RL78/G22

Capacitive Touch Low Power Guide (SNOOZE function)

Introduction

This application note describes the method for implementing low power consumption operation in touch measurement by the Capacitive Sensing Unit (CTSU2La) by using RL78/G22. This method uses the combination of intermittent operations of the CPU using the 32-bit interval timer function (TML32) and the SNOOZE function of the CTSU2La to achieve low power consumption in touch measurement.

Target Device

RL78/G22

When implementing the touch application, conduct an extensive evaluation of the touch function according to the environment.

Using the Low Power Consumption Application Notes

There are two methods for implementing low power consumption operation in touch measurement by the Capacitive Sensing Unit (CTSU2La) by using RL78/G22. If you want to implement low power consumption operation for MCU internal circuits only, refer to the method of using the SNOOZE function described in this application note.

The other method is to use the SNOOZE Mode Sequencer (SMS) in addition to the SNOOZE function. Note, however, that use of the SMS function can further reduce power consumption, but this method requires two pins to be shorted outside the MCU.
For details on the setup procedure, refer to the following application notes.

〇Sample code for concurrent use of the SNOOZE function and the SMS function
・RL78/G22 Capacitive Touch Low Power Guide (SMS / MEC function) [R01AN6847]
(The latest versions of the documents are available on the Renesas Electronics Website.)

〇Setup method for concurrent use of the SNOOZE function and the SMS function
・RL78 Family Capacitive Touch Low Power Application Development using SMS [R01AN7261]
(The latest versions of the documents are available on the Renesas Electronics Website.)

Contents

1. Outline

This application note describes the method for achieving low power consumption by using the SNOOZE function of RL78/G22 to perform intermittent operations in electrostatic capacitance touch measurements. This application note also shows the reference current consumption when touch measurement is performed with 100 ms cycle. In this application note, low power consumption to approximately 1/220 is implemented, compared to a case where touch measurement is performed by the CPU without using intermittent operations.

[Remarks] The SNOOZE function is a function that allows peripherals to be started by external triggers (used as activation sources) in STOP mode without starting the CPU.

2. Touch Measurement Mechanism

2.1 Operation Flow of Low-Power Consumption Touch Measurement (SNOOZE Function)

This section provides a summary of linked operation of peripherals and describes how power consumption is reduced when intermittent operation is performed in touch measurement by using the SNOOZE function.

Figure 2-1 shows an operation image of touch measurement with low power consumption (SNOOZE function).

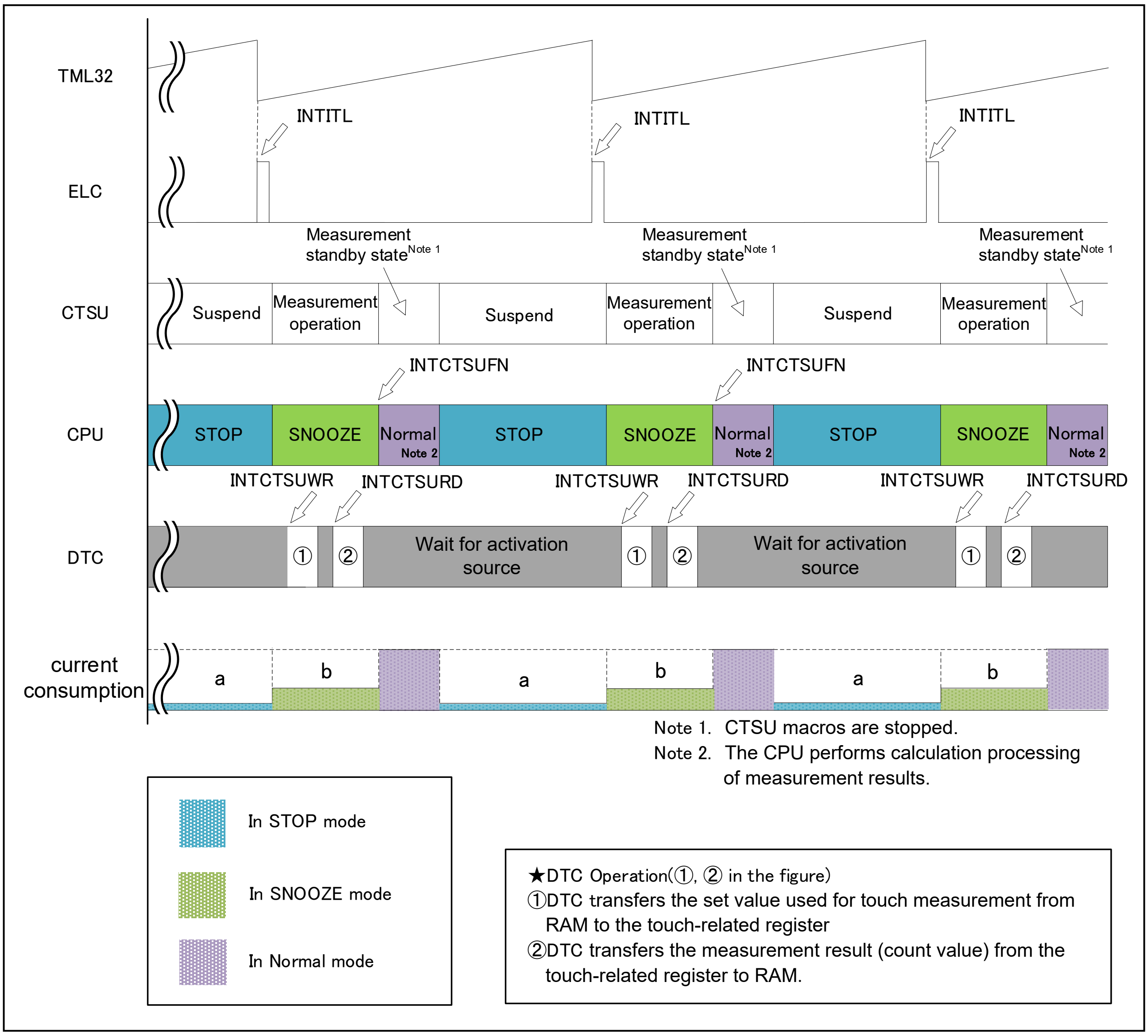


Figure 2-1 Operation image of touch measurement with low power consumption (SNOOZE function)

〇Overview of operation

1. When an interrupt source signal (INTITL) is generated at the touch measurement intervals set in TML32, the suspended (waiting for an external trigger) CTSU starts. At this time, the CPU status transitions from STOP mode to SNOOZE mode.
2. When the CTSU generates a write request interrupt for setting registers for each channel (INTCTSUWR), the set value used for touch measurement is transferred from RAM to the touch-related register via the DTC (① in the figure).
3. The CTSU starts touch measurement operation based on the data transferred to the touch-related register.
4. When the CTSU generates a measurement data transfer request interrupt (INTCTSURD), the measurement result (count value) is transferred from the touch-related register to RAM via the DTC (② in the figure).
5. When the CTSU ends measurement, it generates a CTSU measurement end interrupt (INTCTSUFN) and the CTSU enters the touch measurement standby state. At this time, the CPU status transitions from SNOOZE mode to Normal mode.
6. Software performs touch measurement end processing, and then performs touch on/off judgment processing based on the read measurement result.
7. When the STOP instruction is executed, the CPU status transitions from SNOOZE mode to STOP mode. Then, the CTSU is suspended and waits for the next interrupt source signal (INTITL).
8. Steps 1 to 7 are repeated.

[Remarks] Steps 2 and 3 are repeated according to the number of measurement channels of the CTSU.

〇Effects of current consumption reduction

・During non-touch measurement, stopping the CPU operation by transitioning the CPU to STOP mode can reduce current consumption in section “a” in the figure as compared to normal operation mode.

・Performing touch measurement processing in SNOOZE mode without starting the CPU can reduce current consumption in section “b” in the figure as compared to touch measurement processing performed in normal operation mode.

2.2 Peripherals Implementing Low Power Consumption Touch Measurement (SNOOZE Function)

Table 2-1 lists the peripherals used for performing intermittent operations in touch measurement that uses the SNOOZE function.

Table 2-1 Peripherals in the sample code

Used Peripherals	Functions
Capacitive Sensing Unit (CTSU2La)	Measures electrostatic capacitance of the touch sensor. Specifies the following settings to perform low power consumption operation. ・Select an external trigger ・Enable the SNOOZE function
Data transfer controller (DTC)	Transfers the set value used for touch measurement from RAM to the touch-related register. After touch measurement ends, the DTC transfers the measurement result (count value) from the touch-related register to RAM.
32-bit interval timer (TML32)	Timer to count the touch measurement cycles. This timer generates an interrupt source signal (INTITL) every touch measurement cycle (100 ms).
Event link controller (ELC)	Connects the TML32 interrupt signal source (INTITL) to the CTSU measurement trigger.

3. Operation Check Environment / Conditions

Table 3-1 lists the operation check environment. **Table 3-2** lists the operation check conditions.

Table 3-1 Operation check environment

Item	Description
Microcontroller used	RL78/G22 (R7F102GGE2DFB)
Operating frequency	• Main system clock High-speed on-chip oscillator (f_{IH}) : 32 MHz • CPU/peripheral hardware clock (f_{CLK}) : 32 MHz • Subsystem clock Low-speed on-chip oscillator (f_{IL}) : 32.768 kHz Low-speed peripheral clock frequency (f_{SXP}) : 32.768 kHz
Operating voltage	5.0 V LVD0 detection voltage : Reset mode At rising edge : TYP. 2.67 V (2.59 V to 2.75 V) At falling edge : TYP. 2.62 V (2.54 V to 2.70 V)
Target board	RL78/G22 Capacitive Touch Evaluation System (Product model : RTK0EG0042S01001BJ)
Integrated Development Environment (e^2 studio)	e^2 studio (2024-07) from Renesas Electronics Corp.
Smart Configurator	V24.7.0 from Renesas Electronics Corp.
C compiler (e^2 studio)	CC-RL V1.14.00 from Renesas Electronics Corp.
QE for Capacitive Touch	V4.0.0 from Renesas Electronics Corp.

