baseline is determined by measuring the capacitance value in a non-contact state.

- Sensitivity measurement (Touched)

Capacitance change of the finger is detected by measuring the capacitance value with the finger in contact and calculating the difference from the measurement result in the non-contact state. For a button or keypad, set the threshold to 60% of the difference and set the hysteresis to 5% of the difference.

Note: Do not touch the electrode directly. Touch through the overlay.

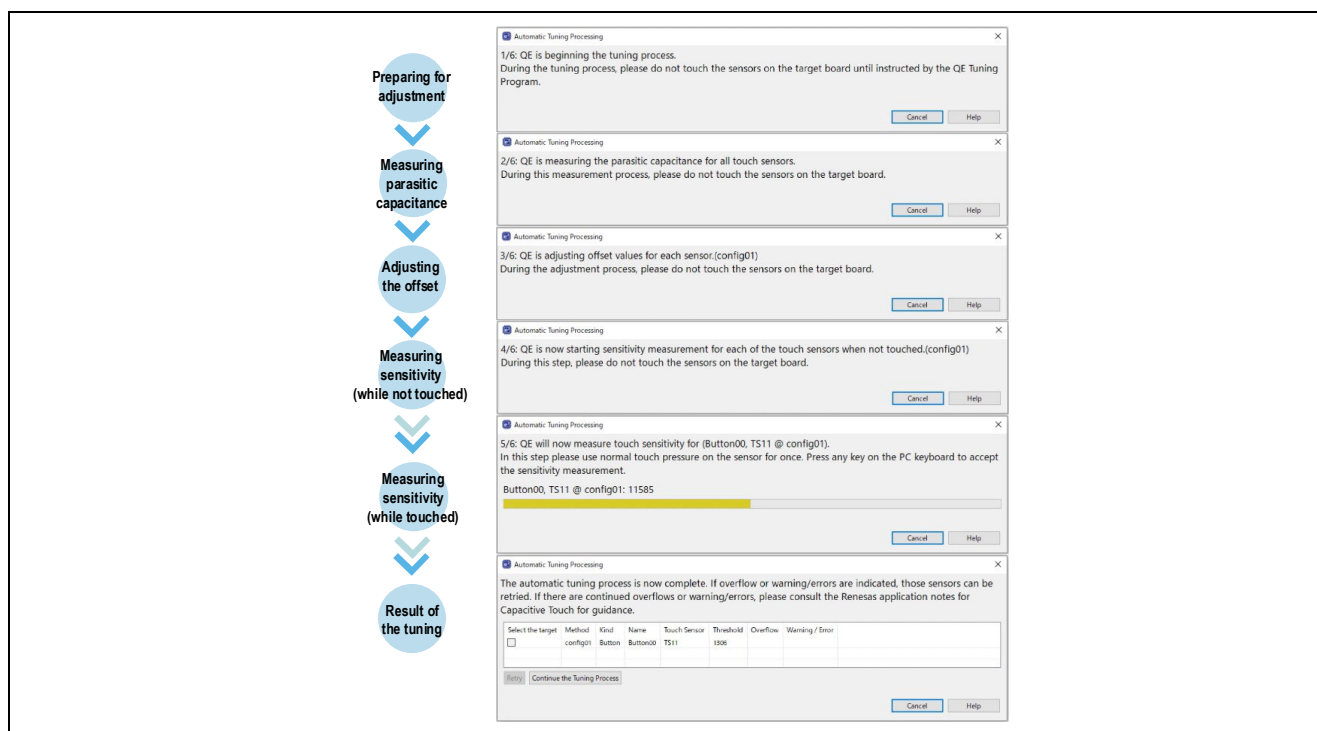


Figure 7.2 Auto tuning using QE for Capacitive Touch

7.2 Manually Tuning with CapTouch Parameters

Manual tuning means that the user changes the parameters as appropriate for auto tuning by QE for Capacitive Touch.

