

# Firmware Integration Technology

## User's Manual

---

### Summary

Firmware Integration Technology (FIT) is a concept aimed at reducing the program development and software asset management burden on software developers, with an emphasis on simplifying the integration of peripheral function module drivers and easing migration between MCUs within the same product family.

This application note provides a general description of FIT modules.

### Target Device

RX Family

### Related Documents

- RX Family Board Support Package Module Using Firmware Integration Technology (R01AN1685)
- RX Family RX Driver Package\*<sup>1</sup>
- RX Smart Configurator User's Guide: CS+ (R20AN0470)
- RX Smart Configurator User's Guide: e<sup>2</sup> studio (R20AN0451)
- RX Smart Configurator User's Guide: IAREW (R20AN0535)
- RX Family Renesas FreeRTOS (R01AN4307)
- RX Family Amazon FreeRTOS download with e<sup>2</sup> studio (R20AN0543)
- RX family Manually Importing Firmware Integration Technology Modules (R01AN5967)

Note: 1. The R numbers of Renesas documents change with each revision. Refer to the RX Driver Package webpage for the R numbers corresponding to the latest revisions.

---

**Contents**

1. FIT Features .....	3
1.1 FIT Configuration.....	3
1.2 Board Support Package Module (BSP).....	4
1.3 RX Driver Package (RDP).....	4
1.4 FIT Module Demo (Sample Project).....	4
2. Tools and Software Supporting Use of FIT Modules .....	5
3. FIT Module File Structure .....	6
3.1 Documentation Folder (doc).....	7
3.2 Configuration Reference Folder (ref).....	7
3.3 Source Folder (src).....	7
3.4 Interface Header File (r_<module_name>_if.h) .....	7
3.5 readme.txt.....	7
3.6 Other Folders.....	8
4. Using FIT Modules .....	9
4.1 Downloading an Integrated Development Environment (IDE) .....	9
4.2 Downloading FIT Modules.....	9
4.3 Creating a FIT Project and Importing FIT Modules .....	9
4.4 Software Development.....	10
4.5 Creating a Project That Uses a RTOS .....	10
Revision History .....	11

## 1. FIT Features

FIT establishes clear rules for the following information related to sample code (middleware and drivers):

- MCU initial settings
- Target board definition method
- File structure
- Function names
- Common interface with user applications

By establishing rules for the above information, FIT makes it easy to incorporate sample code into user applications. In addition, since a common interface with user applications is used by peripheral function drivers and middleware that supports FIT, it is a simple matter to port your current applications when migrating from one MCU to another MCU within the same product family.

### 1.1 FIT Configuration

Each FIT module comprises a board support package (BSP), a peripheral function module, and a middleware module.

[BSP]

This module handles tasks such as MCU initial settings, clock settings, and board settings.

[Peripheral function module]

A driver for controlling a peripheral function of the MCU.

[Middleware module]

Middleware that implements functionality such as TCP/IP or file system.

