RL78 Family

SHA Hash Function Library: Introduction Guide

Introduction

This document explains SHA Hash Function Library for the RL78 Family (hereafter referred to as "SHA Libraly") that depends on MCUs.

The SHA Libraly is the software library that processes HASH calculation for RL78 Family. Also it is designed in dedicated algorithm and fully-tuned up by assembly language.

The library included in this version of the application note can be combined with RL78/G24 FAA(Flexible Application Accelerator) to improve processing speed. For details, refer to 2.3, How to use library functions (When combined with RL78/G24 FAA).

For details of API functions, refer to Renesas Microcomputer SHA Hash Function Library: User's Manual(R20UW0101).

Target Device

RL78/G14, RL78/G23, RL78/G24

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.

Contents

1. Structure of product

This product contains the files listed in Table 1 below.

Table 1. SHA Library product files

Name	Description
sample program(r20an0211xx0202-rl78-sha)	
workspace <DIR>	
Document (doc) <DIR>	
English (en)	
r20uw0101ej0201-sha.pdf	User's manual
r20an0211ej0202-rl78-sha.pdf	Introduction Guide (this document)
Japanease(ja)	
r20uw0101jj0201-sha.pdf	User's manual
r20an0211jj0202-rl78-sha.pdf	Introduction Guide
libsrc <DIR>	Library source
sha <DIR>	SHA Library
src <DIR>	SHA Library source
sha1if.c	SHA-1 API function definition
sha256if.c	SHA-256 API function definition
sha384if.c	SHA-384 API function definition (Not supported by RL78)
shaif.h	Core part of API function
sha1.c	Core part of SHA-1 calculation
sha256.c	Core part of SHA-256 calculation
sha512.c	Core part of SHA-384 / SHA-512 calculation (Not supported by RL78)
r_sha_version.c	SHA-1/SHA-256 version file
include <DIR>	SHA Library header folder
r_sha.h	Rev.2.02 header file
r_mw_version.h	Version data header file
r_stdint.h	Typedef header file
CS+ <DIR>	CS+ project folder
sha_rl78_sim_sample <DIR>	Sample project for RL78/G23
src <DIR>	Source folder
main.c	Sample code
main.h	Sample code header file
libsrc <DIR>	Link to libsrc
smc_gen <DIR>	Smart configurator auto-generated folder
general	Common header file / source file storage folder
r_bsp	Initialization code register definition storage folder
r_config	Driver initialization config header storage folder
sha_rl78_sample_FAA <DIR>	Sample project for RL78/G24 FAA
src <DIR>	Source folder
main.c	Sample code
main.h	Sample code header file
libsrc <DIR>	Link to libsrc

smc_gen <DIR>	Smart configurator auto-generated folder
Config_FAA	FAA-related source file storage folder
general	Common header file / source file storage folder
r_bsp	Initialization code register definition storage folder
r_config	Driver initialization config header storage folder
r_pincfg	Symbolic name setting header storage folder for ports
e² studio <DIR>	e² studio project folder
CCRL	Sample project for CCRL
sha_rl78_sim_sample <DIR> Below omitted.	Sample project for RL78/G23 Below omitted.
sha_rl78_sample_FAA <DIR> Below omitted.	Sample project for RL78/G24 FAA Below omitted.
LLVM	Sample project for LLVM
sha_rl78_sim_sample <DIR> Below omitted.	Sample project for RL78/G23 Below omitted.
IAR	IAR project folder
sha_rl78_sim_sample <DIR> Below omitted.	Sample project for RL78/G23 Below omitted.

2. Product Specifications

2.1 API Function

SHA Library for the RL78 supports the following functions.

Table 2. SHA Library API Functions

API	Outline
R_Sha1_HashDigest[Note]	Generate a SHA-1 hash digest
R_Sha256_HashDigest	Generate a SHA-256 hash digest

Note: When combined with RL78/G24 FAA, this function is not supported.

