RX261 Group

RX261 Capacitive Touch Evaluation System Sample Code

Introduction

This document describes the sample code for the RX261 Capacitive Touch Evaluation System

Target Device

RX261 (R5F52618BGFP)

Contents

1. Overview

This sample code is software that operates with capacitive touch in the RX261 Capacitive Touch Evaluation system.

The following is added to the project created by e² studio.

- Components generated by the smart configurator
- Capacitive touch configuration files and applications tuning with QE for Capacitive Touch (QE)
- LED control application

The functions are shown below.

1. Capacitive touch function operates all electrodes (3 buttons, slider, wheel) of Capacitive Touch Evaluation Application Board.
2. Press the capacitive touch buttons, slider and wheel to control the LEDs on Capacitive Touch Evaluation Application Board.
3. Enables USB serial interface to control serial communication and supports QE serial monitor and serial tuning. For more information on serial monitoring and serial tuning, refer to QE Help and "7. [Additional function] Serial communication monitor setting using UART" in Using QE and FIT to Develop Capacitive Touch Applications.
4. LED control is performed in conjunction with the push button on CPU board. Pressing SW1, LED1 lights up. Pressing SW2, LED2 lights up.

2. Environment for Confirming Operation

The operation of this sample code has been confirmed the following environment.

Table 2-1 Operating Environment

Item	Description
MCU	RX261 (R5F52618BGFP)
Operating frequency	HOCO 64MHz
Operating voltage	5.0V
Evaluation board	RX261 Capacitive Touch Evaluation System (Product No：RTK0EG0055S01001BJ) • RX261 CPU Board（Product No：RTK0EG0054C01001BJ） • Capacitive Touch Evaluation Application Board — Self-Capacitance Buttons / Wheel / Slider Board (Product No：RTK0EG0019B01002BJ)
Integrated development environment	e² studio Version 2024-07 (24.07.0)
C Compiler	CC-RX V3.06.00
Development Assistance Tool for Capacitive Touch Sensors	QE for Capacitive Touch V4.00
Emulator	Renesas E2 Emulator Lite

3. Software specification

3.1 Software structure diagram

Figure 3-1 shows the software structure diagram of this sample code.

This software uses components generated by the smart configurator.

