

Renesas Synergy™ Platform

QE for Capacitive Touch for e² studio Quick Start Guide

Introduction

This document will create an application example that integrates QE for capacitive touch using e² studio, this also provides instructions for building and debugging.

Target Device

S5D9 Microcontroller Group (R7FS5D97E3A01CFC)

Operating Environment

Target Board	PK-S5D9
IDE	e ² studio version 2021-10 and SSP v2.2.0
Toolchains	GNU Arm Embedded Toolchain: 9-2019-q4-update
	(GNU ARM Embedded 9.2.1.20191025)
QE	QE for Capacitive Touch V3.0.2

Note: Please download and install tools from the following URL in advance.

- QE for Capacitive Touch V3.0.2 Release Note download site : <u>https://www.renesas.com/software-tool/qe-capacitive-touch-development-assistance-tool-capacitive-touch-sensors</u>
- Renesas QE download site: <u>https://www.renesas.com/software-tool/qe-tools-particular-applications</u>
- Renesas Synergy™ e2 studio v2021-10 or higher Quick Start Guide download site: <u>https://www.renesas.com/software-tool/e-studio</u>
- Renesas Synergy™ Software Package (SSP) download site: <u>https://www.renesas.com/products/microcontrollers-microprocessors/renesas-synergy-platform-mcus/renesas-synergy-software-package</u>
- Promotion Kit S5D9 (PK-S5D9) User's Manual: <u>https://www.renesas.com/document/mat/promotion-kit-s5d9-pk-s5d9-users-manual</u>
- S5D9 Microcontroller Group User's Manual: <u>https://www.renesas.com/document/mah/s5d9-microcontroller-group-users-manual</u>



Contents

1.	Installation	3
1.1	Install SSP with e ² studio	3
1.2	Install QE for Capacitive Touch	3
2.	Creating a project	3
2.1	Creating a Bare Metal – Minimal Project	
2.2	Adding TOUCH Driver	
3.	Configuration for CapTouch Main	9
3.1	Executing the "Select a Project"	
3.2	Executing the "Preparing a Configuration"	10
4.	Connecting PC and PK-S5D9 Board	15
5.	Tuning on CapTouch Main	16
5.1	Executing the "Start Tuning"	16
5.2	Executing the "Output Parameter Files"	20
6.	Coding on CapTouch Main	21
7.	Debugging	23
8.	Monitoring on CapTouch Main	26
8.1	Monitoring on detects capacitive button S3	26
8.2	Displaying and measuring standard deviation	29
Rev	rision History	31



1. Installation

1.1 Install SSP with e² studio

Refer to section 2.2, Installing e² studio and SSP Independently of Renesas Synergy[™] e2 studio v2021-10 or higher Quick Start Guide.

1.2 Install QE for Capacitive Touch

Refer to section 2.1.1, Install from the Renesas Software Installer menu of e^2 studio of QE for Capacitive Touch V3.0.2 Release Note.

2. Creating a project

e² studio has a simple wizard for creating projects. You can create a new Synergy™ project by specifying the project name, corresponding devices and boards, project type, output object type, and project template.

Start the e² studio application and choose a workspace folder in the Workspace Launcher. To create a new Synergy™ project, follow these steps:

2.1 Creating a Bare Metal – Minimal Project

1. Select File menu > New > Synergy C/C++ Project.

e ,	vorkspace - e² studio				
File	Edit Source Refactor Navigate	Search Project	Re	nesas Views Run Window Help	
	New	Alt+Shift+N >		Renesas C/C++ Project	>
	Open File		Jerena	Synergy C/C++ Project	
	Open Projects from File System		C +	Makefile Project with Existing Code	
	Recent Files	>	C	C/C++ Project	
	Close Editor	Ctrl+W	Ċ	Project	

Figure 2-1. Synergy C/C++ Project

2. Select the Renesas Synergy C Executable Project template. Click Next to continue.

🛐 New Synergy (C/C++ Project	— 🗆	×
Templates for N	lew Synergy C/C++ Project		
All C/C++	Renesas Synergy C Executable Pro RENESAS A C Executable Project for Renesas Syr		^
	Renesas Synergy C Library Project Renesas A C Library Project for Renesas Synergy		
	<		>
	< Back Next > Finish	Cane	cel

Figure 2-2. Renesas Synergy C Executable Project



3. In the next dialog box, enter a project name and click **Next**.

Specify the new project details.	
Project	Toolchains
Project name synergy_CapTouch	GNU ARM Embedded
Use default location	
Location: C:\Synergy\workspace\synergy_CapTouch Browse	2
Choose file system: default 🗸	
? < Back Net	ext > Finish Cancel

Figure 2-3. Project Name and Location

- 4. In the device selection dialog, enter device and tool information:
 - SSP version: 2.2.0
 - Board: S5D9 PK
 - Device: Auto selected
 - Toolchain version: GNU ARM Embedded 9.2.1.20191025
 - Debugger: J-Link ARM
 - Click Next to continue.

 e2 studio - Project Configuration (Syne Device Selection SSP version: 2.2.0 Board: SSD 9 PK Device: R7FS5D97E3A01CFC 				
Select Tools Toolchain: GNU ARM Embedder Toolchain version: 9.2.1.20191025 Debugger: J-Link ARM	1 ~ ~	Available Tools GNU ARM Embedded 9.2.1.20191025 Debuggers J-Link ARM KITOS Express Logic Thread Software Manual IO Registers Supportu Software Manual Sup	ed	
?	< Back Next	> Finish	Cancel	

Figure 2-4. Create New Project for S5D9 PK



5. In the project template dialog, select **BSP** and click **Finish**.

e2 studio - Project Configuration (Synergy C Executable Project) Select the type of project you wish to create.
Project Template Selection
 Blinky Blinky project. [Renesas:Synergy:2.2.0.pack] Blinky with ThreadX Threaded version of Blinky project. [Renesas:Synergy:2.2.0.pack]
Code Generation Settings ☑ Use Synergy Code Formatter

Figure 2-5. Select the type of project

6. Once complete, e² studio creates a new project with the **Synergy Configuration** perspective open and ready for project configuration.