Table 7.1 shows the CapTouch parameters for buttons and keypads to be manually tuned.

Table 7.1 CapTouch parameters

CapTouch parameter	Overview
Average sample count for moving average filter (Depth of the moving average filter)	A moving average filter is used to reduce the effects of noise and specifies the number of moving average operations.
Touch threshold	This value is used to judge the touch ON, and the change required until the measured value is judged from the baseline to the touch ON is specified.
Hysteresis	This value is used to judge the touch OFF, and the change required for the measured value to judge the touch OFF from the touch threshold is specified.
Sample count for drift correction (Drift Correction Interval)	This function updates the baseline at a fixed cycle to follow changes in parasitic capacitance. The period for updating the baseline can be specified.
Debouncing count of touch-on filter (Positive Noise Filter Judgement Count)	This is a chattering-proof process to determine the touch ON. You can specify the duration for touch ON to be confirmed when the touch ON status continues for a certain number of times.
Debouncing count of touch-off filter (Negative Noise Filter Judgement Count)	This is a chattering-proof process to determine the touch OFF. You can specify the duration for touch OFF to be confirmed when the touch OFF status continues for a certain number of times.
Continuous Touch Cancel Count (Long Touch Cancel Judgement Count)	This function forcibly turns touch OFF after a certain period of time when something touches ON. You can specify the duration for the force-to-touch OFF function to be activated.

The following is a detailed explanation of the capacitance measurement function related to CapTouch parameters.

For the parameters not listed in Table 7-1, refer to QE for Capacitive Touch Help.

(1) Moving average

This filter is applied when calculating the measured value.

This parameter is the Average sample count (hereafter referred to as sample count) of the moving average filter. Decreasing the sample count increases the response speed at a finger contact. However, a lower value causes the measured count value to be more sensitive to noise, etc.

On the other hand, if the sample count increases, the resistance to noise, such as sudden changes in measured values, may be improved, but the response may be slowed down.

(2) Touch judgment

This function determines the touch ON/OFF of the self-capacitance button and mutual-capacitance button (keypad).

The following parameters are available.

- Touch threshold

Parameter used for judgment when the button/keypad switches from touch OFF to touch ON.

Reducing the touch threshold makes touch ON easier, while increasing the touch threshold makes touch ON less difficult.

- Hysteresis

This parameter is applied to the touch threshold when the button/keypad changes from touch ON to touch OFF. This parameter functions as a countermeasure against chattering and touch OFF is judged when the value falls below the touch threshold by the amount of hysteresis.

Increasing hysteresis will increase the effectiveness of the anti-chattering measures but be aware that touch ON will make it harder to become touch OFF and may result in poor button response.

- Debouncing count of touch-on filter (Positive noise filter)

This parameter sets the duration from when the measured value of the button/keypad exceeds the touch threshold to when it is judged to be a touch ON. This parameter acts as a anti-chattering measure and confirms the touch ON when the measured value exceeds the touch threshold for a specified period.

Increasing the parameter will increase the effectiveness of the chattering countermeasure but be aware that the button response may be degraded.

- Debouncing count of touch-off filter (Negative noise filter)

This parameter sets the duration from when the measured value of the button/keypad falls below the touch threshold to when it is judged to be a touch OFF. This parameter functions as a anti-chattering measure and confirms the touch OFF when the measured value falls below the touch threshold for a specified period.

Increasing the parameter will increase the effectiveness of the anti-chattering measures but be aware that the button response may be degraded.

(3) Drift correction

This function updates the baseline of the button/keypad according to the surrounding environment. This function works in touch OFF, averaging the measured value in touch OFF in a specified period, and updating the baseline with the averaged value.

There is a drift-correction interval as CapTouch parameter, and you can specify the duration for updating the baseline. Increasing this parameter slows the update of the baseline, while decreasing it accelerates the update of the baseline. Note that if you set it too small, it will immediately follow the measured value when you touch the button. Consequently, it will be difficult to ON the touch.