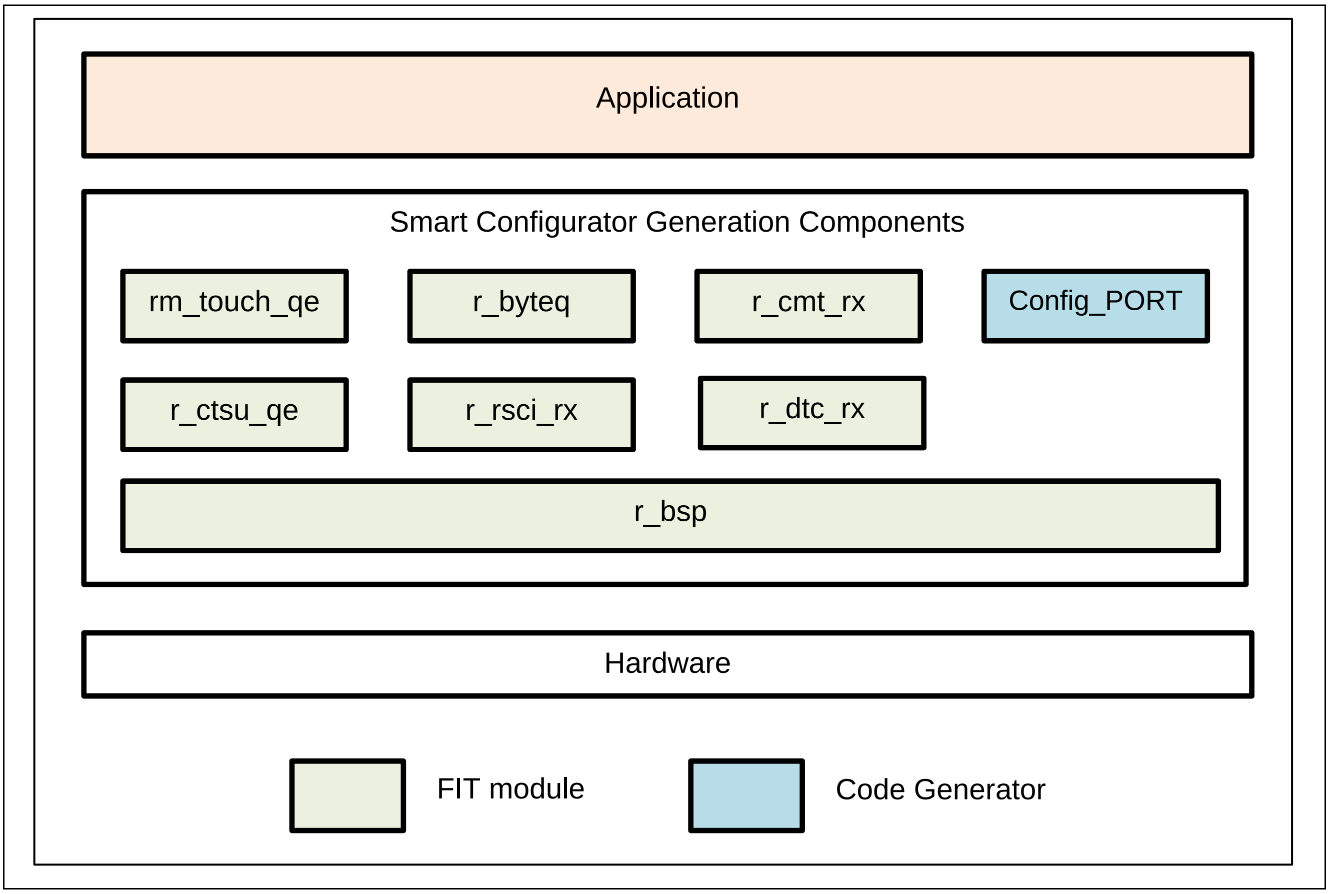

Figure 3-1 Software structure diagram

Table 3-1 shows a list of components and versions. Refer to the smart configurator for component settings.

Table 3-1 Components and versions list

Component	Version	Configuration
Board Support Packages. (r_bsp)	7.51	r_bsp(used)
Byte-based circular buffer library. (r_byteq)	2.10	r_byteq(used)
CMT driver (r_cmt_rx)	5.70	r_cmt_rx(used)
CTSU QE API (r_ctsu_qe)	3.00	r_ctsu_qe(used)
DTC driver (r_dtc_rx)	4.50	r_dtc_rx(used)
Ports	2.4.1	Config_PORT(PORT: used)
RSCI Driver (r_rsci_rx)	2.60	r_rsci_rx(used)
Touch QE API (rm_touch_qe)	3.00	rm_touch_qe(used)