Figure 2-6. New project for S5D9 PK



2.2 Adding TOUCH Driver

1. In Synergy Configuration, select the **Pins** tab at the bottom of the pane. Type **ctsu** in the search box and select the **CTSU0** pin in the search result.

Pins Configuration	Configuration 🗙			0
rins configuration				Generate Project Content
Select pin configuration				Pins Tutorial 🔏 👻 🗐
S5D9-PK.pincfg	✓ Generate data: g_bsp_pi	in_cfg		
Pin Selection	Pin Configuration			
Thi Selection				
ctsu	×			
 Peripherals Input-CTSU 	Module name:	CTSU0		
✓ CTSU0	Operation Mode:	Enabled	\sim	
	Input/Output			
	TSCAP:	✓ P205	\sim	\Rightarrow
	TS00:	✓ P204	\sim	\Rightarrow
	TS01:	✓ P206	\sim	\Rightarrow
	TS02:	✓ P207	\sim	\$
	TS03:	None	~	
	TS04:	✓ P408	\sim	\$
	TS05:	✓ P409	~	\$
	TS06:	None	~	
	TS07:	None	~	
	TS08:	None	~	
	TS09:	None	~	
	TS10:	✓ P414	~	\$
	TS11:	✓ P415	~	4
	TS12:	None	~	
	<			>

Figure 2-7. Select Pin CTSU0

 Ensure that the Operation Mode and the TS pins below are enabled: TSCAP, TS00, TS01, TS02, TS04, TS05, TS10, TS11.
 Note: Refer to section 4.8, Capacitive Touch Interface of *Promotion Kit S5D9 (PK-S5D9) User's Manual*.

[synergy_CapTouch] Synergy Configuration ×				- e
Pins Configuration				Generate Project Content
Select pin configuration				Pins Tutorial 🔏 👻 🚮
S5D9-PK.pincfg \lor Generate d	ata: g_bsp_pin_cfg			
Pin Selection Pin Config				
Pin Selection Pin Config	uration			
ctsu ×				
✓ Peripherals Module n ✓ ✓ Input:CTSU	ame:	CTSU0		
 CTSU0 Operation 	Mode:	Enabled	~	
Input/Out	put			
TSCAP:		P205	~	\Rightarrow
TS00:	~	P204	~	\Rightarrow
TS01:	~	P206	~	
TS02:	~	P207	~	
TS03:	<u> </u>	None	~	
TS04:	~	P408	~	\Rightarrow
TS05:	~	P409	~	
TS06:		None	~	
TS07:		None	~	
TS08:		None	~	
TS09:	_	None	~	
TS10:		P414	~	\Rightarrow
TS11:	~	P415	~	
TS12:		None	~	
۲				>
Summary BSP Clocks Pins Threads Messaging Con	nponents			

Figure 2-8. Enable TS Pins for S5D9 PK



 Move to the Threads tab. Add the Capacitive Touch Driver by clicking New Stack > Framework > Input > Cap Touch Framework on sf_touch_ctsuv2.

Threads Configuration		Ge	onerate Project Content	
Threads ♦ New Thread Remove	HAL/Common Stacks # g_cgc CGC Driver on r_cgc 1	New Stack > Extend Driver > r_elc X-Ware > X-Ware > X-Ware > X-Ware >	input -	Cap Touch Framework on sf_touch_ctsuv2
HAL/Common 🕢 New Object > Objects 🔬 Remove				
	<		>	

Figure 2-9. Add Cap Touch Driver

 Click on CTSU Driver on r_ctsuv2 to display properties. Change Support for using DTC to Enabled.

🎦 Project Explorer 🛛 🗖 🗖	∰ *[synergy_CapTouch] Synergy Configuration ⊠	□ □
	Threads Configuration	O Generate Project Content
) Includes) Signergy > Soript > Soriet > Sor	Threads New Thread Remove Image: Constraint of the second s	Cap Touch Framework on sf.touch_ctsuv2 Stack > Image: Stack stack stack > Image: Stack stack stack stack stack > Image: Stack sta
< >	Summary BSP Clocks Pins 😢 Threads Messaging Com	iponents
🔲 Properties 🗙 🛃 Problems 🍇	Smart Browser	📑 🖇 🗖 🗖
CTSU Driver on r_ctsuv2		
Settings Property API Info Common	Value	^
Support for using		
Interrupt priority I	evel Priority 12	

Figure 2-10. Properties in CTSU Driver on r_ctsuv2



5. Click on Add DTC Driver for Transmission to select the New > Transfer Driver on r_dtc to add the DTC driver for Transmission.



Figure 2-11. Add DTC Driver for Transmission

6. Click on Add DTC Driver for Reception to select the New > Transfer Driver on r_dtc to add the DTC driver for Reception.

VI Driver on	🖑 Cap Touch Framework	New Stack > Arr Arr Arr Arr Arr Arr Arr Arr Arr A		
	(i)			
		· • · · · · · · · · · · · · · · · · · ·		
	CTSU Driver on r_ctsuv.	2	Grimer Add SCI U for monit	
		▲		
	 g_transfer0 Transfer Driver on r_dtc CTSU WRITE WRITE 	Add DTC Driver for Reception [Recommended but optional]		
		New > 🕁	Transfer Driver	on r_dtc
<				>

Figure 2-12. Add DTC Driver for Reception

7. Click on the

Generate Project Content button to generate the source files.