(4) Continuous Touch Cancel Count (Long Touch Cancel Judgement Count)

This function is used to turn the touch OFF when the touch ON period of the button/keypad exceeds a specified period. Due to sudden environmental changes such as strong noises, it may not be possible to recover from the touch ON status because the drift-correction process does not update the baseline. To recover from this state, touch OFF is forced when touch ON continues for a certain period, and drift-correction is activated.

Manual tuning can also be done by directly editing the program's source code, but here we'll describe manual tuning using QE for Capacitive Touch.

For manual tuning, use the "List of CapTouch Parameters" in QE for Capacitive Touch (inside the red frame in Figure 7.3). For example, to make the buttons more responsive, reduce the Touch Threshold value. Here, Figure 7.3 lowers the touch threshold from 3000 to 1000. The red line in Figure 7.3 represents the measured value, the blue line represents the baseline, and the green line represents the touch threshold. You can see that the measured value is judged as touch ON at the threshold of 1000, while it is not judged as touch ON at the threshold of 3000..

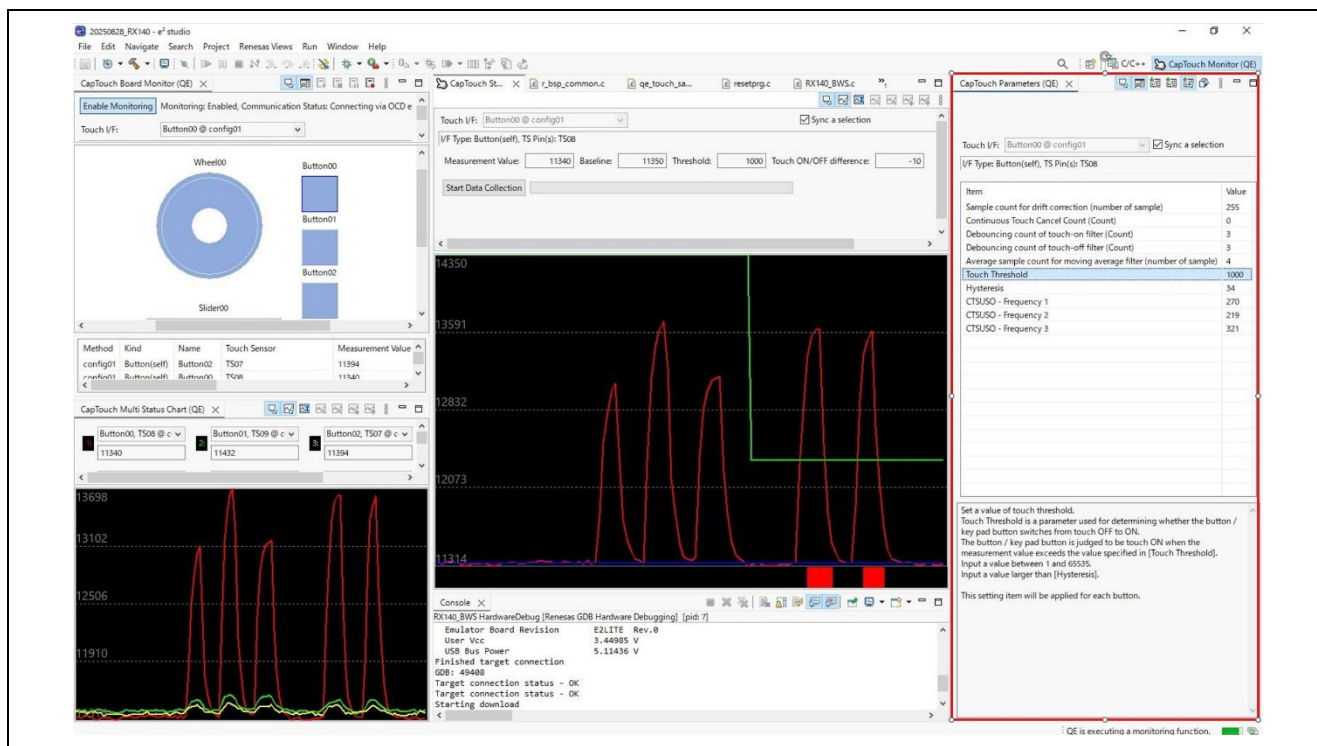


Figure 7.3 Manually Tuning with QE for Capacitive Touch

Please also refer to APN below for parameters.