2.2 How to use library functions

When using the library function, it is necessary to specify the file to be built as follows according to the API to be used. When combined with RL78/G24 FAA, refer to 2.3, How to use library functions (When combined with RL78/G24 FAA).

Table 3. File to be build

API	File
R_Sha1_HashDigest	sha1if.c, sha1.c, r_sha_version.c
R_Sha256_HashDigest	sha256if.c, sha256.c, r_sha_version.c

2.3 How to use library functions (When combined with RL78/G24 FAA)

FAA (The Flexible Application Accelerator) is an application accelerator employing a Harvard architecture that was developed by Renesas Electronics Corporation. Using the FAA for SHA hash operation processing boosts the processing speed of the SHA Library[Note].

Note: When combined with RL78/G24 FAA, only SHA-256 is supported.
Note: When combined with RL78/G24 FAA, only CC-RL compiler is supported.

When combined with FAA, generate code for SHA hash operation processing for FAA in the Smart configurator. Combine the generated code with the code in the libsrc folder included in this library package. In addition to the FAA SHA Library code, specify the code in Table 4 below as the build target.

Table 4. File to be build when combined with RL78/G24 FAA

API	File
R_Sha256_HashDigest	sha256if.c, r_sha_version.c

2.3.1 How to generate code

FAA SHA Library generates code using the Smart configurator.

For more information on how to operate the Smart Configurator, please refer to the following document.

- RL78 Smart Configurator User's Guide: e² studio (R20AN0579)
- RL78 Smart Configurator User's Guide: CS+ (R20AN0580)

(1) Add the **Flexible Application Accelerator** component (referred to below as the FAA component).
The character string specified for **Configuration name:** when adding the component will be reflected in the code names generated by the Smart Configurator. The initial value of the configuration name is **Config_FAA**.