3. Configuration for CapTouch Main

3.1 Executing the "Select a Project"

 From the menu of e² studio, select Renesas Views > Renesas QE > CapTouch Main (QE) to open the main perspective for configuring capacitive touch to the project.



Figure 3-1. Open CapTouch Main (QE)

2. In the CapTouch Main (QE) pane, select the project to configure the Touch interface for by using the pull-down tab and selecting the Synergy_CapTouch project as shown below.

1. Preparation		2. Tuning		:	3. Coding		4. Monitoring
Prepare a project that uses the interfaces.	e touch	QE will automatically perform processing for each touch set	i tuning nsor.		Implement a program using the touch interfaces.		You can check a behavior of touch interfaces and make fine adjustments.
To Select a Project Select the target project.	Ţ	To Connect Target Boa Connect your target board a an emulator. To Start Tuning Follow instructions in the d	and PC via		To Show Code Implement a program that periodically scans the status of the touch sensor in the main() function. Show Sample		To Launch Debug (via Emulator) Launch debugging for your target project and execute the program.
synergy_CapTouch Select or create a touch inte configuration. Modify Configuration	_	Start Tuning Start Tuning Chable advanced To Output Parameter F Output Parameter files from result. Output Parameter Specify an output	illes na tuning Files				To Connect UART Enable a monitoring function via serial communication, if you do not use an emulator. Baud rate Port Auto Connect
		Use an external tr	igger ode				To Enable Monitoring Show monitoring views and enable a monitoring function. Show Views
<							
Tuning Gesture Touch I/F Configuration: <n< td=""><td>lot Selected></td><td></td><td></td><td></td><td></td><td></td><td></td></n<>	lot Selected>						
Method Kind Name	Touch Sensor	Describin Conscious of a D	Canada Daira D	ule	e Frequency[MHz] Threshold Scan T	ime	alms] Overflow

Figure 3-2. Select Project



3.2 Executing the "Preparing a Configuration"

1. Create a new Touch configuration by using the lower pull-down and selecting **Create a new configuration**.

1. Preparation	2. Tuning	3. Coding	4. Monitoring
Prepare a project that uses the touch interfaces.	QE will automatically perform tuning processing for each touch sensor.	Implement a program using the touch interfaces.	You can check a behavior of touch interfaces and make fine adjustments.
To Select a Project Select the target project. synergy_CapTouch	To Connect Target Board Connect your target board and PC via an emulator. To Start Tuning Follow instructions in the dialog. Start Tuning Brandble advanced tuning To Output Parameter Files Output parameter files from a tuning result. Output Parameter Files Specify an output folder Use an external trigger Use diagnostic code	To Show Code Implement a program that periodically scans the status of the touch sensor in the main() function. Show Sample	To Launch Debug (via Emulator) Launch debugging for your target project and execute the program. To Connect UART Enable a monitoring function via serial communication, if you do not use an emulator. Baud rate 115200 Port Auto Connect To Enable Monitoring Show monitoring views and enable a monitoring function.
< Tuning Gesture Touch I/F Configuration: <not selected=""> Method Kind Name Touch Sensor</not>	Use API compatibility mode	ulse Frequency[MHz] Threshold Scan Tir	Show Views

Figure 3-3. Create a new configuration



- 2. Add Slider (vertical) to the canvas:
 - A. Selecting the **Slider (vertical)** menu item from the right-hand side and moving the mouse onto the canvas.
 - B. Click the left-hand mouse button to drop the Slider icon. Do this once to add one slider.
 - C. Press the **ESC** key to exit once the slider are added.

File Name of Touch I/F:	synergy_CapTouch Setup Configuration	Import / Re-edit
Description:		
		Touch I/F
		Capacitance Type
		Self Capacitance 🗸
		Button
	-	Slider (horizontal)
Slider00		Slider (vertical)
		Wheel
		Key pad
		3D Gesture (Al)
		Touch pad
		Shield Pin
	-	TC Pin
		Capacitance Sensor
		Current Sensor
		Diagnosis Pin
Setting		Remove Touch I/F
-	Setup Resistance Value Clear Assigned TSx	Kentove louch yr
😢 There are some problems with	i setting.	

Figure 3-4. Create New Touch Slider

- 3. Add 2 capacitive touch buttons (capacitive button) to the canvas by:
 - A. Selecting the **Button** menu item from the right-hand side and moving the mouse onto the canvas.
 - B. Click the left-hand mouse button to drop the button icon. Do this two times to add two buttons.
 - C. Press the **ESC** key to exit once the two buttons are added.

File Name of Touch I/F: Description:	synergy_CapTouch Setup Configuration	Import / Re-edit
		Touch I/F Capacitance Type Self Capacitance v
		Button
Slider00	Button00	Slider (horizontal)
Silderoo		Slider (vertical)
		Wheel
		Key pad
ľ V	Button01	3D Gesture (Al)
\sim		Touch pad
		Shield Pin
		TC Pin
		Capacitance Sensor
		Current Sensor
		Diagnosis Pin
Setting Setup Touch I/F Se	up Resistance Value Clear Assigned TSx	Remove Touch I/F
Setup Touch I/F Set		
-	y.	

Figure 3-5. Create New Touch Button



- 4. Hardware port assignments.
 - Note: Table 1 is created based on the following reference materials. For settings for step 5 and later, see *Table 1*.

Reference material :

- section 1.7, Pin Lists of S5D9 Microcontroller Group User's Manual.
- section 4.8, Capacitive Touch Interface of *Promotion Kit* S5D9 (PK-S5D9) User's Manual.