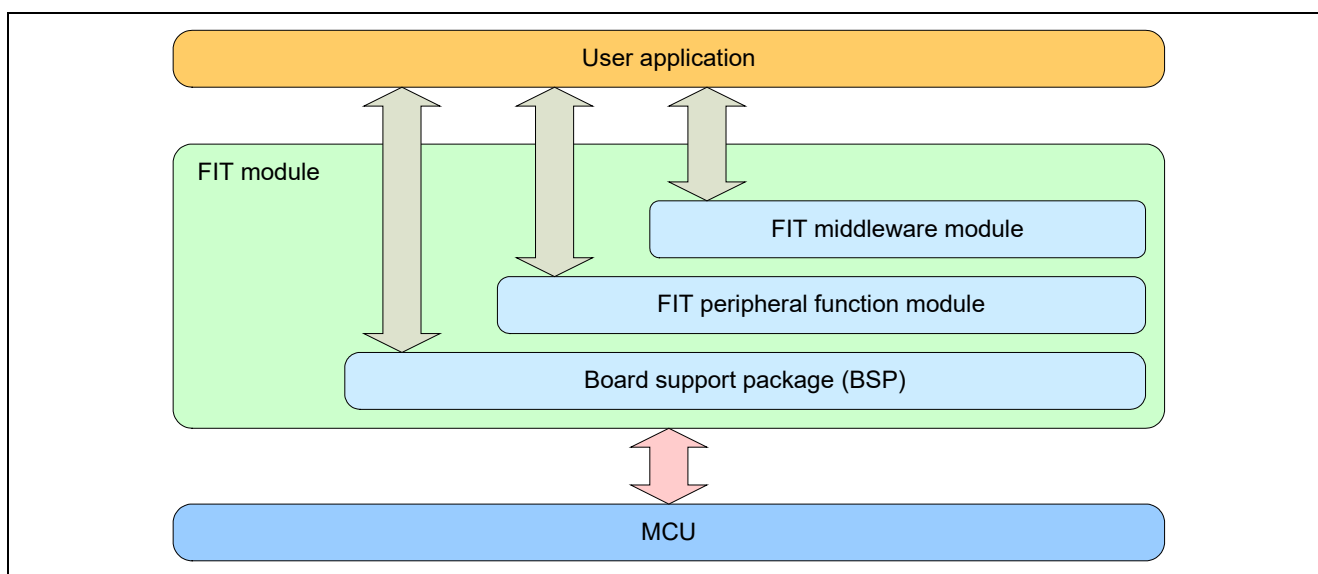


Figure 1.1 FIT Configuration

## 1.2 Board Support Package Module (BSP)

The board support package module (BSP) contains information essential to every FIT module and forms the basis for any project that makes use of FIT modules. The board support package module supplied by Renesas is contained in the `r_bsp` folder.

The code in `r_bsp` performs processing that takes place between a reset and the user application's `main()` function. It provides startup processing (initial pin settings, clock settings, interrupt level settings, I/O register definitions, initialization of resource locking functionality, etc.) specific to each platform. You will therefore need to modify the contents of `r_bsp` as necessary to match the requirements of the platform you are using.

The header file `platform.h` is used to select the platform. Configuration options for `r_bsp` are specified in `r_bsp_config.h`.

## 1.3 RX Driver Package (RDP)

The RX Driver Package bundles a variety of FIT modules (BSP, flash memory, timers, UART, A/D converter, USB, Ethernet, etc.) in a single software package.

The RX Driver Package is available for download on the RX Driver Package webpage.

[RX Driver Package webpage]

<https://www.renesas.com/jp/en/products/software-tools/software-os-middleware-driver/software-package/rx-driver-package.html>

## 1.4 FIT Module Demo (Sample Project)

Each FIT module is available for download individually on the web and many of them include demo software. The RX Driver Package consists of the FIT modules only; the demos are not included.

You can confirm the operation of a FIT module by installing the associated demo project in an integrated development environment (IDE). Either import the demo project manually or use the Smart Configurator function to import it. Refer to the RX Smart Configurator User's Guide for instructions on importing projects.

## 2. Tools and Software Supporting Use of FIT Modules

The e<sup>2</sup> studio integrated development environment is based on Eclipse, an open source software package that is widely used internationally, and it incorporates a Smart Configurator that performs three key functions: importing middleware for Renesas devices, generating driver source code, and making pin settings. These functions make it easy to import FIT modules into a project created in e<sup>2</sup> studio. In addition, FIT modules are also compatible with CS+ and the IDE published by IAR.

The IDE from IAR does not come with a Smart Configurator. If you wish to use the IDE from IAR, use the standalone version of Smart Configurator.

Table 2.1 lists tools and software that support the use of FIT modules. For details of the user procedure, refer to section 4, Using FIT Modules.