Table 3-2 Operation check conditions

Item	Description
Touch measure cycle	100 ms
Sensor drive pulse frequency	2.0 MHz
Touch Sensor (TS pin)	TS28
Judgement Type	Value Majority Mode (Value Majority Mode : VMM)
Measurement mode	Self-capacitance method (MD1 bit of CTSUCRAL register = 0)
Scan mode	Multi-scan mode (MD0 bit of CTSUCRAL register = 1)
Measurement start trigger	External trigger (ELC) (CAP bit of CTSUCRAL register = 1)
Enables/Disables the SNOOZE function	Enables the SNOOZE function (SNZ bit of CTSUCRAL register = 1)
Boost power	Boost power on (PUMPON bit of CTSUCRAL register = 1)
Measurement power-supply	Measurement power-supply voltage = 1.5 V (ATUNE0 bit of CTSUCRAL register = 0)
Measurement power-supply current adjustment	40 μA (ATUNE1 bit of CTSUCRAL register = 1 ATUNE2 bit of CTSUCRAH register = 0)
Non-measured pin output select	Low-level output (POSEL[1:0] bits of CTSUCRAH register = 00b)
Sensor drive pulse select	Sensor unit clock (SUCLK) mode (SDPSEL bit of CTSUCRAH register = 1, FCMODE bit of CTSUCRAH register = 1)
Sensor stabilization wait time	32 cycles (SST[7:0] bits CTSUCRBL register = 0x1f)
Measurement count setting	7 (SNUM[7:6] bits CTSUSO1 register = 00b, SNUM[5:0] bits CTSUSO0 register = 00111b)

4. Hardware Descriptions

4.1 Example of Hardware Configuration

Figure 4-1 shows an example of the hardware configuration used in the application note.

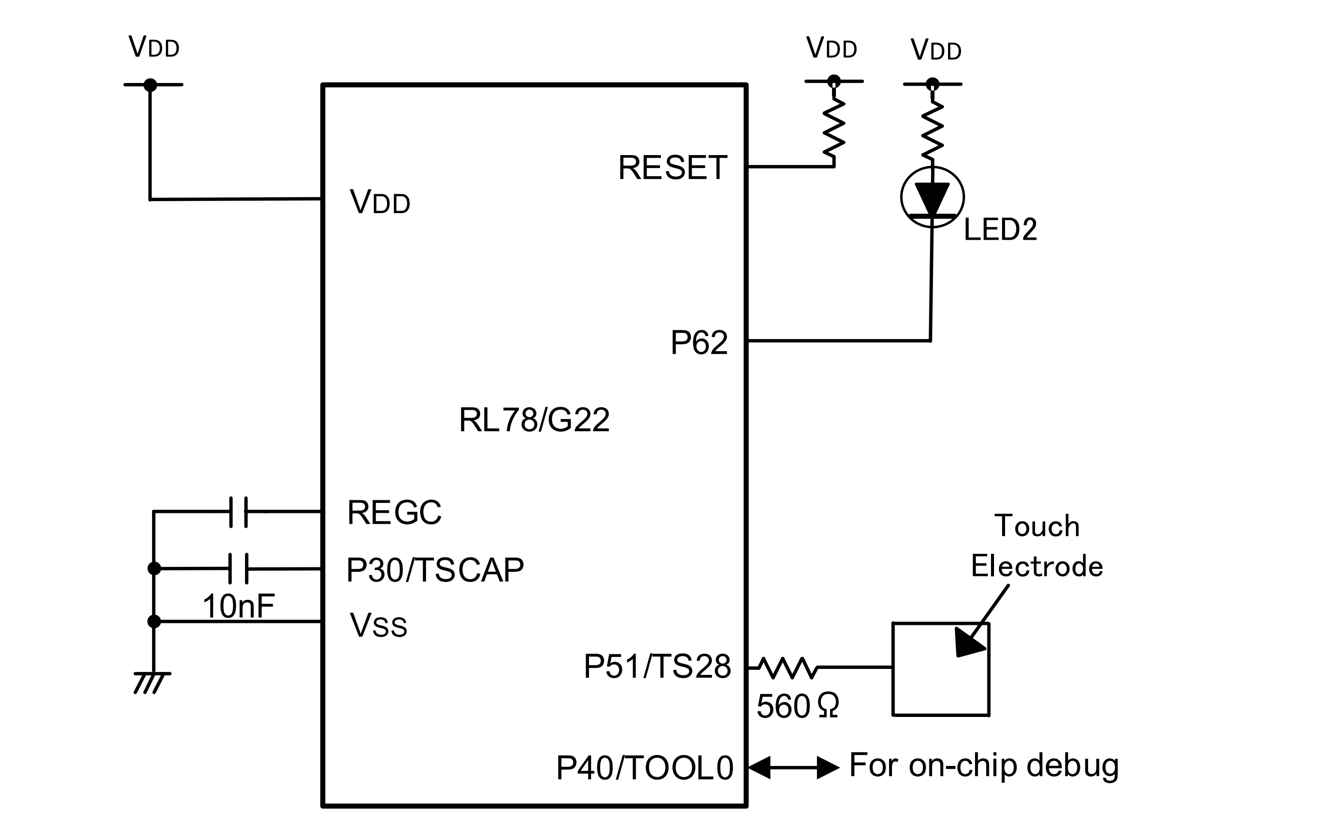


Figure 4-1 Hardware Configuration

Caution 1. This simplified circuit diagram was created to show an overview of connections only. When actually designing your circuit, make sure the design includes appropriate pin handling and meets electrical characteristic requirements (connect each input-only port to VDD or VSS through a resistor).

Caution 2. VDD must not be lower than the reset release voltage (VLVD0) that is specified for the LVD0.

4.2 List of Pins to be Used

Table 4-1 lists the pins to be used and their functions.

Table 4-1 Pins to be Used and Their Functions

Pin name	I/O	Function
P51/TS28	I/O	Pins for electrostatic capacitance measurement
P62	Output	LED2 control pin
P30/TSCAP	—	Connection pin to capacitor for measurement secondary power

Caution In this application note, only the used pins are processed. When actually designing your circuit, make sure the design includes sufficient pin processing and meets electrical characteristic requirements.

5. Software Description

5.1 Sample Code Outline of Operation

The sample code operates as follows by using the code generated by Smart Configurator shown in **Figure 5-1**.

1. After reset release by power-on, RM_TOUCH_Open function is executed to initialize CTSU.
2. Set the ELC event source to 32-bit interval timer (TML32) channel 0 compare match (INTITL0). Set the event link destination peripheral to the Capacitive Sensing Unit (CTSU).
3. When the touch measurement setting and SNOOZE function are enabled by executing the RM_TOUCH_ScanStart function[Note], the CTSU is suspended and waiting for an external trigger.
4. Start the TML32 timer (measurement cycle:100 ms).
5. CPU status transitions to STOP mode by executing the STOP instruction.
6. When the TML32 interrupt source (INTITL0) is generated, the Capacitive Sensing Unit (CTSU) is started by an external trigger from the ELC. Then, the CPU status transitions to SNOOZE mode and the CTSU starts touch measurement.
7. When the CTSU ends measurement, it generates a measurement end interrupt (INTCTSUFN) and enters the measurement standby state. Then, the CPU status transitions to Normal (normal operation) mode.
8. Get the measurement results by executing the RM_Touch_DataGet function. Turn on the LED2 when touch-on is detected, or turn off the LED2 when touch-off is detected.
9. Repeat from step 5 to step 8.

Note In the processing of the RM_TOUCH_ScanStart function, the CAP bit of the CTSUCRAL register is set to 1, the SNZ bit is set to 1, and then the STRT bit is set to 1. For details on setting conditions, refer to the description of the SNZ bit in section 27.2.3 CTSU control registers AL and AH (CTSUCRAL, CTSUCRAH) in the RL78/G22 User's Manual: Hardware (R01UH0978).

5.2 Smart Configurator Settings

Figure 5-1 shows the Smart Configurator components used by the sample code. Add the following components from the [Components] tab of Smart Configurator.

Component	Version	Configuration
Board Support Packages. - v1.70 (r_bsp)	1.70	r_bsp(used)
Capacitive Sensing Unit driver. (r_ctsu)	2.00	r_ctsu(used)
Event Link Controller	1.3.0	Config_ELC(ELC: used)
Interval Timer	1.5.0	Config_ITL000(ITL000: used)
Ports	1.5.0	Config_PORT(PORT: used)
Touch middleware. (rm_touch)	2.00	rm_touch(used)
Voltage Detector	1.4.0	Config_LVD0(LVD0: used)

Figure 5-1 Components of Smart Configurator

5.2.1 Component Settings

Table 5-1 describes the settings of each component in the sample code.

Table 5-1 Settings for each component

Component	Configuration	Settings
Capacitive Sensing Unit driver	r_ctsu	Set the DTC for data transfer when an interrupt source (INTCTSUWR or INTCTSURD) is generated. Specify to use all TSCAP pin and TS pins.
Touch middleware	rm_touch	Default setting
Event Link Controller	Config_ELC	Set the event source to 32-bit interval timer unit 0. Set the output destination to CTSU2La Capacitive Sensing Unit.
Interval Timer	Config_ITL000	Set the operating clock to fSXP, clock frequency to fITL0/128, and interval time to 100 ms.
Ports	Config_PORT	Set P62 to High-level output.

5.3 Capacitive touch settings

Figure 5-2 shows the touch interface configuration. **Figure 5-3** shows the CTSU measurement touch sensor, sensor drive pulse frequency, and CTSU measurement time.