Table 1. Hardware port assignments

CTSU0	PORT	CAPACITIVE TOUCH
TSCAP	P205	-
TS00	P204	S1-1
TS01	P206	S3-1
TS02	P207	S2-5
TS04	P408	S2-4
TS05	P409	S2-3
TS10	P414	S2-2
TS11	P415	S2-1

5. Setup capacitive button:

- A. Double click on Button00 icon.
- B. Rename Button00 to S3 for using S3 Touch Switch on PK-S5D9.
- C. Using the pull-down and select **TS01** as the MCU sensor to assign to this button.

File Name of Touch I/F: Description:	synergy_CapTouch	Setup Configuration	Import / Re-edit
Sider00	Button01	iouch Interface X ton(self) me S3 uuch Sensor Resistance[ohm] 01 \$60 VK Cancel Help	Touch VF Capacitance Type Self Capacitance V Button Slider (horizontal) Slider (vertical) Wheel Key pad 3D Gesture (Al) Touch pad Shield Pin TC Pin
			Capacitance Sensor Current Sensor
			Diagnosis Pin
	tup Resistance Value Clear Assigned	TSx	Remove Touch I/F
Othere are some problems with s	etting.		

Figure 3-6. Setup Touch Sensor to Button00



- D. Double click on Button01 icon.
- E. Rename Button01 to S1 for using S1 Touch Switch on PK-S5D9.
- F. Using the pull-down and select **TS00** as the MCU sensor to assign to this button.

File Name of Touch I/F: Description:	synergy_CapTouch	Setup Configuration	Import / Re-edit
			Touch I/F Capacitance Type
			Self Capacitance V
			Button
			Slider (horizontal)
Slider00		up Touch Interface X	Slider (vertical)
\sim	1501	Button(self)	Wheel
		Name S1	Key pad
i V	Button01	Touch Sensor Resistance[ohm] TS00 560	3D Gesture (Al)
\sim			Touch pad
		OK Cancel <u>H</u> elp	Shield Pin
			TC Pin
			Capacitance Sensor
			Current Sensor
			Diagnosis Pin
Setting Setup Touch I/F Se	etup Resistance Value Clear Assig	170	Remove Touch I/F
Setup rouch // Setup	[152 152	

Figure 3-7. Setup Touch Sensor to Button01

- 6. Setup capacitive touch slider:
 - A. Double click on **Slider00** icon.
 - B. Rename Slider00 to S2.
 - C. Select Number of Touch Sensor 5.
 - D. Using the pull-down and select **TS02**, **TS04**, **TS05**, **TS10**, and **TS11** as the MCU sensors to assign to this slider.

File Name of Touch I/F:	synergy_CapTouch	Setup Configuration	Import / Re-edit
			Touch I/F Capacitance Type
	Setup Tour	ch Interface X	Self Capacitance v
	Slider		Slider (horizontal)
Slider00	53 Name Numb	er of Touch Sensor 5	Slider (vertical)
\sim	TS01	n Sensor Resistance[ohm]	Wheel
\sim	51 TS02 51 TS04 TS05	560 560 560	Key pad 3D Gesture (Al)
	TS00	560 560 V	Touch pad
		Reverse	TC Pin
		OK Cancel <u>H</u> elp	Capacitance Sensor
			Current Sensor
			Diagnosis Pin
Setting Setup Touch I/F Se Othere are some problems with s	tup Resistance Value Clear Assigned	TSx	Remove Touch I/F
Inere are some problems with s	etung.		Create Cancel Help

Figure 3-8. Setup Touch Sensor to Slider00



7. All sensor settings change icon to green, the warning **"There are some problems with setting."** disappears.

Click **Create** button to set up the Touch Interface.

	😢 There are some problem	s with setting.	
Create Configuration of Touch Int			
File Name of Touch I/F:	synergy_CapTouch	Setup Configuration	Import / Re-edit
Description:			Touch I/F
			Capacitance Type
			Self Capacitance 🗸 🗸
			Button
			Slider (horizontal)
52	S3		Slider (vertical)
ŤS11	TS01		Wheel
TS10			Key pad
TS05	S1		3D Gesture (Al)
	T500		Touch pad
TS02			Shield Pin
			TC Pin
			Capacitance Sensor
			Current Sensor
			Diagnosis Pin
Setting Setup Touch I/F Se	tup Resistance Va	x	Remove Touch I/F
Setup Toder ()1		n	

Figure 3-9. Create New Touch Interface



8. The CapTouch Main (QE) window will now display the configuration of the touch interface in the main view pane.



Figure 3-10. Main View Pane Displays the Touch Configuration

4. Connecting PC and PK-S5D9 Board

The picture below shows the connection between the host PC and the PK-S5D9 board. Setting to jumper J1 is in position 1-2 (default), the MCU boots in normal mode (from its ROM). Connecting the USB cable, USB port J19 for power supply and J-Link OB.



Figure 4-1. PK-S5D9 Board Connection



5. Tuning on CapTouch Main

5.1 Executing the "Start Tuning"

1. To start the automatic tuning process, click the button Start Tuning in the CapTouch Main (QE).

1. Preparation	2. Tuning	3. Coding	4. Monitoring
Prepare a project that uses the touch interfaces.	QE will automatically perform tuning processing for each touch sensor.	Implement a program using the touch interfaces.	You can check a behavior of touch interfaces and make fine adjustments.
To Select a Project Select the target project. synergy_CapTouch	To Connect Target Board Connect your target board and PC via an emulator. To Start Tuning Follow instructions in the dialog.	To Show Code Implement a program that periodically scans the status of the touch sensor in the main() function.	To Launch Debug (via Emulator) Launch debugging for your target project and execute the program.
To Prepare a Configuration Select or create a touch interface configuration. synergy_CapTouch.tifcfg v Modify Configuration	Start Tuning	Show Sample	To Connect UART Enable a monitoring function via serial communication, if you do not use an emulator. Baud rate 115200
	Output Parameter Files		Port Auto v Connect
	Use an external trigger		To Enable Monitoring Show monitoring views and enable a monitoring function.