[RL78 Family CTSU Module Software Integration System \(R11AN0484\)](#)

[RL78 Family TOUCH Module Software Integration System \(R11AN0485\)](#)

[RX Family QE CTSU module Firmware Integration Technology \(R01AN4469\)](#)

[RX Family QE Touch module Firmware Integration Technology \(R01AN4470\)](#)

[RA Flexible Software Package Documentation: CTSU \(r_ ctsu\)](#)

[RA Flexible Software Package Documentation: Touch \(rm_ touch\)](#)

7.3 Tuning with advanced mode

Advanced mode allows you to adjust individual parameters such as drive pulse frequency and measurement time. For details, please refer to the application note below.

[Capacitive Sensor MCU QE for Capacitive Touch Advanced mode Parameter Guide \(R30AN0428\)](#)

8. Development Environment

There are two types of development environments: "e² studio integrated development environment" and "QE for Capacitive Touch" published on our website, and "standalone version QE for Capacitive Touch" for customers who use CS+ and IAR integrated development environments.

For more information and to download "QE for Capacitive Touch", please refer to the introduction page of QE for Capacitive Touch, a development support tool for capacitive touch sensors, from the following URL.

Multiple built-in system development, tuning guide, and tutorial videos are available. A standalone version is also included in the same package.

[QE for Capacitive Touch: Development Assistance Tool for Capacitive Touch Sensors](#)

[QE for Capacitive Touch: Development Assistance Tool for Capacitive Touch Sensors Information for Users](#)

8.1 Capacitive Touch Sensor Development Support Tool “QE for Capacitive Touch”

The Capacitive Touch Sensor Development Support Tool QE for Capacitive Touch is a solution toolkit that runs on an integrated development environment e² studio. It supports the initial setting and sensitivity tuning of the touch interface required for the development of embedded systems that use a capacitive touch sensor, thus shortening the development period.

It can be developed with a simple GUI according to the workflow. Figure 8.1 shows the operation QE for Capacitive Touch.