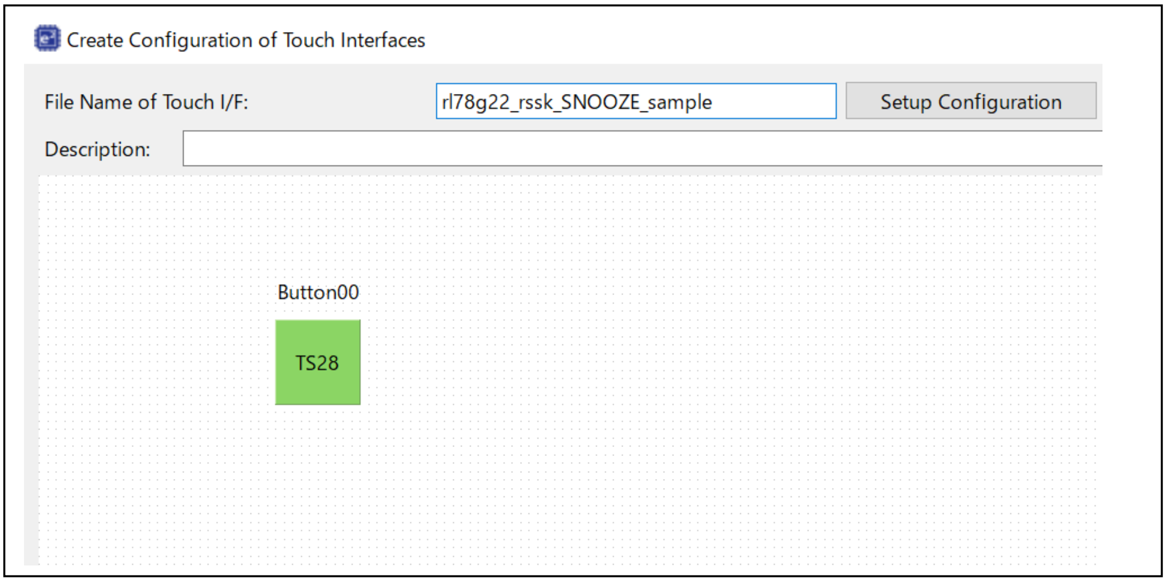


Figure 5-2 Touch interface configuration

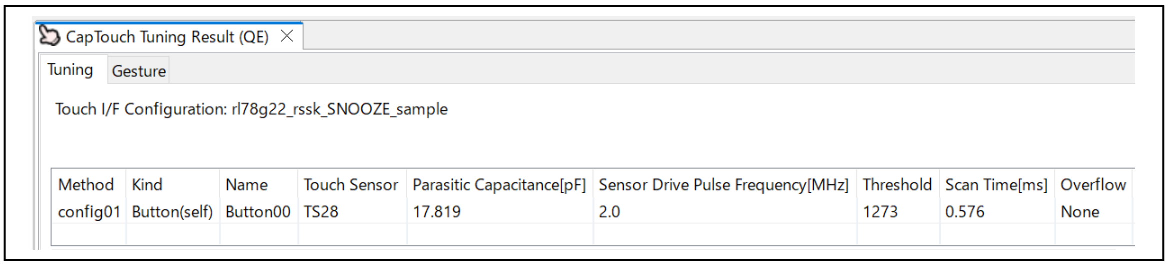


Figure 5-3 CTSU measurement Touch sensor and Sensor Drive Pulse Frequency and CTSU Scan Time

5.4 CTSU Startup Setting

In this project, "Use an external trigger" is enabled in QE for Capacitive Touch as shown in **Figure 5-4**.

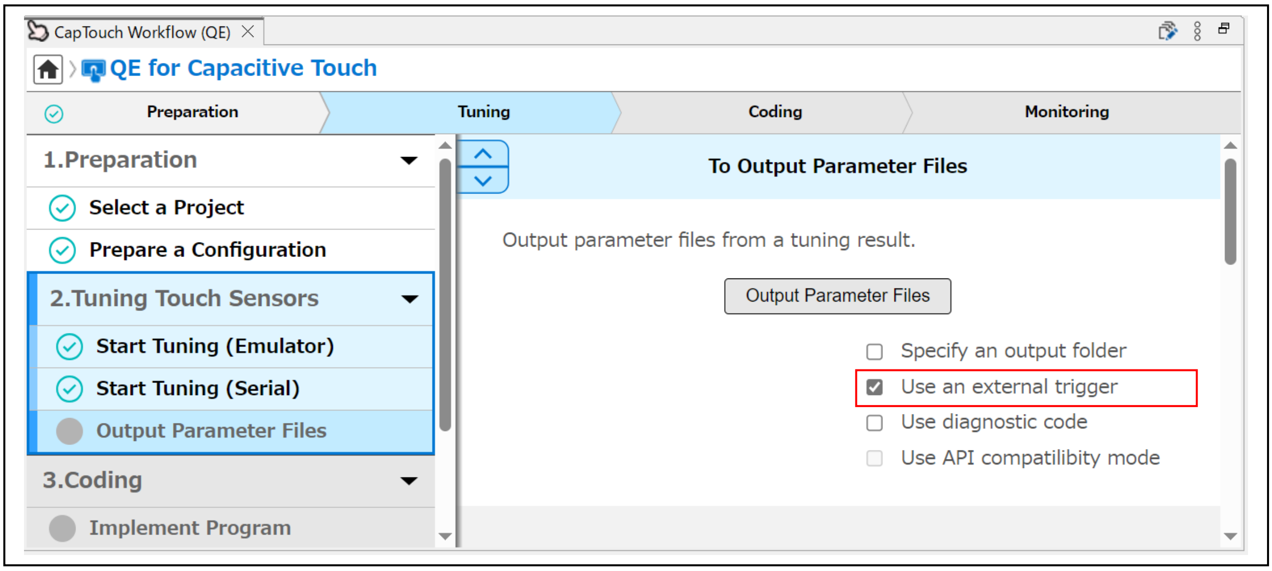


Figure 5-4 Setting to select external trigger in QE for Capacitive Touch

5.5 Setting of Option Byte

Table 5-2 shows the option byte settings.

Table 5-2 Option Byte Settings

Address	Setting Value	Contents
000C0H / 020C0H	11101111B (0xEF)	Disables the watchdog timer. (Counting stopped after reset)
000C1H / 020C1H	11111100B (0xFC)	LVD0 detection voltage : reset mode At rising edge : 2.67V (TYP) (2.59V～2.75V), At falling edge : 2.62V (TYP) (2.54V to 2.70V)
000C2H / 020C2H	11101000B (0xE8)	HS mode High-speed on-chip oscillator clock (f_{IH}) : 32 MHz
000C3H / 020C3H	10000100B (0x84)	Enables on-chip debugging

5.6 Folder/File Configuration

Table 5-3 lists the sample code folders generated by Smart Configurator and QE for Capacitive Touch.
Table 5-4 lists the file added or changed in the sample code.

Table 5-3 Sample code folders

Folder name	Description
qe_gen	Folder generated by QE for Capacitive Touch
smc_gen	Folder generated by Smart Configurator

Table 5-4 Files added or changed in the sample code

File name	Outline	Change/Additional contents
rl78g22_rssk_SNOOZE_sample.c	Main processing	Added the following processing : - Maskable interrupts are enabled. - A touch measurement control processing function was added.
mcu_clocks.c	Clock setting	Changed the setting value of CMC register (Connection of unused pins for P123 and P124)
qe_touch_sample.c	Touch measurement control processing	Added the following processing : - Processing to provide low power consumption was added. - LED control processing was added. Changed the setting value of CSC register (Connection of unused pins for P123 and P124)

5.7 Sample Code Details

5.7.1 List of variables

Table 5-5 lists the global variable that is used by this sample program

Table 5-5 Global variable

Type	Variable name	Description
uint64_t	button_status	Variable to check the button status
uint8_t	g_qe_touch_flag	Measurement completion flag

5.7.2 List of Functions

The specification for functions added or changed in this sample code are shown below.

main()		
Synopsis	Main processing	
Declaration	void main(void)	
Explanation	Call the qe_touch_main()	
Argument	-	-
Return value	-	

qe_touch_main()		
Synopsis	Touch measurement control processing	
Declaration	void qe_touch_main(void)	
Explanation	Configure the touch measurement settings to use the SNOOZE function, and then control the touch measurement. Touch measurement is repeated, and each time a measurement result is obtained, LED2 is turn on when touch-on is detected, and LED2 is turn off when touch-off is detected.	
Argument	-	-
Return value	-	

5.7.3 Flowcharts

5.7.3.1 Main Processing

Figure 5-5 shows the flowchart for the main processing.

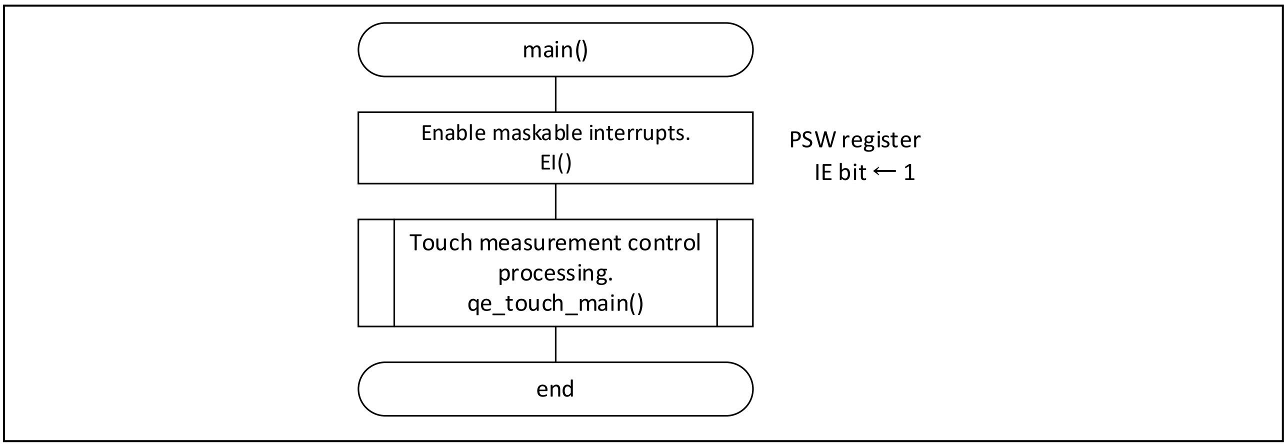


Figure 5-5 Main processing

5.7.3.2 Touch Measurement Control Processing

Figure 5-6, **Figure 5-7** and **Figure 5-8** show the flowchart for touch measurement control processing.