3.2 File structure

This is the file structure of this sample code. The project configuration file and smart configurator generation file of the development environment are omitted.

```
Rx671_rssk_sample
 |
 ├─QE-Touch
 |     rx261_rssk_sample_log_tuning.log      ・・・QE Tuning log
 |     rx261_rssk_sample.tifcfg              ・・・Touch interface configuration file
 |
 ├─qe_gen
 |     qe_touch_config.c                     ・・・Touch configuration source
 |     qe_touch_config.h                     ・・・Touch configuration header
 |     qe_touch_define.h                     ・・・Touch define header
 |     qe_touch_sample.c                     ・・・Touch sample application
 |     qe_touch_sample.h                     ・・・Touch sample application header
 |
 └─src
    | rx261_rssk_sample.c                    ・・・Main file
    | r_rssk_switch_led.c                    ・・・Switch & LED function source
    | r_rssk_switch_led.h                    ・・・Switch & LED function header
    | r_rssk_touch_led.c                     ・・・Touch electrode LED function source
    | r_rssk_touch_led.h                     ・・・Touch electrode LED function header
    |
    └─smc_gen
       ├─Config_PORT                         ・・・PORT Driver folder
       ├─general                             ・・・general setting folde
       ├─rm_touch_qe                         ・・・TOUCH FIT folder
       ├─r_bsp                               ・・・BSP folder
       ├─r_byteq                             ・・・BYTEQ FIT folder
       ├─r_cmt_rx                            ・・・CMT FIT folder
       ├─r_config                            ・・・FIT configuration folder
       ├─r_dtc_rx                            ・・・DTC FIT folder
       ├─r_ctsu_qe                           ・・・CTSU FIT folder
       ├─r_pincfg                            ・・・PIN configuration folder
       └─r_rsci_rx                           ・・・RSCI FIT folder
```

3.3 Constants

Table 3-2 lists the constants.

Table 3-2 List of Constant

Constant Name	Value	Description
File Name : qe_touch_sample.c		
TOUCH_SCAN_INTERVAL_EXAMPLE	(20)	Software delay value [unit : msec]
File Name : r_rssk_switch_led.c		
RSSK_SW1_PORT	(PORTE.PIDR.BIT.B6)	Pointer to port control register connected to SW1
RSSK_SW2_PORT	(PORTE.PIDR.BIT.B5)	Pointer to port control register connected to SW2
RSSK_LED1_PORT	(PORT2.PODR.BIT.B5)	Pointer to port control register connected to LED1
RSSK_LED2_PORT	(PORT2.PODR.BIT.B4)	Pointer to port control register connected to LED2
SW_EDGE_RIZE	(0x07U)	Switch rising judgment
SW_EDGE_FALL	(0x08U)	Switch falling judgment
SW_EDGE_BIT_MASK	(0x0FU)	Switch state judgement mask
RSSK_LED_ON	(0x01U)	Turn on the LED
RSSK_LED_OFF	(0x00U)	Turn off the LED
File Name : r_rssk_touch_led.c		
LED_COL0	(PORTE.PODR.BIT.B0)	Pointer to port control register connected to COL0
LED_COL1	(PORTE.PODR.BIT.B1)	Pointer to port control register connected to COL1
LED_COL2	(PORTA.PODR.BIT.B7)	Pointer to port control register connected to COL2
LED_COL3	(PORT5.PODR.BIT.B0)	Pointer to port control register connected to COL3
LED_ROW0	(PORT1.PODR.BIT.B2)	Pointer to port control register connected to ROW0
LED_ROW1	(PORT1.PODR.BIT.B3)	Pointer to port control register connected to ROW1
LED_ROW2	(PORT5.PODR.BIT.B1)	Pointer to port control register connected to ROW2
LED_ROW3	(PORT5.PODR.BIT.B2)	Pointer to port control register connected to ROW3
LED_COL_MAX	(4)	Number of COL signals
LED_COL_ON	(0x01U)	COL signal ON
LED_COL_OFF	(0x00U)	COL signal OFF
LED_ROW_OFF	(0x01U)	ROW signal OFF
SLIDER_LED_NUM	(5U)	Number of slider LED
SLIDER_RESOLUTION	(100U)	Maximum slider touch result
WHEEL_LED_NUM	(8U)	Number of wheel LED
WHEEL_LED_MSB	(1U << (WHEEL_LED_NUM - 1))	Wheel LED control bit MSB
WHEEL_RESOLUTION_DEGREE	(360U)	Maximum wheel touch result
WHEEL_POSITION_OFFSET_DEGREE	(112)	Wheel touch position offset [unit : degree]
ALL_LED_NUM	(16U)	Total number of touch electrode board LEDs
LED_TEST_INTERVAL	(100U)	LED lighting interval time