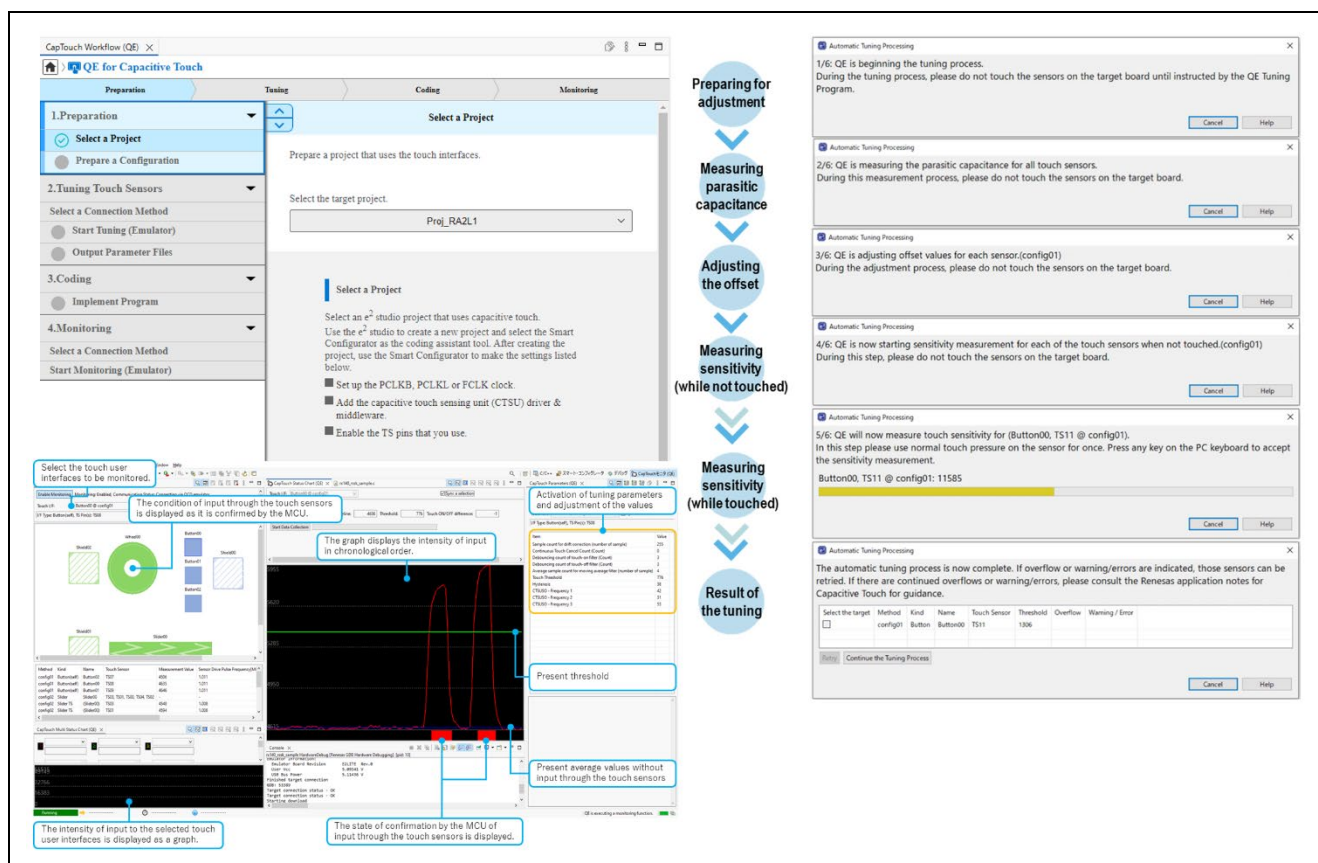


Figure 8.1 Operation display of QE for Capacitive Touch

8.2 Stand-alone Version of QE for Capacitive Touch

For customers who use CS + and IAR integrated development-environment, a stand-alone version of QE for Capacitive Touch (hereafter, a stand-alone version of QE) is available. The stand-alone version of QE acquires measurements through the serial port for tuning and monitoring functions. Figure 8.2 shows the operation of the stand-alone QE.