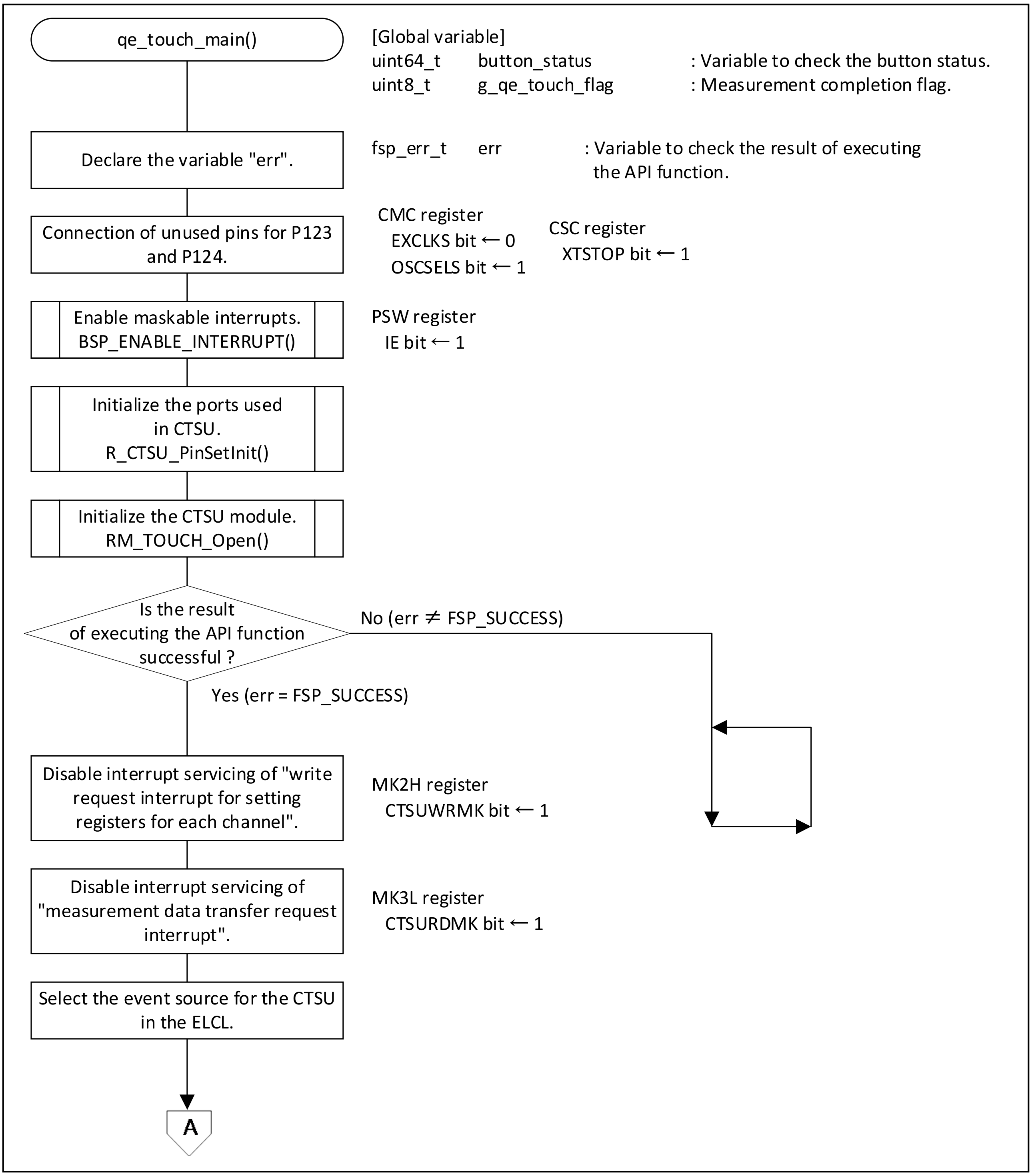


Figure 5-6 Touch measurement control processing (1/3)

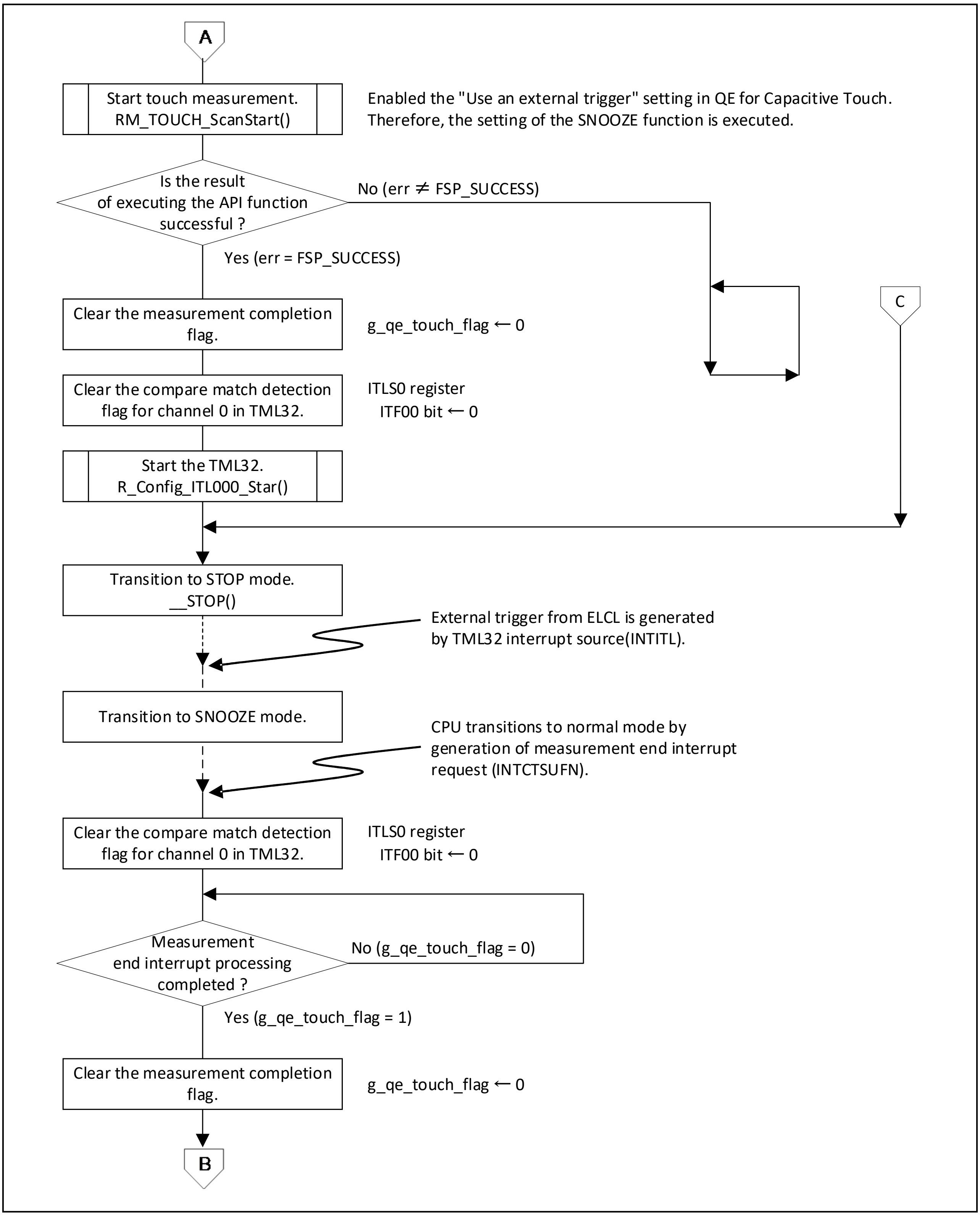


Figure 5-7 Touch measurement control processing (2/3)