3.4 Enumerations

Table 3-3 lists the rssk_sw_status_t enum

Table 3-3 rssk_sw_status_t

Member	Value	Description
RSSK_SW_OFF	0x00	Switch OFF state
RSSK_SW_ON	0x01	Switch ON state

3.5 Variables

Table 3-4 lists the global valiables

Table 3-4 List of Global Valiables

Variable Name	Types	Description
File Name : qe_touch_sample.c		
gs_cmt_ch	uint32_t	Use channel of CMT
File Name : r_rssk_switch_led.c		
rssk_get_sw1_status	uint8_t	State of switch SW1
rssk_get_sw2_status	uint8_t	State of switch SW2
File Name : r_rssk_touch_led.c		
g_led_drive_colmun	uint8_t	Touch electrode board LED drive information
g_button_idx[]	uint8_t	Button index array

3.6 Functions

Table 3-5 lists the functions

Table 3-5 List of Functions

Function Name	Description
qe_touch_sample.c	
qe_touch_main	Main function
r_rssk_initialize	Initialization processing of Capacitive Touch Evaluation System
r_rssk_led_test	LED test processing for Capacitive Touch Evaluation System
r_rssk_timer_callback	CMT interrupt callback
r_rssk_switch_led.c	
r_rssk_switch_led_init	CPU board LED initialization processing
r_rssk_switch_led_control	CPU board LED control processing
r_rssk_led1_on	CPU board LED1 lights up
r_rssk_led1_off	CPU board LED1 lights off
r_rssk_led2_on	CPU board LED2 lights up
r_rssk_led2_off	CPU board LED2 lights off
r_rssk_touch_led.c	
r_rssk_touch_led_test	Touch electrode board LED test processing
r_rssk_touch_led_control	Touch electrode board LED control processing

3.7 List of Peripheral Functions Used and Pins Used

Table 3-6 shows a list of peripheral functions used, Table 3-7 shows a list of used pins, and Table 3-8 shows a list of handling of unused pins in this sample software.

Table 3-6 List of Peripheral Functions Used

Peripheral Function	Usage
CTSU, DTC	CTSU measurement
RSCI0	QE serial monitoring and serial tuning
CMT	LED control trigger
PORT	LED control

Table 3-7 List of used pins

Pin No.	Pin Name	I/O	Usage
18	TS00	I/O	CTSU measurement
19	TS01	I/O	
20	TS02	I/O	
21	TS03	I/O	
22	TS04	I/O	
31	TS05	I/O	
32	TS06	I/O	
39	TS11	I/O	
40	TS12	I/O	
45	TS13	I/O	
47	TS15	I/O	
65	TS27	I/O	
68	TS30	I/O	
74	TS33	I/O	
76	TS35	I/O	
48	TSCAP	I/O	
27	P21/RXD0	I	QE serial monitoring
28	P20/TXD0	O	
72	PE6/SW1	I	Switch Input
73	PE5/SW2	I	
23	P25/LED1	O	LED control
24	P24/LED2	O	
78	PE0/LED_COL0	O	
77	PE1/LED_COL1	O	
63	PA7/LED_COL2	O	
44	P50/LED_COL3	O	
34	P12/LED_ROW0	O	
33	P13/LED_ROW1	O	
43	P51/LED_ROW2	O	
42	P52/LED_ROW3	O	

Table 3-8 List of Handling of Unused Pins

Pin No.	Pin Name	I/O	Handling
5	VCL	I	Connect the pin to GND via a capacitor (4.7uF).
7	MD/FINED	I/O	Connect the pin to VCC or GND via a register (4.7Kohm).
8	XCIN	-	Open
9	XCOUT	-	Open
10	RES#	I	Connect the pin to reset citcuit
12 / 62	VSS	I	Connect the pin to GND.
14 / 60	VCC	I	Connect the pin to VCC.
15	UPSEL	I	Connect the pin to VCC via a register (10Kohm).
36	-	-	Open
37	-	-	Open
45	UB/TS13	I	Connect the pin to GND via a register (4.7Kohm) or TS13.
97	AVCC0	I	Connect the pin to GND via a capacitor (0.1uF).
99	AVSS0	I	Connect the pin to GND.
Pins than the above		-	Low output

The peripheral function settings using Smart Configurator are shown below.