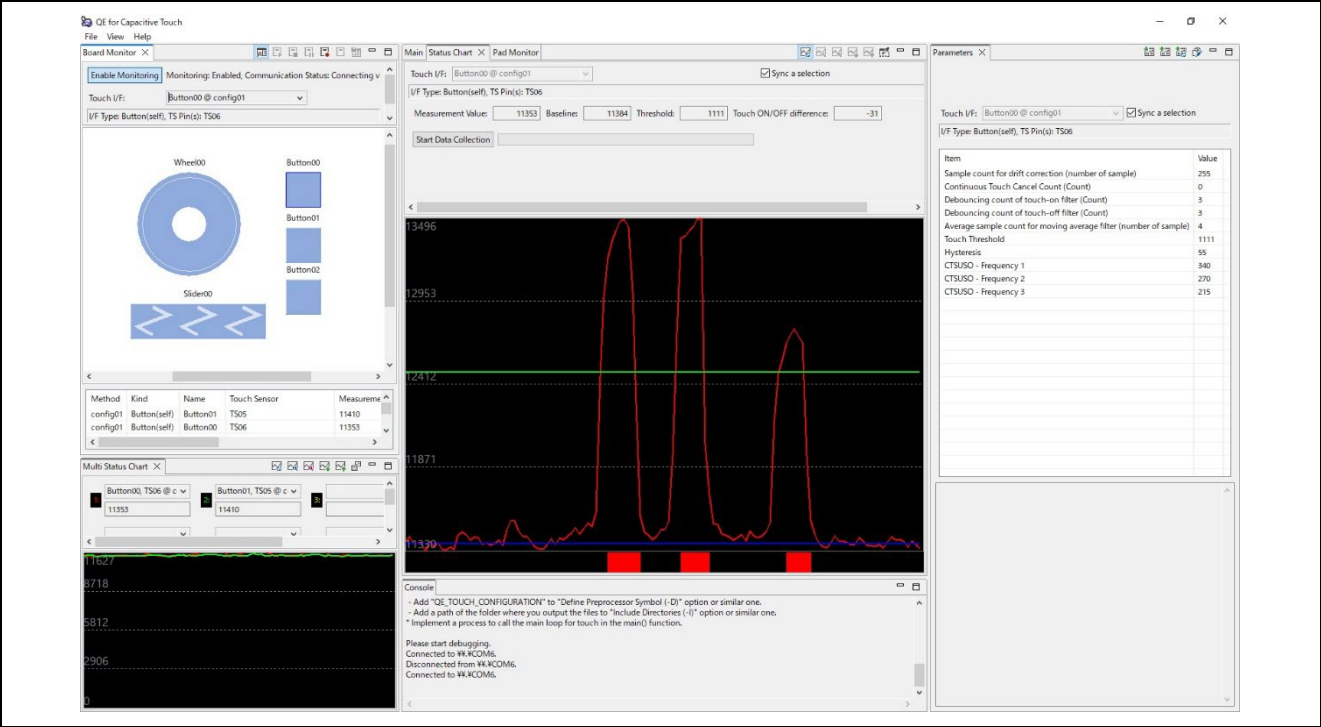


Figure 8.2 Stand-alone version of QE

8.3 Measurement of Standard Deviation and SNR

QE for Capacitive Touch has a function that measures the standard deviation and the SNR of the measured value. After the data collection and standard deviation calculations are completed for each of the Touch OFF and Touch conditions, the "Signal Value" and "SNR Value" are calculated.

When pressing the "Start Data Collection" button in the "CapTouch Status Chart (QE)" view, the "Data Collection Settings" dialog box is displayed. SNR measurement supports buttons only and does not support wheels or sliders. For details, refer to QE for Capacitive Touch Help. Figure 8-3 shows the Button Sensitivity (SNR) Derivation Method in VMM.