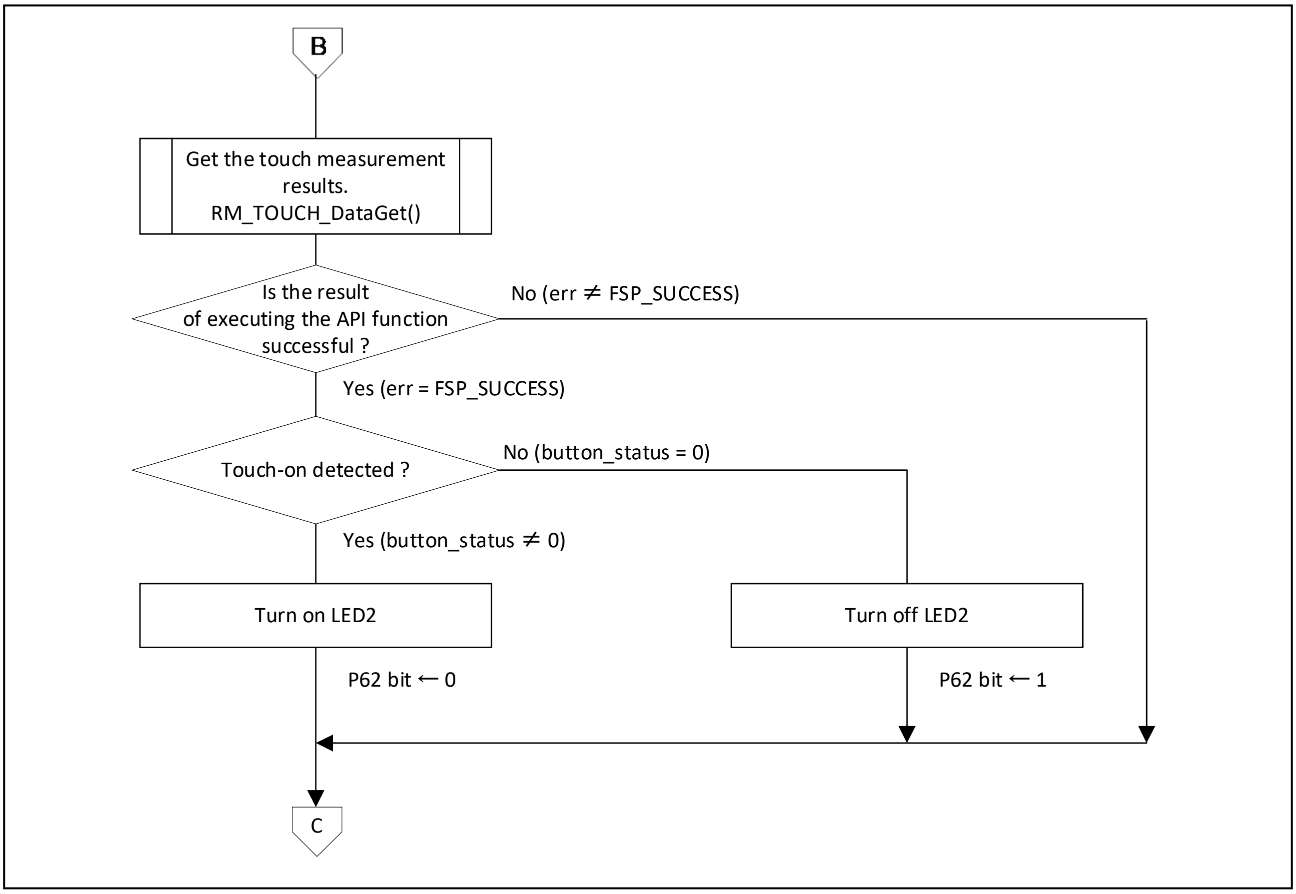


Figure 5-8 Touch measurement control processing (3/3)

6. Current consumption

6.1 Assumed System

The capacitive touch low-power operation in the sample code assumes the system shown in the red box in **Figure 6-1**. The system is in standby state and only the electrostatic capacitance touch button (power button) is measured at 100 ms cycle to determine whether a touch is detected.

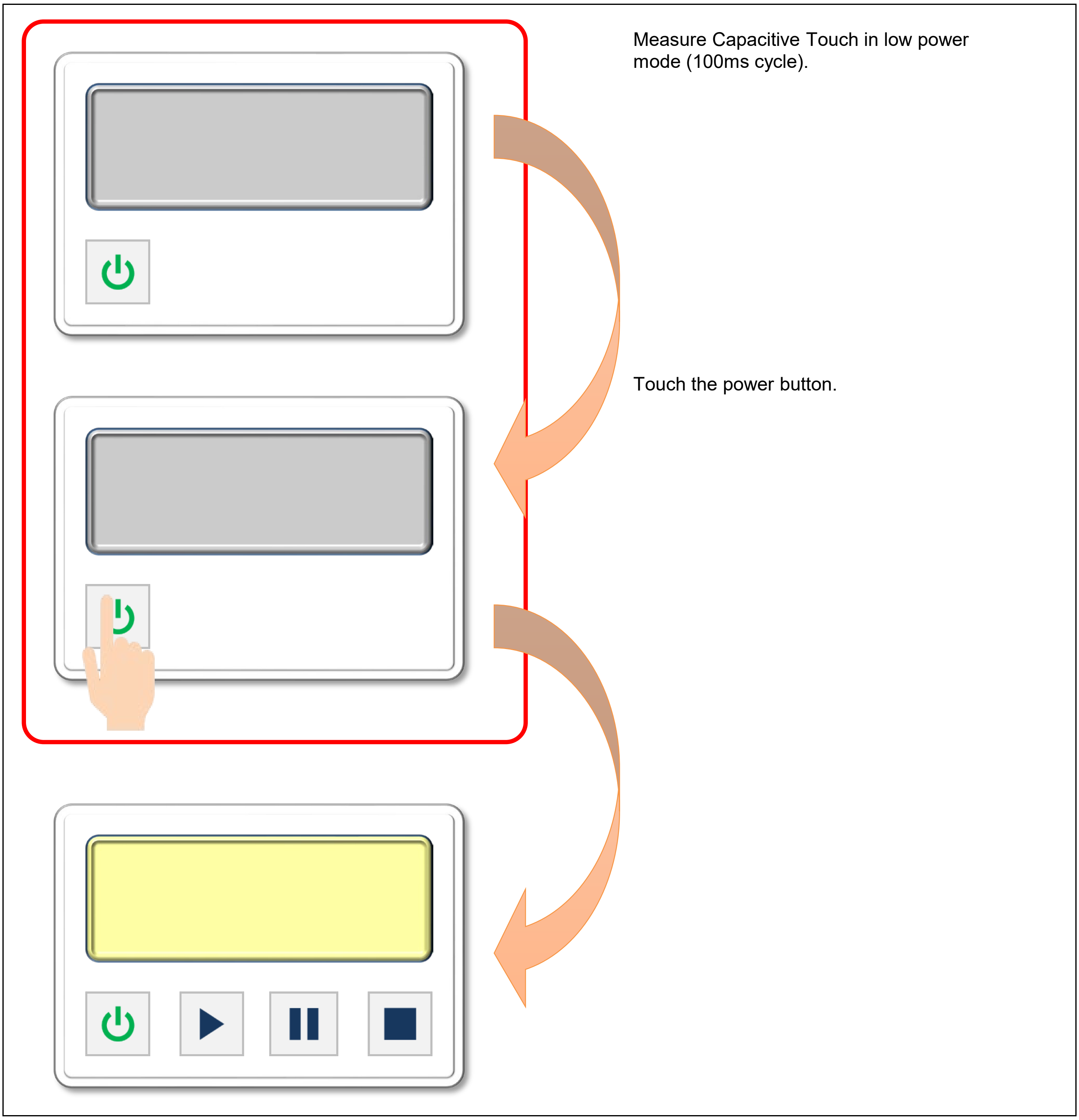


Figure 6-1 Model of the electrostatic capacitive touch low power consumption operation

6.2 Environment to measure current consumption

Figure 6-2 shows environment to measure current consumption.

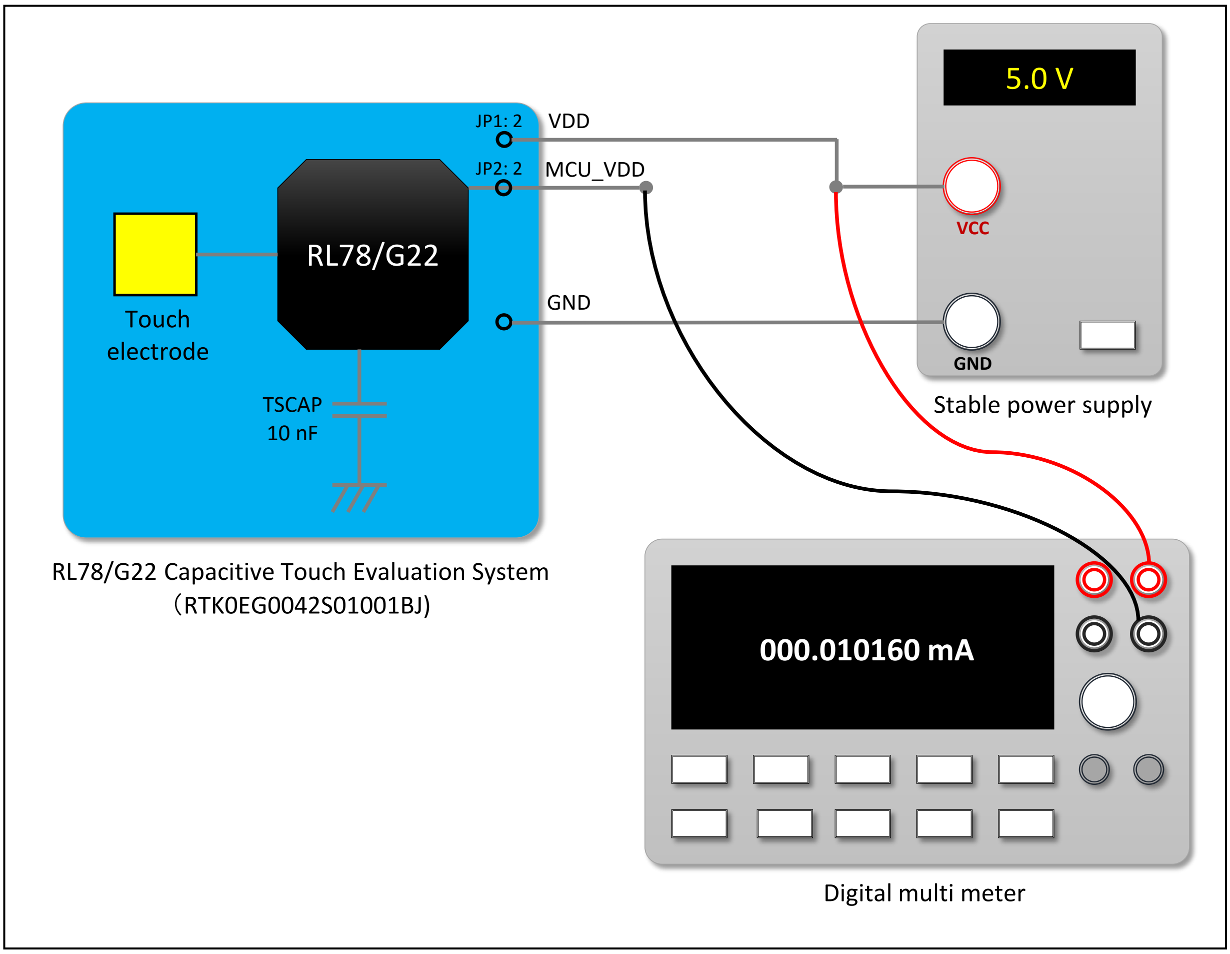


Figure 6-2 Environment to measure current consumption

6.3 Equipment / Software

Table 6-1 shows the equipment and software used in current consumption measurement

Table 6-1 Equipment / Software

Type	Name	Use
Digital multi meter	Keithley DMM7510	Measure current consumption.
Stable power supply	KENWOOD PWR18-2TP	Supply power to RL78/G22 Capacitive Touch Evaluation System (RTK0EG0042S01001BJ) CPU board.
Software	KickStart（V1.9.8.21）	Get result of current consumption measurement from Keithley DMM7510 and output the result to log-file.