Figure 5-1. Select a "Start Tuning"

 At the start of the first debug session, e² studio may display a message indicating that is will switch to the Debug perspective. Click the **Remember my decision** check box and **Switch** to continue the Debug process and the QE for Capacitive Touch automatic tuning.

👩 Conf	firm Perspective Switch	×
?	This kind of launch is configured to open the Debug perspective when it suspends. This Debug perspective supports application debugging by providing views for displaying the debug stack, variables and breakpoints. Switch to this perspective?	
Rem	nember my decision Switch No	

Figure 5-2. Switch Perspective

3. The QE for Capacitive Touch automatic tuning will now begin. Please read the tuning dialog windows carefully as they will guide you through the tuning process. An example screen is shown below. Typically, no interaction is required during the initial tuning process steps.

Automatic Tuning Processing
1/8: QE is beginning the tuning process. During the tuning process, please do not touch the sensors on the target board until instructed by the QE Tuning Program.
Cancel Help

Figure 5-3. Tuning Dialog Window



4. After several automated steps, you will arrive at the dialog box with information like what is shown below. This is the Touch sensitivity measurement step of the tuning process. Press using normal Touch pressure on the sensor being indicated in the dialog box (S1, TS00). When you press, the bar graph will increase to the right and the Touch counts go numerically up. While holding that pressure, press any key on the PC keyboard to accept the measurement.

Automatic Tuning Processing	×
5/8: QE will now measure touch sensitivity for (S1, TS00 @ config01). In this step please use normal touch pressure on the sensor for once. Press any key on the PC keyboard to accept the sensitivity measurement.	
S1, TS00 @ config01: 16365	
Cancel Help	

Figure 5-4. Measure Touch Button S1

5. Repeat the process for capacitive button S3 and TS01.

Automatic Tuning Processing	×
6/8: QE will now measure touch sensitivity In this step please use normal touch press key on the PC keyboard to accept the ser	ssure on the sensor for once. Press any
S3, TS01 @ config01: 16356	
	Cancel Help

Figure 5-5. Measure Touch Button S3

6. Next, we will adjust the slider. Move your finger across the slider 3-4 times. After that, press any key on the PC keyboard to accept the measurement.



Figure 5-6. Repeat Measurement 3 Times



7. Once complete, you will see a screen like what is shown below. This is the detection threshold that is used by the middleware to determine if a Touch event has occurred.

				tried. If there are con Renesas application			
Select the target	Method	Kind	Name	Touch Sensor	Threshold	Overflow	Warning / Error
	config01	Slider	S2	TS02, TS04, TS05, TS10, TS11	773		
	config01	Button	S3	TS01	940		
	config01	Button	S1	TS00	1024	J	
Retry Continue	the Tuning	Process	<u> </u>			Cancel	Help

Figure 5-7. Threshold for Touch Event

 Click the Continue the Tuning Process button in the dialog box shown. This will exit the tuning process and disconnect from the Debug session on the target. You should return to the default CapTouch Main (QE) screen in the e² studio IDE.

indicated, the	se senso	ors can	be re	ow complete. If overf tried. If there are con Renesas application	tinued o	verflows	or
Select the target		Kind	Name	Touch Sensor	Threshold	Overflow	Warning / Error
	config01	Slider		TS02, TS04, TS05, TS10, TS11			
	config01	Button		TS01	940		
	config01	Button	S1	TS00	1024		
Retry Continue	the Tuning	Process]			Cancel	Help

Figure 5-8. Select a "Continue the Tuning Process"



9. After the tuning process is completed, the default view of **the CapTouch Main (QE)** presented here will be the results of the tuning process for the sensors in the configuration. This gives the user a quick way to examine the tuning results.

Workflow D	iagram								Ď
1. Prepa	ration	ſ		2. Tuning		3. Coding	4. Mon	itoring	
Prepare a pr interfaces.	roject that uses			QE will automatically processing for each t	perform tuning	Implement a program using the touch interfaces.		check a behavior of and make fine ad	touch
Select the synergy To Prepa Select or c configurat synergy	t a Project target project _CapTouch are a Configur create a touch in ion. cCapTouch.tifc	ration nterface cfg	>	Enable ad To Output Parar Output parameter f result. Output Par Specify ar Use an ex Use diagn	In the dialog. Iuning vanced tuning neter Files ameter Files ameter Files a output folder ternal trigger	To Show Code Implement a program that periodically scans the status of the touch sensor in the main() function. Show Sample	Launch project To Cor Enable t commur emulato Baud Port		target ogram.
Tuning Ge Touch I/F C	esture Configuration: :	synergy_(CapTou	ch					>
	Kind	Name			Parasitic Capacitance[F] Sensor Drive Pulse Frequency[MHz]	Threshold	Scan Time[ms]	Overflow
Method			TS02, 1	TS04, TS05, TS10, TS11	-	-	773	-	None
config01		S2			17.587	0.943 (BASE: 1.0)	-	0.592	-
config01 config01	Slider TS	(S2)	TS02					0.500	
config01	Slider TS		TS02 TS04		19.338	0.943 (BASE: 1.0)	-	0.592	
config01 config01	Slider TS Slider TS	(S2)				0.943 (BASE: 1.0) 0.943 (BASE: 1.0)	•	0.592	-
config01 config01 config01	Slider TS Slider TS Slider TS	(S2) (S2)	TS04		19.338				•
config01 config01 config01 config01	Slider TS Slider TS Slider TS Slider TS	(S2) (S2) (S2)	TS04 TS05		19.338 24.104	0.943 (BASE: 1.0)	•	0.592	- - -
config01 config01 config01 config01 config01 config01	Slider TS Slider TS Slider TS Slider TS Slider TS	(S2) (S2) (S2) (S2)	TS04 TS05 TS10		19.338 24.104 21.267	0.943 (BASE: 1.0) 0.943 (BASE: 1.0)		0.592 0.592	- - - None

Figure 5-9. Tuning Result



5.2 Executing the "Output Parameter Files"

1. Output the tuning parameter files. Click the button Output Parameter Files.



Figure 5-10. Output Parameter Files

2. Look at the **Project Explorer** window and you will see that files have been added. These contain the needed tuning information to enable Touch detection using.