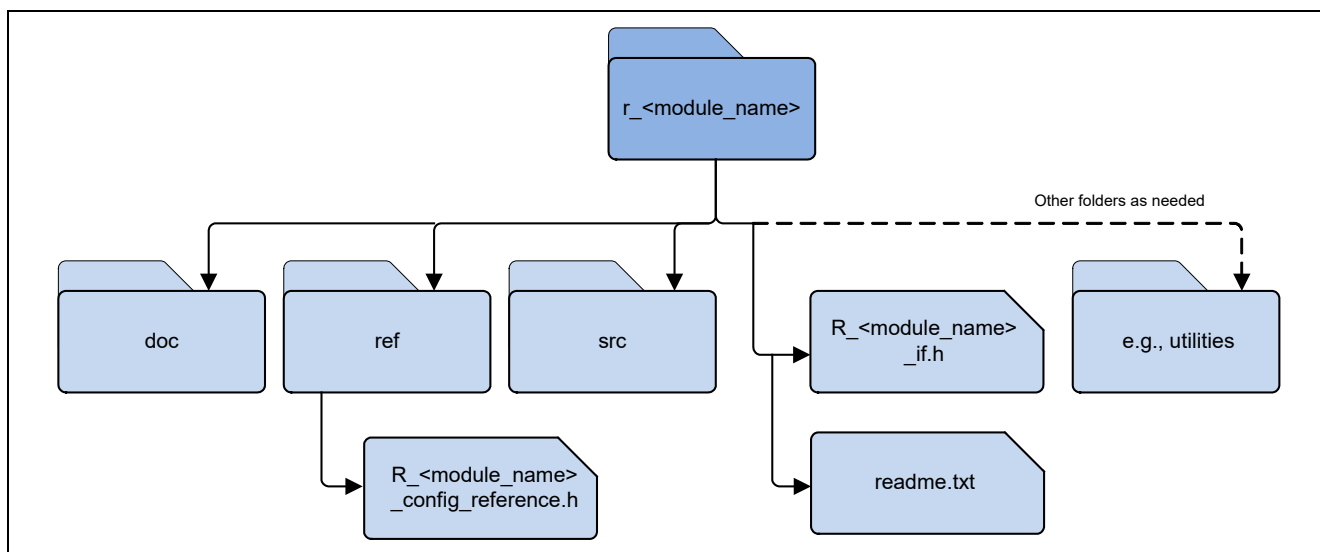
**Table 2.1 Tools and Software Supporting Use of FIT Modules**

	Supported Devices
	RX Family
Integrated development environments (IDEs)	e <sup>2</sup> studio (Eclipse) CS+ (Renesas IDE) IAR Embedded Workbench for Renesas RX (EWRX)
Configurator	Smart Configurator
Compilers	C/C++ Compiler for RX Family GCC for Renesas RX IAR C/C++ Compiler for Renesas RX version
RTOSes	Realtime OS for RX Family (μITRON specification) [RI600V4, RI600PX] FreeRTOS Amazon FreeRTOS

### 3. FIT Module File Structure

FIT rules define the file structure for all modules. Having common directory structures and file naming conventions makes it easier for users to quickly find the files they need.

The file structure is shown below.



**Figure 3.1 FIT Module File Structure**

Examples of r\_<module\_name> include:

- r\_bsp (Renesas Board Support Package module)
- r\_s12ad\_rx (A/D converter driver module)
- r\_flash\_rx (Flash memory programming driver module)

Many module names end with a suffix (e.g., \_rx), indicating the specific family of products supported. When the suffix is the name of a product (e.g., rx130), this indicates that the module supports that specific product.

The various folders and files are explained in more detail in the following sections.

### 3.1 Documentation Folder (doc)

This folder contains documentation related to the module.

### 3.2 Configuration Reference Folder (ref)

This folder contains a reference version of the module's configuration file. When a FIT module is imported by using Smart Configurator, the reference configuration file is copied to the `r_config` folder and the suffix `_reference` is removed from the file name. Putting all the configuration files in the same folder makes them easy to locate and allows the use of a single include path at setup.

### 3.3 Source Folder (src)

All MCU source code for a FIT module is stored in the `src` folder. There can be multiple folders within this folder if needed. A FIT module that supports multiple MCUs will have a `targets` folder that contains multiple folders with different versions of the source code for different MCUs or peripheral modules.