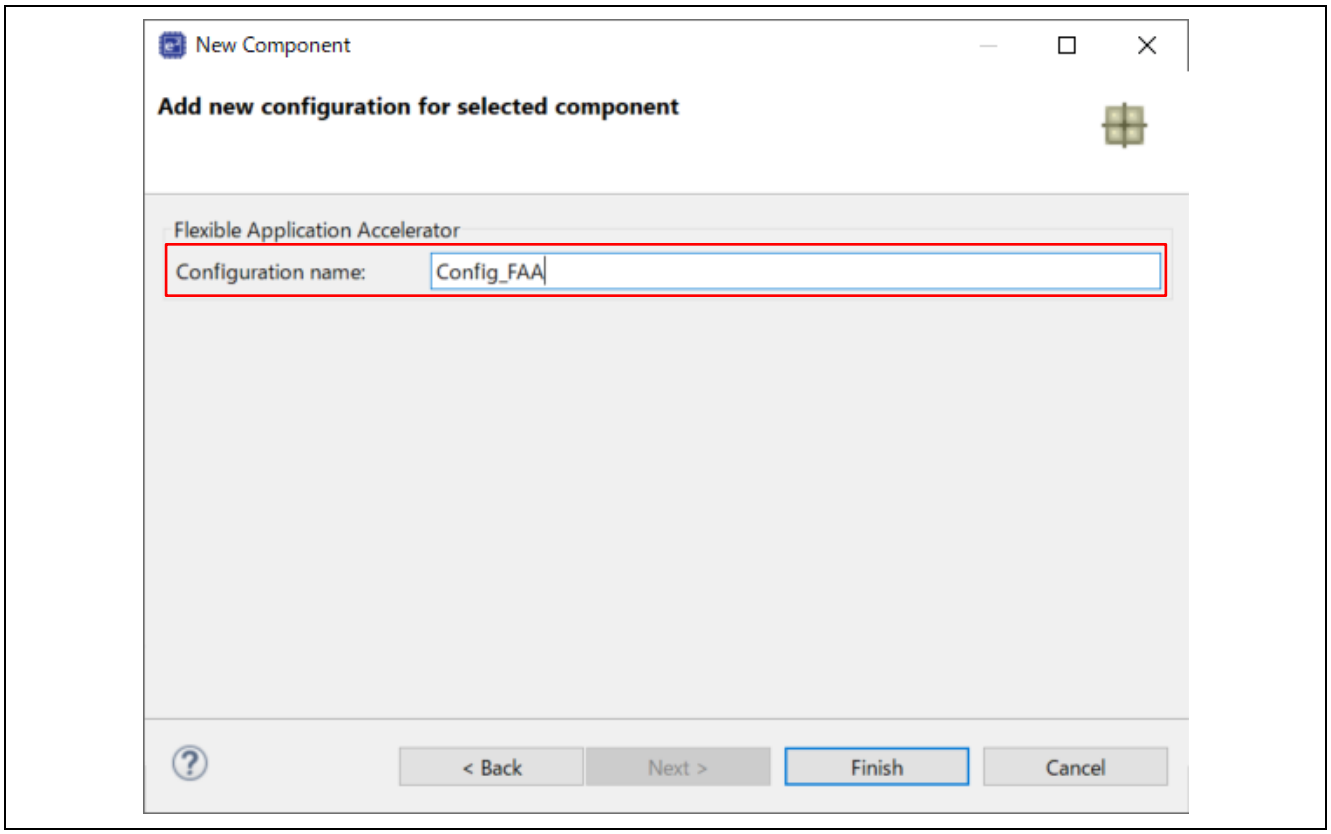

Figure 1. Adding a Component

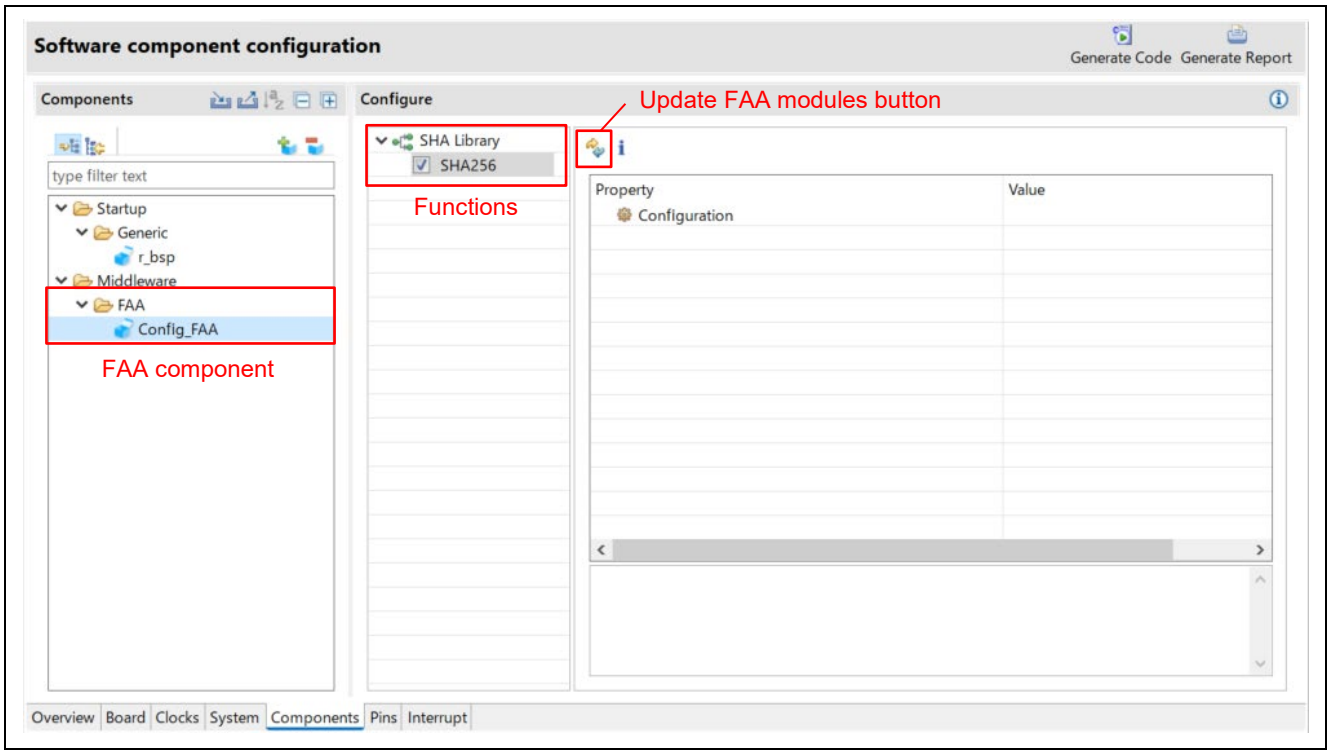

Figure 2. Software component configuration Window

(2) Download FAA SHA Library.
Click the **Update FAA modules** button to display the FAA modules download screen and select FAA SHA Library to download.

(3) Select **SHA256** in the function to perform code generation. The code is generated in **\src\smc_gen\Config_FAA**. For details on the generated code, refer to 2.3.3, Generated Code Details.

2.3.2 Build Settings

After generating code with the Smart Configurator, perform the following build settings before building.

(1) Add the files in Table 4 to the build target.

(2) Specify **R_CONFIG_FAA_SHA256** in the macro definition of the compiler's preprocessor.

2.3.3 Generated Code Details

The following is a detailed description of the code generated by the Smart Configurator.

Table 5. Generated Code Details

File[Note1]	Explanation
“XXX”_common.c	FAA common function C source file
“XXX”_common.h	FAA common function header file
“XXX”_common.inc	iodefine header file for FAA
“XXX”_sha256.c	SHA-256 calculation C source file for FAA
“XXX”_sha256.h	SHA-256 calculation header file for FAA
“XXX”_src.dsp	SHA-256 calculation assembler file for FAA

Note: 1. “XXX” in the function name represents the configuration name. The configuration name is specified in Smart Configurator when adding the FAA component. For details, refer to 2.3.1,.How to generate code