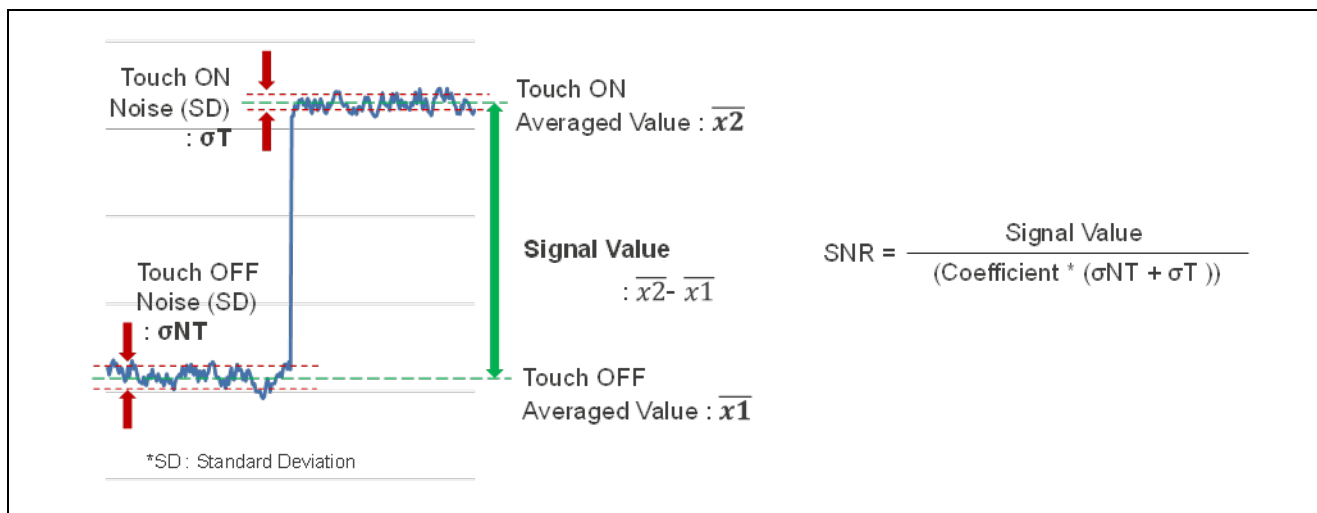


Figure 8.3 Button Sensitivity (SNR) Derivation Method in VMM

9. Evaluation kit

9.1 Capacitive Touch Evaluation System

The Capacitive Touch Evaluation System is a kit that can easily evaluate the touch key solutions provided by Renesas. Using the boards and software included in the kit, you can start the evaluation immediately after purchasing the kit.

Refer to the following for details.

<https://www.renesas.com/en/design-support/boards-kits>



Figure 9.1 Capacitive Touch Evaluation System

10. Other

10.1 Term

Term	Description
CCO	CCO (Current Control oscillator) refers to the current-controlled oscillator used by the capacitive touch sensor. Some application notes are referred to as ICO.
ICO	Same as CCO.
TSCAP	This capacitor is used to stabilize the internal-voltage of CTSU.
Damping resistance	This resistor is used to reduce the influence of terminal breakdown and noise from external noise. Refer to the Capacitive Touch Electrode Design Guide (R30AN0389) for more information.
VDC	VDC (Voltage Down Converter) is the power supply for the capacitance sensor built in CTSU.
Multi-clock measurement	This function measures using multiple sensor unit clocks with different frequencies. It refers to the multi-clock measurement function.

10.2 Q&A's

Questions	Description
Is it possible to switch between self-capacitance and mutual capacitance methods with 1 touch key MCU?	By using the self-capacitance and the mutual capacitance methods switching, it is possible to use both measurement methods with 1 MCU.
If the MCU is reset and started with a finger in contact with the electrode, can the contact state be detected?	The electrostatic capacitance of the finger in contact with the electrode is adjusted as parasitic capacitance by the offset tuning process by software and therefore cannot be detected.
Why does the measured value drop when I release my finger after executing the offset tuning process with my finger in contact with the Self-capacitance electrode?	The offset tuning process is executed with parasitic capacitance and finger capacitance added. In this case, when the finger is released, the capacitance decreases and the measured value drops.
Why does the button become unresponsive when I release my finger after executing the offset tuning process with my finger in contact with the Self-capacitance electrode?	The offset tuning process is executed with the finger capacitance added to the parasitic capacitance. In this case, when the finger is released, the capacitance decreases and the measured value drops, but the baseline remains the same. The button response will be poor until the baseline is tracked by the drift process.

10.3 Technical inquiries

Contact to Technical Support or search for a Knowledge Base common question (FAQ) or community forum.