6.4 CPU Board Jumper Settings

Table 6-2 shows the jumper settings of the CPU board of the RL78/G22 Capacitive Touch Evaluation System to measure current consumption.

Table 6-2 CPU board jumper settings

Position	Circuit group	Jumper	Use
JP1	VDD power	Open (Refer to the **Figure 6-2**)	Power supply from JP1: 2
JP2	MCU_VDD power	Open	Measure current consumption
SW4	USB Serial Conversion / Application Header (CN2)	ON (2-3, 5-6pin)	—
SW5	Clock Circuit / Application Header (CN1)	ON (2-3, 5-6pin)	—
SW6	Push Switch & LED / Application Header (CN1)	ON (2-3, 5-6pin)	Used to turn on LED2
SW7	Capacitive Touch	OFF(2-3pin)	—

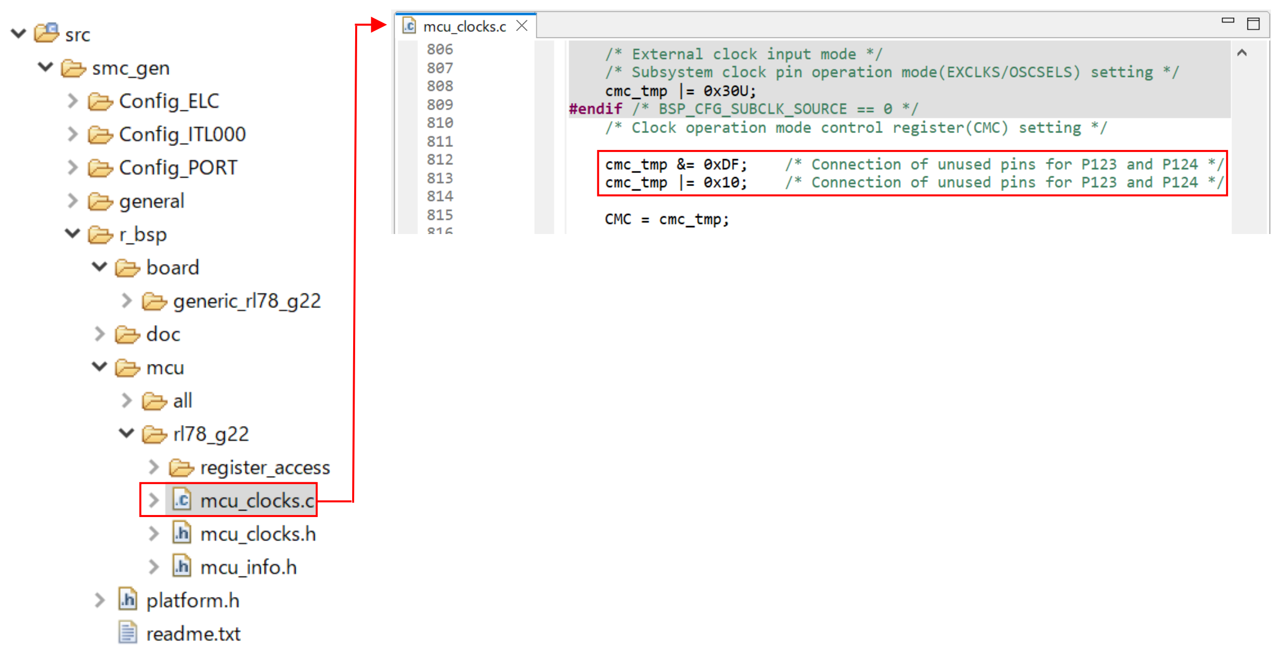
6.5 Setting of Unused Pins

Table 6-3 shows the settings specified in Smart Configurator input/output ports to prevent current from flowing into unused pins during current consumption measurement by using RL78/G22 Capacitive Touch Evaluation System (RTK0EG0042S01001BJ). For any pins that also function as TS pins and that are not included in the following table, “Use” is selected in the CTSU module.

Table 6-3 Port setting

Pin number	Port name	Destination of connection	Setting	Remarks
1	P60	—	Input	connect the pins to VDD via resistors
2	P61	—	Input	connect the pins to VDD via resistors
4	P63	LED3	High Output	Set not to turn on LED3
31	P21	CN1(pin number : 14(LED_ROW1))	High Output	Set not to turn on the matrix LED on the electrode board
32	P20	CN1(pin number : 13(LED_ROW0))	High Output	Set not to turn on the matrix LED on the electrode board
37	P120	CN1(pin number : 12(LED_ROW3))	High Output	Set not to turn on the matrix LED on the electrode board
38	P41	CN1(pin number : 11(LED_ROW2))	High Output	Set not to turn on the matrix LED on the electrode board
41	P124/XT2	—	Input	Set the EXCLKS bit to 0 and the OSCSELS bit to 1 in the clock operation mode control register (CMC), set the XTSTOP bit in the clock operation status control register (CSC) to 1[Note]
42	P123/XT1	—	Input	
43	P137	—	Input	connect the pins to VSS via resistors
44	P122	—	Low Output	Open
45	P121	—	Low Output	Open

Note The CSC register is set by the qe_touch_main function in qe_touch_sample.c. The CMC register is set by the mcu_clock_setup function in mcu_clocks.c as shown in **Figure 6-3**.



Caution If you change the version of the "r_bsp" component in the Smart Configurator, mcu_clocks.c will be overwritten and any code added by the user will be erased.
Therefore, whenever you change the version of the "r_bsp" component, you will need to add the code for the above settings.

Figure 6-3 Edit the mcu_clocks.c

6.6 Settings of Current Consumption Measurement Software

Figure 6-4 shows the settings of current consumption measurement software (Keithley KickStart).

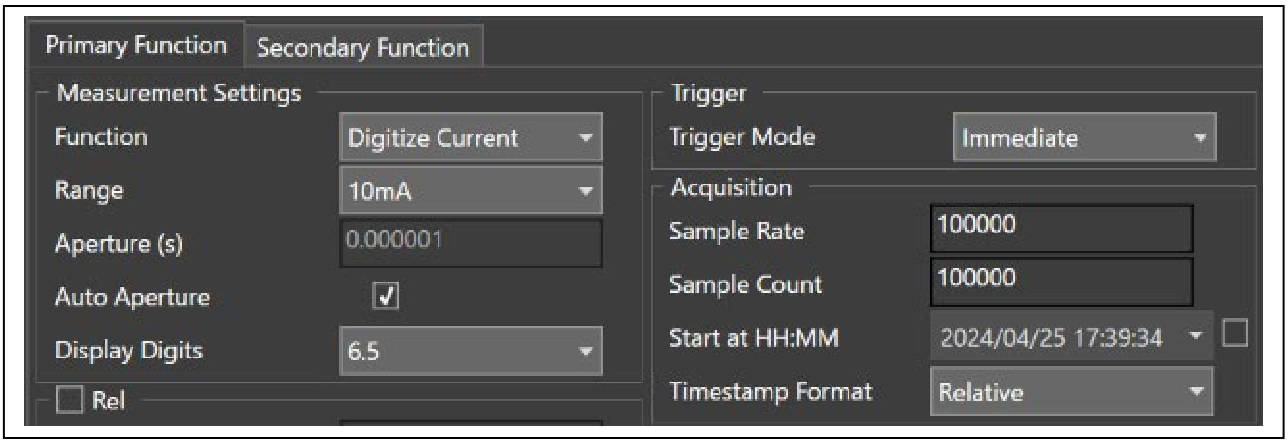


Figure 6-4 Settings of current consumption measurement software (Keithley KickStart)

6.7 Current Consumption Measurement Results

This section shows a comparison of the results of current consumption measurement in low power consumption operation and those in normal operation.
For low power consumption operation, intermittent operation is performed in touch measurement by using the SNOOZE function.
For normal operation, touch measurement is always performed in Normal (normal operation) mode without using the SNOOZE function or intermittent operation.
In both operations, current consumption is measured with LED2 turned off.

6.7.1 Current Consumption Waveforms in Intermittent Operation

Figure 6-5 shows the current consumption waveforms when touch measurement is performed every 100 ms in low power consumption operation.

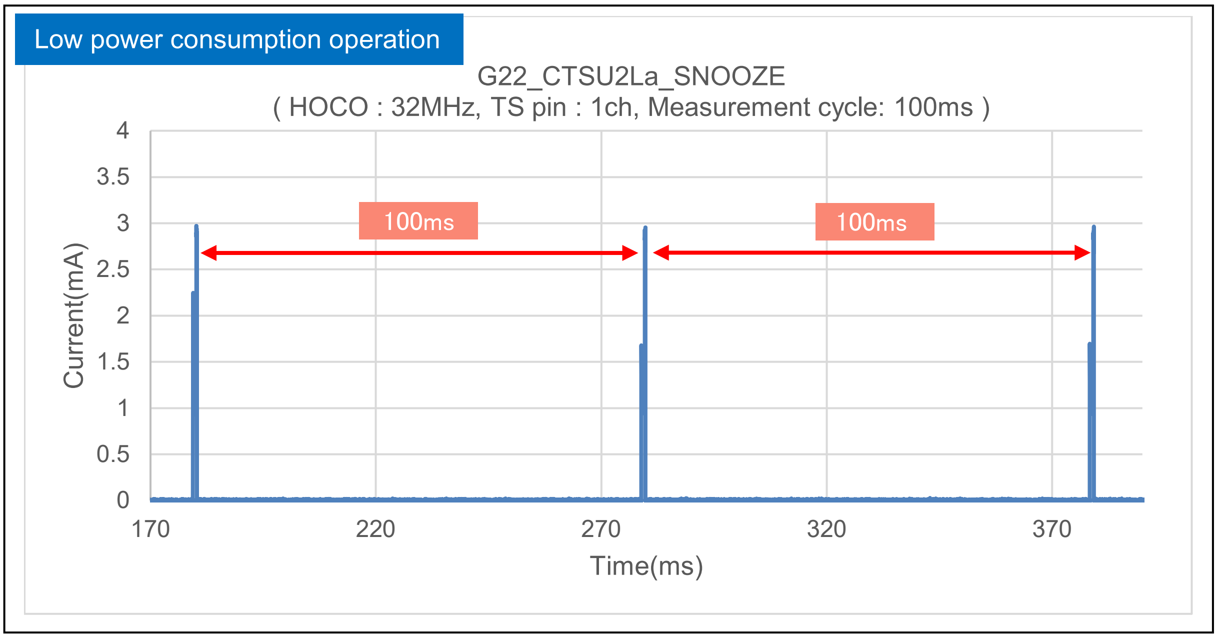


Figure 6-5 Current consumption waveforms when touch measurement is performed every 100 ms (Low power consumption operation)

Figure 6-6 shows the current consumption waveforms when touch measurement is performed every 100 ms in normal operation.