陷 Project Explorer 🗙	🖻 🔁 🏹 🕴 🗖 🖬	
✓		
> 🎼 Binaries		
> 🔊 Includes		
v 📴 ge gen		
> c qe_touch_config.c		
> h ge_touch_config.h		
> h) qe_touch_define.h		
> 🖉 src		
> 🔑 synergy		
> 🗁 Debug		
> 🔁 QE-Touch		
> 🗁 script		
> 🗁 synergy_cfg		
🔅 configuration.xml		
R7FS5D97E3A01CFC.pincfg		
S5D9-PK.pincfg		
📄 synergy_CapTouch Debug.jlinl	c	
📄 synergy_CapTouch Debug.laur	nch	
synergy_cfg.txt		
> ⑦ Developer Assistance		
📃 Console 🔀	🖹 🚮 📴 🛃 🚍 🖛 🗂 🖛 🗖	
QE for Capacitive Touch		
2022/03/03 11:15:07	^	
Succeeded to output the paramet	ter files.	
C:/Synergy/workspace/synergy_Co	apTouch/qe_gen/qe_touch_define.h	
	apTouch/qe_gen/qe_touch_config.h	
C:/Synergy/workspace/synergy_Ca	apTouch/qe_gen/qe_touch_config.c	
<		

Figure 5-11. Created Output Parameter Files



6. Coding on CapTouch Main

1. To implement application code to scan and report the state of the touch sensor, click the button **Show Sample** in the CapTouch Main (QE) e² studio IDE



Figure 6-1. Select a "Show Sample"

2. A new menu window will open with shows the sample code in text. Click the button Output to a File.

💽 Show Sample Code			×
Sample code of main() function:			
* * * FILE : qe_sample_main.c * DATE : 2021-11-17 * DESCRIPTION : Main Program * * NOTE:THIS IS A TYPICAL EXAMPLI * * #include "qe_touch_config.h" #define TOUCH_SCAN_INTERVAL_E	***************************************		^
void qe_touch_main(void); uint64_t button_status; #if (SF_TOUCH_CTSU_CFG_NUM_SI uint16_t slider_position[SF_TOUCH_ #endif #if (SF_TOUCH_CTSU_CFG_NUM_W uint16_t wheel_position[SF_TOUCH #endif	CTSU_CFG_NUM_SLIDERS]; /HEELS != 0)		~
Copy to the Clipboard	Output to a File	Show the Application Note	
Copy to the Cipboard	output to a rife		

Figure 6-2. Output to a sample code



3. Created a new project file that describes the sample code. In the Project Explorer window and you will see that **qe_touch_sample.c** files have been added.



Figure 6-3. Created a qe_touch_sample.c

4. Open the hal_entry.c, add the qe_touch_main() of the sample main program.



Figure 6-4. Adding a code

5. Building the code should result in no errors or warnings for this simplified application example.



Figure 6-5. Result in no errors



7. Debugging

This chapter is debugging on detects the S3 Capacitive Touch Button. Add a **button_status** to the watch expression, verify that the detected value is correct. Step 2-7 to set up Watch Expression, resume in step 8, and detect in step 9.

- 1. Start a Debug session by clicking the Bug icon in the upper left-hand corner of e² studio. A Debug session will commence.
- 2. Open the **qe_touch_sample.c** file.

Find the **button_status** on **g_qe_touch_instance_config01.p_api->dataGet()** function. Right-click the **button_status**, and click the **Add Watch Expression...**.

	#endi	F			
			4	Undo	Ctrl+2
				Revert File	
	⊖void {	qe_touch_main(void)		Save	Ctrl+
		sp_err_t err;		Open Declaration	F
				Open Type Hierarchy	F
		* Open Touch <u>middleware</u> */		Open Call Hierarchy	Ctrl+Alt+
000046a0 000046b4		rr = g_qe_touch_instance_config01.p_api->open(g_qe_touch_instance_config01.p_ctrl, g_qe_touch_inst f (<i>SSP_SUCCESS</i> != err)	tance_c	Quick Outline	Ctrl+
0000+00+	{	(33F_30CC233 := EIT)		Quick Type Hierarchy	Ctrl+
000046f4		while (true) {}		Explore Macro Expansion	Ctrl+
	3			Toggle Source/Header	Ctrl+Ta
		* Main loop */		Open With	
	⊖ w {	hile (true)		Show In	Alt+Shift+V
		/* for [CONFIG01] configuration */	of	Cut	Ctrl+
000046b6		<pre>err = g_qe_touch_instance_config01.p_api->scanStart(g_qe_touch_instance_config01.p_ctrl);</pre>		Сору	Ctrl+
000046c6	Θ	if (SSP_SUCCESS != err)		Paste	Ctrl+
000046f2		<pre>while (true) {}</pre>		Quick Fix	Ctrl+
000046c8		<pre>while (0 == g_qe_touch_flag) {}</pre>		Source	Alt+Shift+
000046ce		g_qe_touch_flag = 0;		Refactor	
000046da		err = g_qe_touch_instance_config01.p_api->dataGet_g_qe_touch_instance_config01.p_ctrl, &button	status	Declarations	
	Θ	if (SSP_SUCCESS == err)		References	
				Search Text	
		/* TODO: Add your own code here. */ }		Build Selected File(s)	Ctrl+Alt+Shift+
				Clean Selected File(s)	Ctrl+Alt+Shift+
		/* FIXME: Since this is a temporary process, so re-create a waiting process yourself. */			
000046de		R_BSP_SoftwareDelay(TOUCH_SCAN_INTERVAL_EXAMPLE, BSP_DELAY_UNITS_MILLISECONDS);		Resource Configurations	
	}			Step Into Selection	Ctrl+F
	1		=>	[Run to Line	Ctrl+
	<			Move to Line	
onsole 🖾 🐰	11 Registe	s 🗓 Debug Shell 📳 Problems 🙀 Debugger Console 🆓 Smart Browser 📋 Memory	<u> </u>	Resume at Line	
		nesas GDB Hardware Debugging]	X+) =?	Add Watch Expression	
GDBServ	ver endi	an : little		Run As	