- CTSU, DTC (CTSU measurement)

CTSU is used for touch measurement. DTC is used to acquire CTSU register settings and measurement results.

Table 3-9 and Table 3-10 shows the settings of each peripheral function.

Table 3-9 CTSU Setting

Item	Setting
Data transfer by interrupt	DTC

Table 3-10 DTC Setting

Item	Setting
DTCER control	Clear all DTCER registers in R_DTC_Open
Address mode	Full address mode
Transfer Data Read skip	Enable transfer data read skip
DMAC FIT check	DMAC FIT module is not used with DTC FIT module
Sequence transfer	Sequence transfer not used

Table 3-11 BSP Setting

Item	Setting
Heap size	0x1000

*If setting "Data transfer of INTCTSUWR and INTCTSURD" to "DTC" in the CTSU settings, need to increase the "Heap size" in the BSP settings.

- RSCI0

Use RSCI0 for serial monitoring of QE for Capacitive Touch. Table 3-12 shows the RSCI0 settings.

Table 3-12 RSCI0 Setting

Item	Setting
Mode	ASYNC mode
Channel	Channel 0
Transmit end interrupt	Enable
ASYNC mode TX queue buffer size for channel 0	800

3.8 Processing Flowchart

Figure 3-2 shows processing flowchart of this sample code.