2.3.4 Error Code

In the FAA SHA Library, the following error code is added to the return value of the R_Sha256_HashDigest function.

For details of API functions, refer to Renesas Microcomputer SHA Hash Function Library: User's Manual(R20UW0101).

Table 6. Error Code

Symbol	Value	Explanation
R_SHA_ERROR_FAA_ALREADY_RUNNING	-4	The function was terminated without performing SHA hash operation because the FAA processor was already running.

2.4 Notes

- The following macro specifications cannot be used with RL78.
  __COMPILE_EMPHASIS_SPEED__

3. CC-RL

3.1 Development environment

Please use the same or a later version of the toolchain listed below:

- Integrated Development Environment:
- CS+ for CC V8.05.00
- e^2 studio 2021-04
- C compiler:
- CC-RL V1.09.00

3.2 ROM / RAM / Stack Size and Performance

The various sizes and performance when building with the following options are described for reference.

Compiler options

-cpu=S3 -memory_model=medium –Odefault

Link options

-NOOPtimize

Table 7. ROM, RAM Size

API	ROM size [byte]	RAM size [byte]
R_Sha1_HashDigest	1814	0
R_Sha256_HashDigest	3033	0

Table 8. Stack Size

API	stack size [byte]
R_Sha1_HashDigest	174
R_Sha256_HashDigest	96

Table 9. Performance

system clock = 32MHz

input message length[byte]	SHA-1 [us]	SHA-256 [us]
0	800	1,200
64	1,500	2,300
128	2,200	3,400
192	2,900	4,600
256	3,600	5,700

Note: Input message is 1 block with padding processing.