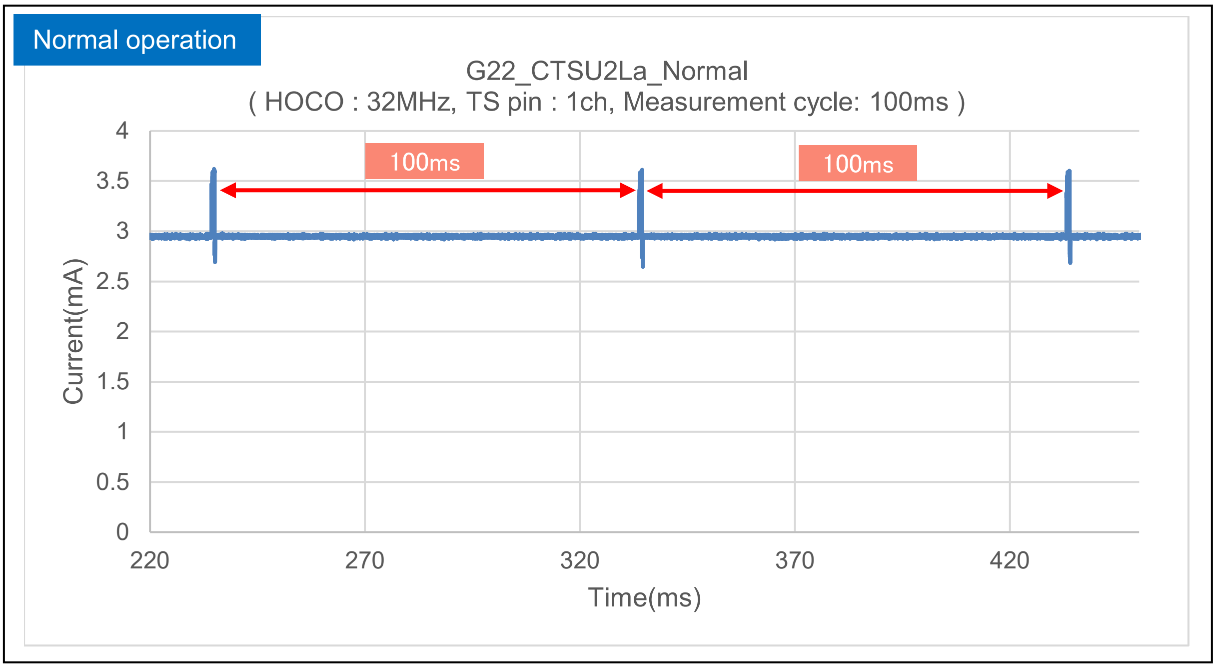


Figure 6-6 Current consumption waveforms when touch measurement is performed every 100 ms (Normal operation)

6.7.2 Current Consumption Waveforms During Touch Measurement Processing

Figure 6-7 shows the current consumption waveforms during touch measurement processing in low power consumption operation. **Table 6-4** shows the status of the CPU and CTSU in each measurement section. This figure shows the current consumption waveforms for a series of operations in which the CPU status transitions from STOP mode to SNOOZE mode to perform touch measurement processing, and then to Normal (normal operation) mode to perform touch measurement end processing + Touch On/Off judgment processing.

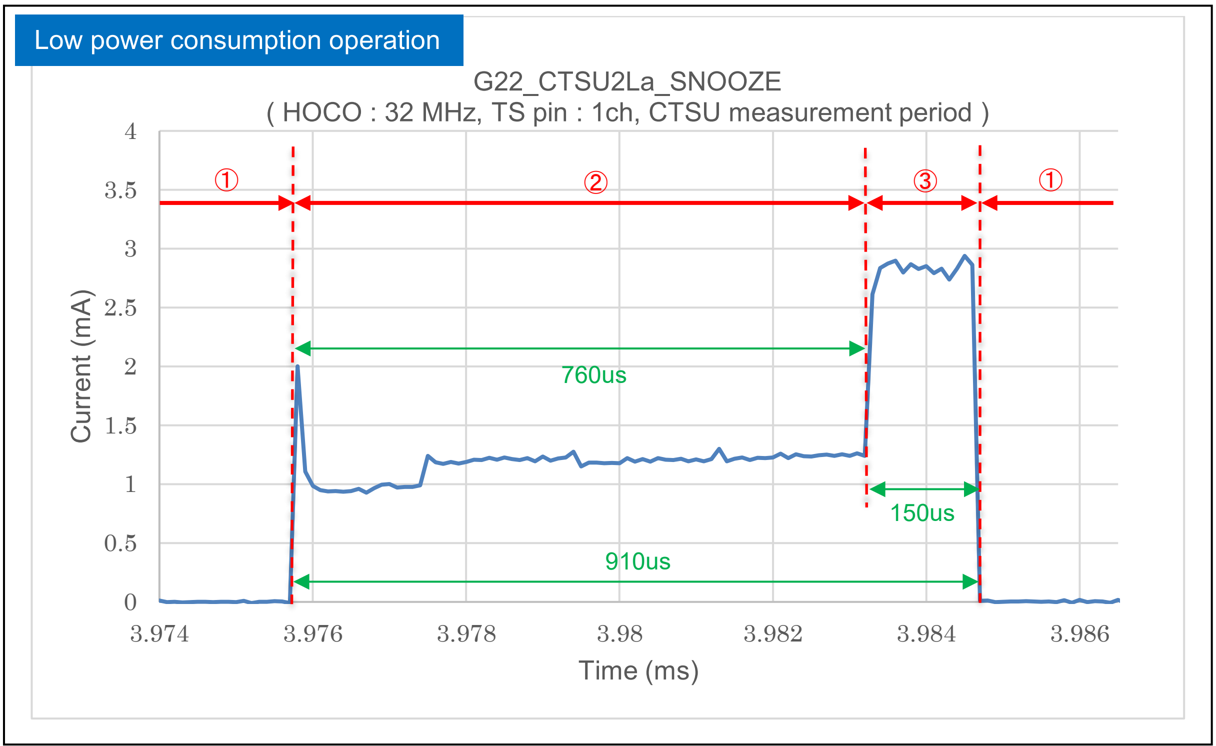


Figure 6-7 Current consumption waveforms during touch measurement processing (Low power consumption operation)

Table 6-4 Status of the CPU and CTSU in each measurement section (Low power consumption operation)

Number in the figure	Status of the CPU	Status of the CTSU
①	STOP mode	Suspend
②	SNOOZE mode	Measurement operation
③	Normal (Normal operation) mode[Note 2]	Measurement standby[Note 1]

Note 1. CTSU macros are stopped.
Note 2. The CPU performs calculation processing of measurement results.

Figure 6-8 shows the current consumption waveforms during touch measurement processing in normal operation. **Table 6-5** shows the status of the CPU and CTSU in each measurement section. This figure shows the current consumption waveforms for a series of operations (touch measurement processing, touch measurement end processing, and touch on/off judgment processing) when the CPU is always running in Normal (normal operation) mode.