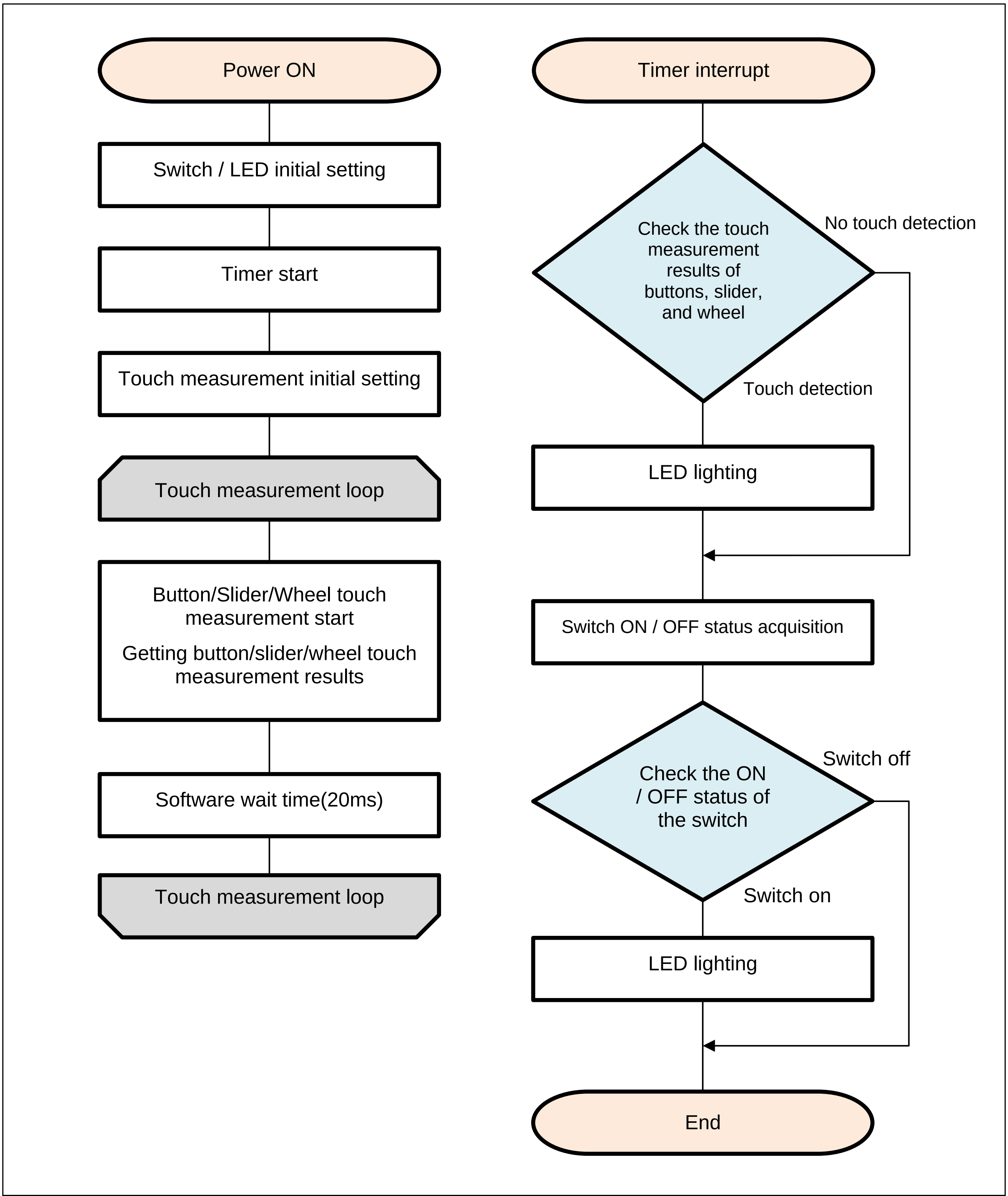

Figure 3-2 Processing Flowchart (Self-Capacitance Buttons / Wheel / Slider Board)

4. Capacitive Touch Setting

These are the touch interface configuration, configuration (method) settings and tuning results of this sample code. These use the tuning function of QE.

When tuning the TS13 (TS-B2) in the RX261 Group Capacitive Touch Evaluation System, use the tuning function via the serial connection to QE. For details, refer to the User's Manual of the RX261 Group Capacitive Touch Evaluation System "2.3.5 TS13 Usage Precaution".

4.1 Touch Interface Configuration

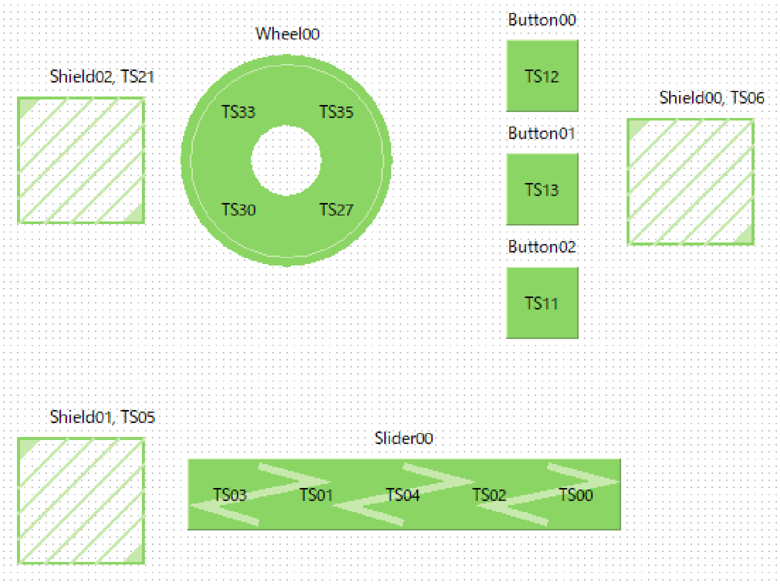

Figure 4-1 Touch interface configuration (Self-Capacitance Buttons / Wheel / Slider Board)

4.2 Configuration (methods) Settings

config01 sets 3 buttons, Shield(TS6).

config02 sets Slider, Shield(TS5).

config03 sets Wheel, Shield(TS21).