4. CC-RL(When combined with RL78/G24 FAA)

4.1 Development environment

Please use the same or a later version of the toolchain listed below:

- Integrated Development Environment:
- CS+ for CC V8.10.00
- e^2 studio 2023-07
- C compiler:
- CC-RL V1.12.01
- DSP assembler:
- FAA Assembler V1.04.02

4.2 ROM / RAM / FAACODE / FAADATA / Stack Size and Performance

The various sizes and performance when building with the following options are described for reference.

Compiler options

-cpu=S3 -memory_model=medium –Odefault

Link options

-NOOPtimize

Table 10. ROM, RAM, FAACODE, FAADATA Size

API	ROM size [byte]	RAM size [byte]	FAACODE [byte]	FAADATA [byte]
R_Sha256_HashDigest	1073	0	684	524

Table 11. Stack Size

API	stack size [byte]
R_Sha256_HashDigest	46

Table 12. Performance

system clock = 32MHz

input message length[byte]	SHA-256 [us]
0	6,00
64	1,100
128	1,600
192	2,000
256	2,500

Note: Input message is 1 block with padding processing.

5. IAR Embedded Workbench

5.1 Development environment

Please use the same or a later version of the toolchain listed below:

- Integrated Development Environment:
  IAR Embedded Workbench for Renesas RL78 version 4.21.1
- C compiler:
  IAR C/C++ Compiler for Renesas RL78 : 4.20.1.2260

5.2 ROM / RAM / Stack Size and Performance

The various sizes and performance when building with the following options are described for reference.

Compiler options

--core=S3 --code_model=far --data_model=near --near_const_location=rom0 -e -Oh

--calling_convention=v2

Table 13. ROM, RAM Size

library file name	ROM size [byte]	RAM size [byte]
R_Sha1_HashDigest	2,009	0
R_Sha256_HashDigest	3,283	0

Table 14. Stack Size

API	stack size [byte]
R_Sha1_HashDigest	184
R_Sha256_HashDigest	138

Table 15. Performance

system clock = 32MHz

input message length[byte]	SHA-1 [us]	SHA-256 [us]
0	2,500	5,300
64	5,000	10,600
128	7,300	15,800
192	9,700	20,900
256	12,100	26,100

Note: Input message is 1 block with padding processing.

6. LLVM

6.1 Development environment

Please use the same or a later version of the toolchain listed below:

- Integrated Development Environment:
  e2 studio 2022-01
- C compiler:
  LLVM for Renesas RL78 10.0.0.202203

6.2 ROM / RAM / Compiler option / Performance

The various sizes and performance when building with the following options are described for reference.

Compiler options

CPU Type : S3-core

Optimization Level : Optimize size (-Os)

Table 16. ROM, RAM Size

library file name	ROM size [byte]	RAM size [byte]
R_Sha1_HashDigest	2,731	0
R_Sha256_HashDigest	4,312	0

Table 17. Stack Size

API	stack size [byte]
R_Sha1_HashDigest	178
R_Sha256_HashDigest	104

Table 18. Performance

system clock = 32MHz

input message length[byte]	SHA-1 [us]	SHA-256 [us]
0	1,900	3,000
64	3,700	5,800
128	5,500	8,700
192	7,300	11,500
256	9,100	14,300

Note: Input message is 1 block with padding processing.

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Oct 16, 2012	—	First edition issued
1.01	Sep 30, 2014		Improved document. Fixed problem when input pointer is an odd address. Added support for the small model and the large model.
1.02	Apr 01, 2015	—	Supported IAR Embedded Workbench.
1.03	Jul 01, 2016	—	Supported CC-RL. Supported IAR Embedded Workbench 7.4(v2.21.1).
2.00	Apr 21, 2021	—	Changed the library provision form from Lib Format to C source
2.01	Jun 30, 2022	—	Supported LLVM.
2.02	Aug 01, 2023	—	Added library for RL78/G24 FAA.

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

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

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

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

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

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

Trademarks

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

Contact information

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