Figure 7-1. Add Watch Expression

3. Click OK.



Figure 7-2. Add expression to watch

4. button_status in the Expression window list is added.

(x)= Variables 💁 Breakpoint	ts 🛋 Modu	les Project Explorer 🕨	Expressions ×	e Eve
Expression	Туре	Value	Address	
(×)= button_status	uint64_t	2436570548560480285	0x1ffe05d8	
🐈 Add new expression				

Figure 7-3. Added in the "Expression window list"



5. Right-click the button_status, Select to the Real-time Refresh > Real-time Refresh Interval.

Expres	sion	Туре	Value			Address	
(×)=	button_status	uint64_t	243657	05485	60480285	0x1ffe05d8	
	Select All		Ctrl+A	L_			
	Copy Expressions		Ctrl+C	⊢			
ж	Remove			F			
8	Remove All						
	Number Format		>				
	Add Expression Group	•	>	L.			
	Find		Ctrl+F	F			
	Show Details As		>	E			
4	Add Watch Expression	ı		⊢			
_	Disable			E			
	Enable						
	Edit Watch Expression						
66	Add Watchpoint (C/C	++)		⊢			
<i>ø</i> 0	Cast To Type			⊢			
×[]	Display As Array			F			
<u>_</u>	Enable Real-time Refr	esh					
	Real-time Refresh		>	Re	Add To Cha	rt	ł
x+y ≡?	Watch			R.	Start Record	ling	
_					Real-time R	efresh Interval	
					Edit Express	ion Value	

Figure 7-4. Select "Real-time Refresh Interval"

6. Set is the Interval-time 100milliseconds. click OK.

🕲 Real-time Refresh Interval	×
Milliseconds 100	÷
OK Cancel	

Figure 7-5. Set "Real-time Refresh Interval"



7. Right-click the button_status. Select Enable Real-time Refresh.

Expression		Туре	Value		Address	
(×)= button_status		uint64 t	2//265705//8560		0x1ffe05d8	
🐈 Add new expre		elect All		Ctrl+A		
		Copy Expressio	ns	Ctrl+C		
	IK F	lemove				
3	🕺 F	lemove All				
	1	lumber Forma	t	>		
	4	dd Expression	Group	>		
	F	ind		Ctrl+F		
	S	how Details As	;	>		
4	b 4	dd Watch Exp	ression			
	-)isable				
	F	nable				
	F	dit Watch Exp	ression			
64		dd Watchpoir				
		Cast To Type	(c, c · ·)			
		isplay As Arra				
		nable Real-tim				
্		eal-time Refre		<u> </u>	_	
			sn	,		
X+ =2	^{≠y} ∖	Vatch				

Figure 7-6. Enable Real-time Refresh

8. Click the **Resume** button located approximately in the middle of the e² studio menu bar to continue code execution.

🕲 workspace - synergy_CapTouch/synergy/ssp/src/bsp/crmis/Device/RENESAS/SSDR/Source/startup_SSDR.c - e*studio					
File Edit Source Refactor Navigate Search Project Renesas Views Run Window Help					
🔦 🗱 🔳 🎋 Debug 🗸 🕅 synergy_CapTouch Debug 🗸 🎲 🗄 🖄 🐨 🖓 🕶 📸 🐂 🤅 🎕	D = N 3. 3 12 14 =, 21 34 4 + 9 9 16 16. D 2 32 21 30 0 1 2 89 20 - 1.				

Figure 7-7. Click the "Resume"

9. Press S3 on the board, which was configured as S3 in section 3.2. When pressed, a "2" will appear for the capacitive button in the Expressions window, indicating a binary indication of touch.

(x)= Variables	🛋 Module	s 陷 Project Explorer 🚥	Expressions 🗙 🥵 Eve
Expression	Туре	Value	Address
🎄 button_status	uint64_t	2	0x1ffe05d8
💠 Add new expression			

Figure 7-8. See S3 values



8. Monitoring on CapTouch Main

This chapter is monitoring on detects capacitive button S3. Monitor detailed values for The Capacitive Touch Interface using the Monitoring feature.

8.1 Monitoring on detects capacitive button S3

Monitors the detection status of the Capacitive Touch Interface.

1. Open the Monitoring view from **Show Views** of CapTouch Main (QE).

1. Preparation	2. Tuning	3. Coding	4. Monitoring
Prepare a project that uses the touch interfaces.	QE will automatically perform tuning processing for each touch sensor.	Implement a program using the touch interfaces.	You can check a behavior of touch interfaces and make fine adjustments.
To Select a Project Select the target project. synergy_CapTouch To Prepare a Configuration Select or create a touch interface configuration. synergy_CapTouch.tifcfg Modify Configuration	Image: Connect Your target Board Connect your target board and PC via an emulator. Image: Connect Your Structions in the dialog. Image: Connect Your Your Your Your Your Your Your Your	To Show Code Implement a program that periodically scans the status of the touch sensor in the main() function. Show Sample	To Launch Debug (via Emulator) Launch debugging for your target project and execute the program. To Connect UART Enable a monitoring function via serial communication, if you do not use an emulator. Baud rate 115200 Port Auto Connect To Enable Monitoring Show monitoring views and enable a monitoring function.