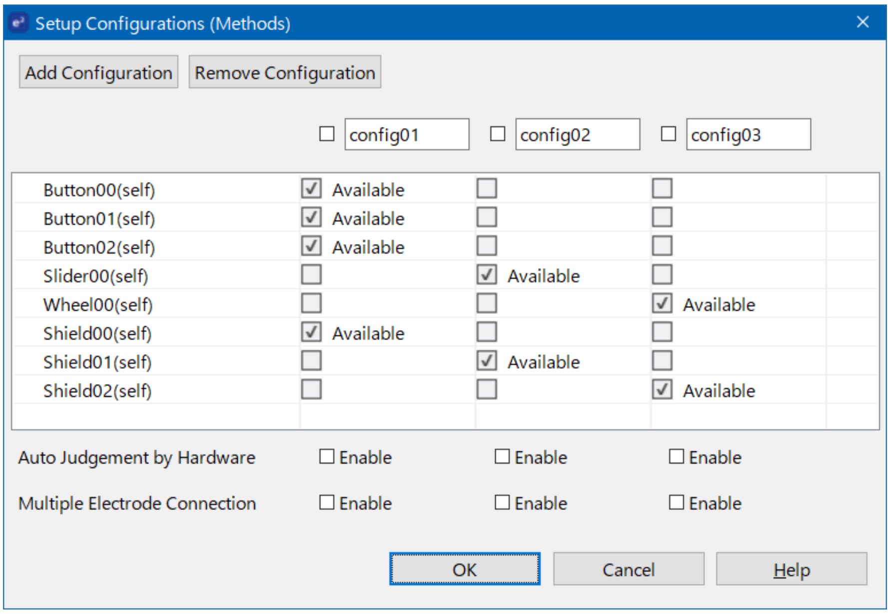

Figure 4-2 Configuration (methods) setting

4.3 Tuning Results

Table 4-1 shows tuning results in QE tuning. Sample code operates with the setting values shown in the QE tuning result list.

Since the values in QE tuning result list depend on the operating environment at QE tuning, these values may change at QE tuning again.

Table 4-1 QE tuning result list (Self-Capacitance Buttons / Wheel / Slider Board)

methods	Button name	Touch senser	Parasitic capacitance [pF]	Drive pulse frequency [MHz]	Threshold	Scan time [ms]	so	snum	sdpa
config01	Button00	TS12	8.972	1	853	0.576	0x037	0x07	0x0B
config01	Button01	TS13	14.861	1	839	0.576	0x072	0x07	0x0B
config01	Button02	TS11	8.278	1	737	0.576	0x030	0x07	0x0B
config01	Shield00	TS06	44.132	1	-	0.576	-	-	-
config02	Slider00	TS03	8.236	1	762	0.576	0x02F	0x07	0x0B
config02	Slider00	TS01	7.556	1	762	0.576	0x029	0x07	0x0B
config02	Slider00	TS04	9.514	1	762	0.576	0x03C	0x07	0x0B
config02	Slider00	TS02	8.132	1	762	0.576	0x02E	0x07	0x0B
config02	Slider00	TS00	9.0	1	762	0.576	0x038	0x07	0x0B
config02	Shield01	TS05	45.16	1	-	0.576	-	-	-
config03	Wheel00	TS35	14.833	1	912	0.576	0x05C	0x07	0x0B
config03	Wheel00	TS33	12.715	1	912	0.576	0x071	0x07	0x0B
config03	Wheel00	TS30	9.042	1	912	0.576	0x037	0x07	0x0B
config03	Wheel00	TS27	9.042	1	912	0.576	0x037	0x07	0x0B
config03	Shield02	TS21	42.583	1	-	-	-	-	-

so : Variables for sensor offset settings
snum : Variables for setting the measurement period
sdpa : Clock division setting variable

4.4 Sensitivity adjustment

Button sensitivity adjustment uses QE for Capacitive Touch. The sensitivity adjustment method is as follows.