<https://www.renesas.com/us/en/contact-us>

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Mar.31.23	-	First edition issued
1.10	Apr.04.23	20	Updated Figure 10.1
1.20	Dec.12.23	3	Updated contents of Section 1
		5	Updated contents of Section 2.1
		7	Added description to TSCAP.
		8	Updated contents of Section 2.2.2
		10	Updated contents of Section 2.3.1
		11	Updated contents of Section 2.3.2
		13	Updated contents of Table 3.1.
		16	Updated contents of Section 3.3.2
		18	Updated contents of Figure 3.5.
		20	Updated contents of Figure 4.1.
		21	Updated contents of Figure 4.2.
		24-27	Added RL78/G16 Group
		29	Updated contents of Table 4.11.
		30	Updated contents of Table 4.12.
		33	"Initial Offset Tuning" changed to "Offset Tuning". Updated LINK.
		34	Added contents of Software Filter.
		35	Updated contents of Tuning.
		40	Added contents of Software Filter.
		41	Added LINK.
		42	Updated contents and title of Section 7.3.
		44	Added contents of Q&A's.
1.30	Oct.22.24	5	Updated contents of Section 2.1
		6	Updated contents of Section 2.2
		7	Updated contents of Section 2.2.1
		8	Updated Figure 2.4
		9	Updated contents of Section 2.3
		11	Updated Figure 2.8
		12	Updated contents of Section 2.4
		13	Updated contents of Section 3
		13	Updated contents of Section 3.1
		14	Updated contents of Section 3.2.1
		15	Updated contents of Section 3.3.1
		16	Added contents of Section 3.3.1(a)
		16	Added contents of Section 3.3.1(b)
		17	Updated contents of Section 3.3.2
		18	Updated contents of Section 3.3.3
		18	Updated contents of Section 3.3.4
		18	Added contents of section 3.3.5
		19	Updated contents of Section 3.3.6
		19	Updated contents of Section 3.3.8
		20	Updated Figure 4.1 in Section 4.1
		28-30	Updated Table 4.10 to Table 4.12
		32	Updated contents of Section 6
		32	Updated contents of Section 6.1
		33	Updated contents of Section 6.1.1

		33	Added contents of Section 6.1.3
		34	Updated contents of Section 6.2
		34	Updated contents of Section 6.3
		34	Updated contents of Section 6.4
		35	Updated contents of Section 7
		37	Updated contents of Section 7.2
		40	Updated contents of Section 7.3
		41	Updated contents of Section 8
1.40	Feb.19.25	12	Updated contents of Section 2.4
		13	Updated contents of Section 3.1
		14	Updated contents of Section 3.2.1
		16	Updated contents of Section 3.3.1
		17	Updated contents of Section 3.3.2
		18	Updated contents of Section 3.3.3
		18	Updated contents of Section 3.3.4
		18	Updated contents of Section 3.3.5
		20	Updated Figure 4.1 in Section 4.1
		28-30	Updated Table 4.10 to Table 4.12
		36	Updated contents of Section 7.1
		37-39	Updated contents of Section 7.2
		44	Updated contents of Section 10.2
1.50	Sep.24.25	3	Updated contents of Section 1
		5	Updated contents of Section 2.1
		8	Updated contents of Section 2.2.2
		11	Updated contents of Section 2.3.2
		12	Updated contents of Section 2.4
		13	Updated contents of Section 3.1
		15 - 16	Updated contents of Section 3.3.1
		18	Updated contents of Section 3.3.3 to 3.3.5
		19	Deleted contents of Section 3.3.8
		20	Updated contents of Section 4.1
		23	Updated contents of Section 4.2.2
		27	Updated contents of Table 4.9
		28 - 31	Updated contents of Section 4.4
		33	Updated contents of Section 6.1
		34	Updated contents of Section 6.1.1
		35	Updated contents of Section 6.3
		37	Updated contents of Section 7.1
		38 - 40	Updated contents of Section 7.2
		42	Updated contents of Section 8 to 8.1
		43	Updated contents of Section 8.2
		44	Added contents of Section 8.3
		46	Updated contents of Section 10.2
1.60	Dec.23	35	Updated the link for Functional Safety in Section 6.2 to the APN.

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/.