Figure 8-1. Select "Show Views"

2. It may be necessary to drag the pane up for better viewing, however you should see the **CapTouch Board Monitor (QE)** pane appear like the image below.



Figure 8-2. CapTouch Board Monitor (QE) window



3. Click the **Enable Monitoring** button. The dialog text will change to Monitoring: Enabled.



Figure 8-3. Enable Monitoring

4. Touch the capacitive button S3 on the PK-S5D9 board. The **CapTouch Board Monitor (QE)** will show a touch with a finger image on the button like the below image.

CapTouch Board Enable Monitoring Touch I/F:	Monitor (QE) X Monitoring: Enabled, Communication Statu	us: Connecting via OCD emulato	
			~
¢	52 53 51 51		~

Figure 8-4. Show a touch with a finger image

5. To see a graphical representation of the 'touch counts' from the board, use the CapTouch Status Chart (QE). Using the pulldown, select "S3 @ config01".

						5	
Touch I/F: S3 @ cont	fig01 🗸 🗸	Sync a selection					
I/F Type: Button(self),	Channel(s): TS01						
Count Value:	16588 Reference Value:	16605 Threshold:	940 Differe	nce: -17]		
Start Data Collec	tion						
Noise [NT]:	Average [NT]:	Minimum:	Maxim	um:]		
Noise [T]:	Average [T]:	Signal:	SNR:				





6. The graph will begin to display running values. Capacitive button S3 on the board and you should see the "touch counts" show as a step change on the running graph. The GREEN line is the touch "Threshold", which the middleware uses to determine whether a button is actuated/touched. The RED BELT at the bottom of the graph is a visual indication to the user that the "touch counts" have crossed above the threshold and a touch is detected.



Figure 8-6. Display Running Values for Capacitive Button S3 on the Board



8.2 Displaying and measuring standard deviation

This setting should only be set when displaying and measuring standard deviation. Also, if the measured standard deviation is unexpected value, re-tune from Chapter 5.1.

1. Next, measure standard deviation. Click the **Start Data Collection** button. Don't touch the electrode as this will collect data of touch-off state. The green bar is the data collection rate. When the green bar goes all the way to the right, the data collection of touch-off state is complete.

Touch I/F: β3 @ config01	✓ □ Sync a selection				
I/F Type: Button(self), Channel(s): TS01					
Count Value: 15440	Reference Value: 15422 Threshold: 627 Difference: 18				
Start Data Collection					
Noise [NT]:	Average [NT]: Minimum: Maximum:				
Noise [T]:	Average [T]: Signal: SNR:				

Figure 8-7. Touch-off Start Data Collection

2. Click the **Stop Data Collection** button, when the green bar goes all the way to the right.

Touch I/F: S3 @ config01	~	Sync a se	ection			
I/F Type: Button(self), Channe	el(s): TS01					
Count Value: 1542	7 Reference Value	15429	Threshold:	627	Difference:	-2
Stop Data Collection						
Noise [NT]:	1 Average [NT]:	15423	Minimum:	15393	Maximum:	15459
Noise [T]:	Average [T]:		Signal:		SNR:	

Figure 8-8. Touch-off Stop Data Collection

3. Next, Touch the electrode as this will collect data of touch-on state. Click the **Start Data Collection** button while touching the electrode.

I/F Type: Button(self), Channel(s): TS01
Count Value: 15427 Reference Value: 15429 Threshold: 627 Difference: -2
Start Data Collection
Noise [NT]: 11 Average [NT]: 15423 Minimum: 15393 Maximum: 15459
Noise [T]: Average [T]: Signal: SNR:

Figure 8-9. Touch-on Start Data Collection

4. Click the **Stop Data Collection** button, when the green bar goes all the way to the right. The SNR is displayed when data collection is complete.

Touch I/F: S3 @ config01	~	Sync a selection			
I/F Type: Button(self), Channel(s): TS01				
Count Value: 16619 Stop Data Collection	Reference Value:	15425 Thresh	old: 627	Difference:	1194
Noise [NT]: 11	Average [NT]:	15424 Minimu	ım: 15393	Maximum:	15462
	Average [N1]: Average [T]:	16590 Signal:			13402
Noise [1]: 25	Average [1]:	16590 Signai:	1100	SINK:	17

Figure 8-10. Touch-on Stop Data Collection



Website and Support

Visit the following URLs to learn about key elements of the Synergy Platform, download components and related documentation, and get support.

Synergy Platform MCUs Synergy Software Package Software add-ons SSP Components MCU Components Kits	www.renesas.com/renesas-synergy-platform-mcus www.renesas.com/synergy/ssp www.renesas.com/synergy/addons www.renesas.com/synergy/ssp-components www.renesas.com/synergy/renesas-synergy-platform-mcus www.renesas.com/synergy/kits
Synergy Solutions Gallery:	
Partner projects	www.renesas.com/synergy/partnerprojects
Application projects	www.renesas.com/synergy/applicationprojects
Self-service support resources:	
Knowledgebase	www.renesas.com/synergy/knowledgebase
Forums	www.renesas.com/synergy/forum
Training	www.renesas.com/synergy/training
Videos	www.renesas.com/synergy/videos
Chat and web ticket	www.renesas.com/synergy/resourcelibrary



Revision History

		Descript	ion
Rev.	Date	Page	Summary
1.00	Aug.16.21	—	First release document
1.10	Mar.14.22	1	Update to Operating Environment
		—	Changed version SSP to v2.2.0
			Changed version QE for Capacitive Touch to v3.0.2



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 reseting 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

- 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.
- 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.
- This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
 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

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/.