- The method using monitoring function of QE for Capacitive Touch

  Follow the tutorial from the QE for Capacitive Touch main window (Cap Touch Main).

- Real-time change method using monitoring function of QE for Capacitive Touch

  Display the Cap Touch parameter list of QE for Capacitive Touch and adjust it by the following steps.

1. Select the touch I/F corresponding to the button you want to adjust.
2. Click [Enable Monitoring] icon to start monitoring.
3. When the item is displayed, change the value of [Touch Threshold].
4. Click [Enable Auto Writing] to change the touch threshold.
5. Repeat steps 3 to 4 to adjust the sensitivity.

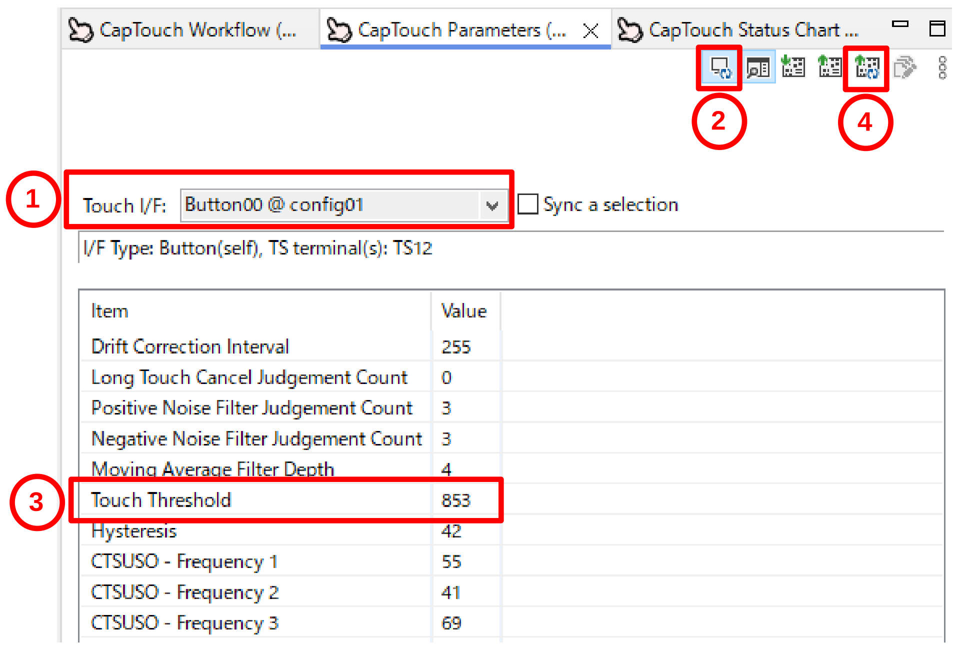

Figure 4-3 List of Parameters

- How to change the code manually

  It can be adjusted by changing the member variable of the structure variable "g_qe_touch_button_cfg_config01" in qe_touch_config.c.

  The variables to change are:

  ・ threshold : Touch detection threshold

5. Support

For information on capacitive touch, download tools and documentation, and technical support, please visit the website below

RX261 Capacitive Touch Evaluation System renesas.com/rssk-touch-rx261

RX Family Using QE and FIT to Develop Capacitive Touch Applications

renesas.com/document/apn/rx-family-using-qe-and-fit-develop-capacitive-touch-applications?language=en

QE for Capacitive Touch renesas.com/qe-capacitive-touch

Renesas support renesas.com/support

Revision History

Rev.	Date	Description	
		Page	Summary
1.0	Oct.22.2024	-	First edition issued

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.

Corporate Headquarters

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

Trademarks

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/.