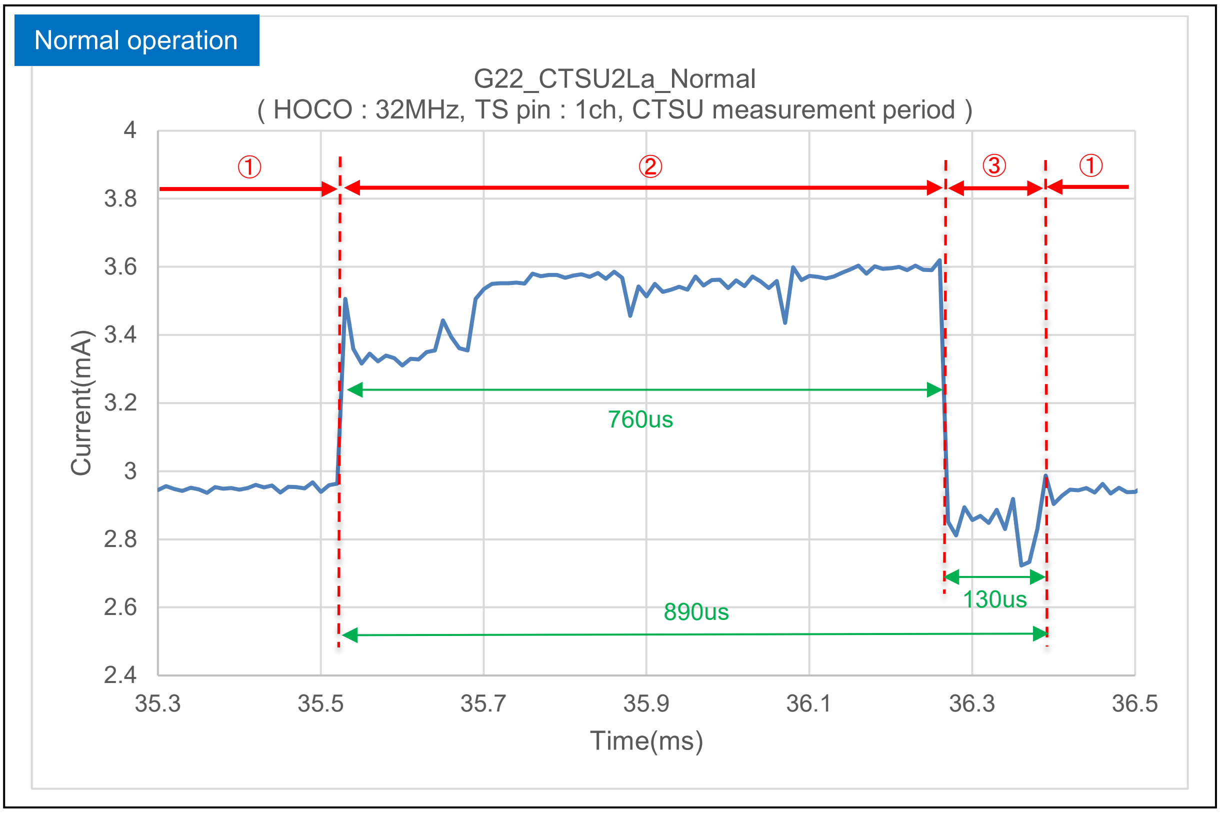


Figure 6-8 Current consumption waveforms during touch measurement processing (Normal operation)

Table 6-5 Status of the CPU and CTSU in each measurement section (Normal operation)

Number in the figure	Status of the CPU	Status of the CTSU
①	Normal (Normal operation) mode[Note 2]	Suspend
②	Normal (Normal operation) mode	Measurement operation
③	Normal (Normal operation) mode	Measurement standby[Note 1]

Note 1. CTSU macros are stopped.
Note 2. The CPU performs calculation processing of measurement results.

6.7.3 Average Current Consumption per 100 ms

The following shows the average current consumption results obtained by the Keithley/KickStart software. **Table 6-6** shows the average current consumption for each CPU mode and the average current consumption per 100 ms in low power consumption operation.

Table 6-6 Average current consumption results (Low power consumption operation)

CPU mode	Time (ms)	Average current consumption (uA)
STOP mode	99.09	0.68
SNOOZE mode < touch measurement processing >	0.76	1160
Normal (Normal operation) mode < touch measurement end processing + touch on/off judgment processing >	0.15	2630
STOP mode + SNOOZE mode + Normal (Normal operation) mode	100	13.43

Average current consumption per 100 ms ≒ 13.43 uA

Table 6-7 shows the average current consumption per 100 ms in normal operation.

Table 6-7 Average current consumption results (Normal operation)

CPU mode	Time (ms)	Average current consumption (uA)
Normal (Normal operation) mode < touch measurement processing + touch measurement end processing + touch on/off judgment processing >	100	2951

Average current consumption per 100 ms ≒ 2.951 mA

Table 6-8 shows the difference in the average current consumption per 100 ms in each operation condition based on **Table 6-6** and **Table 6-7**.

Table 6-8 Difference in average current consumption for operation condition

Operation condition	SNOOZE function and intermittent operation	CPU mode transition	Average current consumption per 100 ms
Low power consumption operation	Used	STOP mode → SNOOZE mode → Normal (Normal operation) mode	13.43uA
Normal operation	Unused	Always in Normal (Normal operation) mode	2.951mA

Reduced to approximately 1/220

7. Sample Code

The sample code is available on the Renesas Electronics Website.

8. Points for Caution when Using the Sample Code (About the timing when the touch measurement result is determined)

8.1 Effect of Positive Filter / Negative Noise Filter

The timing when the touch measurement result is determined depends on the setting of the positive noise filter and negative noise filter. The touch measurement result is determined when the capacitance measurement value is either touch ON or touch OFF for a period of times (each noise filter setting value + 1 times) consecutively.

In the sample code, the positive noise filter and negative noise filter cycle is set to "3" each. Therefore, the RM_TOUCH_DataGet function is called every touch measurement cycle to get the measurement result, and the touch measurement result is determined to be ON or OFF when the status of either touch ON or touch OFF is four times consecutively.

For details about positive noise filter and negative noise filter and how to change the setting value, see the application Note RL78 Family TOUCH Module Software Integration System [R11AN0485].

8.2 Effect of moving average

The timing when the touch measurement result is determined depends on the setting of the moving average.

In the sample code, the number of times the moving average is set to "4".

The number of times the moving average is set by the variable "num_moving_average" in the qe_touch_config.c file.

Figure 8-1 shows the example of moving average processing operation.

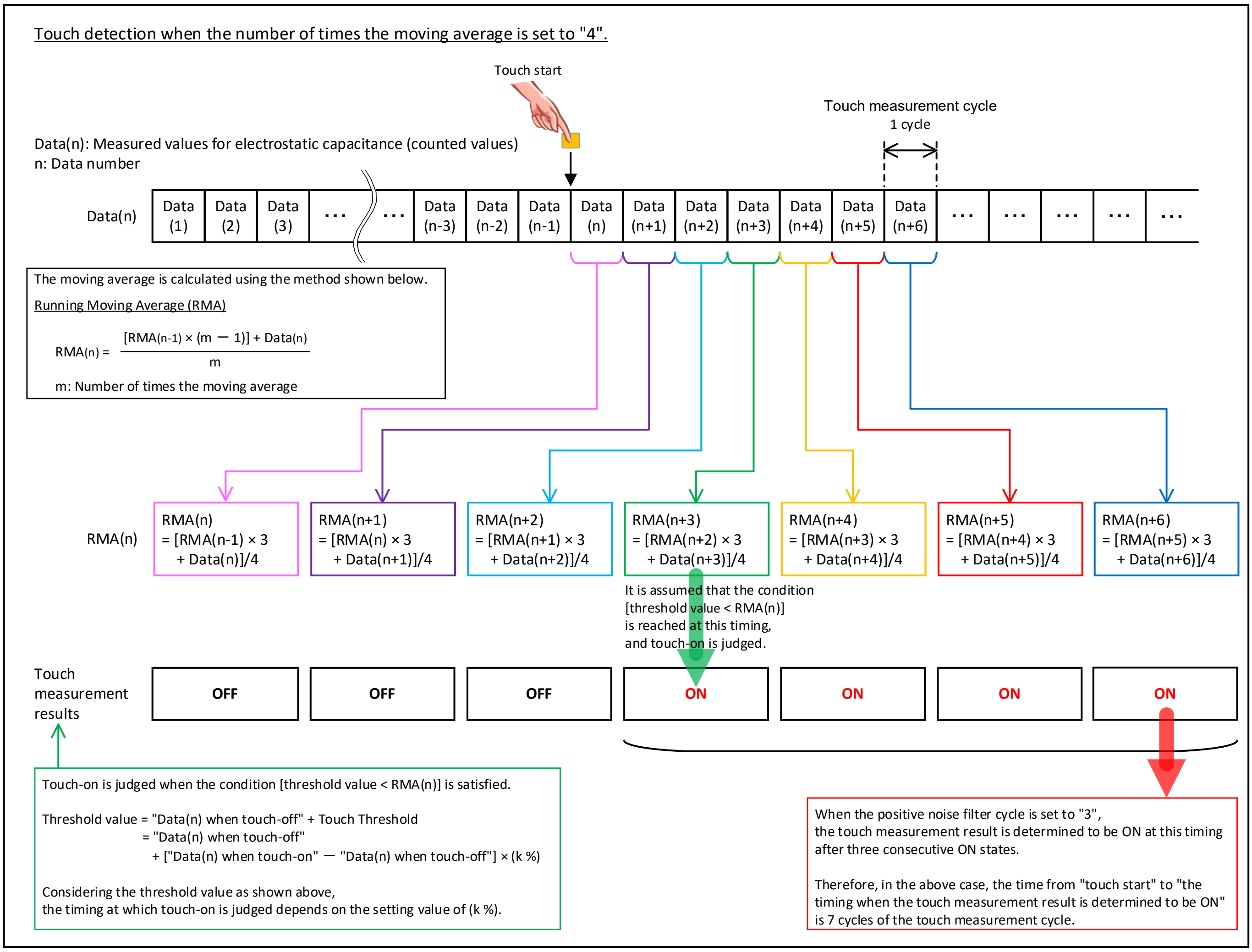


Figure 8-1 Example of moving average processing operation

9. Documents for Reference

〇User's Manual (The latest versions of the documents are available on the Renesas Electronics Website.)

RL78/Gxx User's Manual: Hardware

- RL78/G22 User's Manual: Hardware [R01UH0978]

RL78 Family User's Manual: Software [R01US0015]

Development Tools

RL78/G22 Capacitive Touch Evaluation System (RTK0EG0042S0100BJ) [R12UZ0110]

〇Technical Updates/Technical Brochures

(The latest information of the documents is available on the Renesas Electronics Website.)

〇Application Note (The latest versions of the documents are available on the Renesas Electronics Website.)

RL78 Family Capacitive Touch Sensing Unit (CTSU2L) Operation Explanation [R01AN5744]

RL78 Family Using QE and SIS to Develop Capacitive Touch Applications [R01AN5512]

RL78 Family CTSU Module Software Integration System [R11AN0484]

RL78 Family TOUCH Module Software Integration System [R11AN0485]

Capacitive Sensor Microcontrollers CTSU Capacitive Touch Electrode Design Guide [R30AN0389]

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

RL78 Family Capacitive Touch Low Power Application Development using SMS [R01AN7261]

Capacitive Sensor MCU QE for Capacitive Touch Advanced Mode Parameter Guide [R30AN0428]

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Capacitive Sensing Unit related page
https://www.renesas.com/solutions/touch-key
https://www.renesas.com/qe-capacitive-touch

Inquiries
http://www.renesas.com/contact/

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Oct.30.24	-	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/.