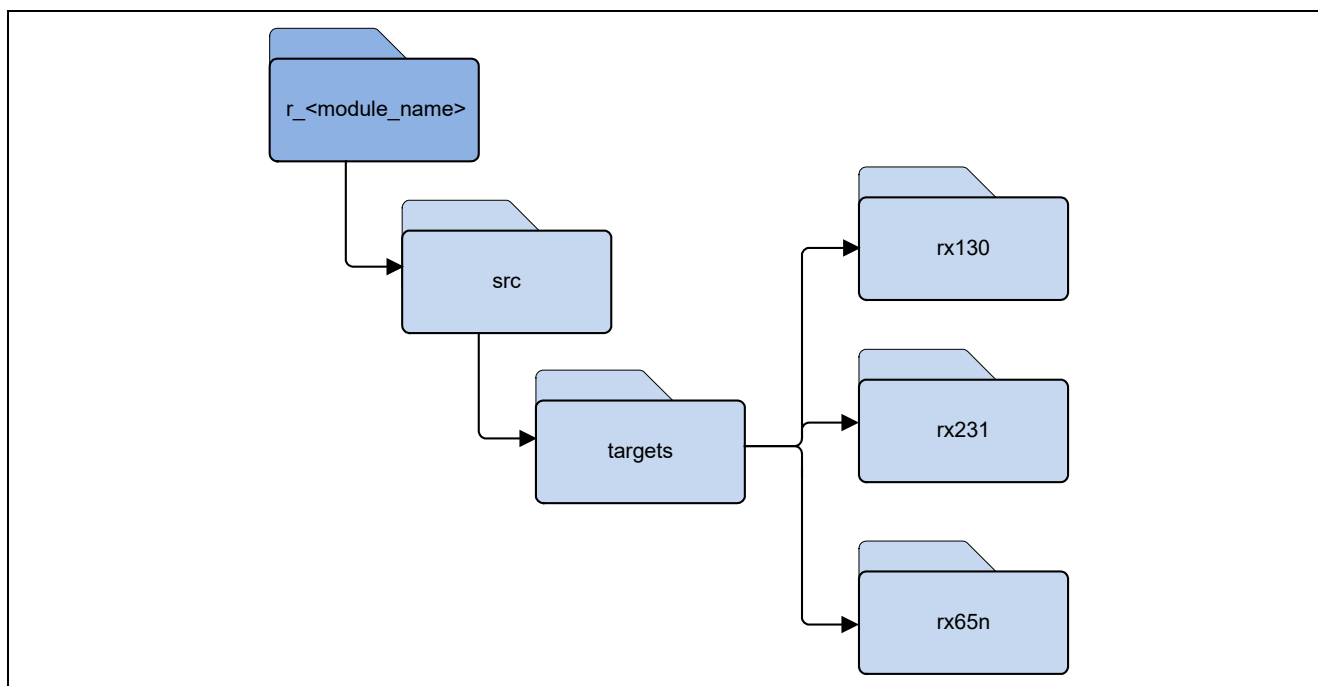


Figure 3.2 FIT Module with targets Folder

### 3.4 Interface Header File (`r_<module_name>_if.h`)

In order to use a FIT module with a project, you need to include the interface header file in your code. The interface header file contains the API function prototype declarations, data structures, and macro definitions required to use the FIT module. The name of the interface file consists of the module name with `_if.h` appended to it.

### 3.5 `readme.txt`

This file contains information about the FIT module.

### 3.6 Other Folders

In Figure 3.1 the dotted line indicates a folder that may be added optionally. The utilities folder is an example of such a folder.

For example, a module might come with code that needs to be run on a PC, and this code could be included with the module in an extra folder. Alternatively, a module might come in library form, and the module could have a folder named lib which contains the library.



## 4. Using FIT Modules

The procedure for using FIT modules is described in this section.

### 4.1 Downloading an Integrated Development Environment (IDE)

e<sup>2</sup> studio and CS+ are available for download on the Renesas website, and the IAR IDE can be downloaded from the IAR website. The IAR IDE does not include functionality for importing FIT modules. If you wish to use the IDE from IAR, download the standalone version of Smart Configurator from the Renesas website.

Either the Renesas compiler or the GNU compiler can be used with e<sup>2</sup> studio. CS+ only supports use of the Renesas compiler. The IAR IDE supports use of the IAR compiler.

[Renesas website]

<https://www.renesas.com/jp/en/products/software-tools/tools.html>

[IAR website]

<https://www.iar.com/jp/iar-embedded-workbench/partners/renesas/>

### 4.2 Downloading FIT Modules

FIT modules are available for download on the Renesas website and from the webpages for development tools (e<sup>2</sup> studio or Smart Configurator).

To download FIT modules from the Renesas website, visit either the FIT webpage or the webpage of the MCU product you are using.

[FIT webpage]

<https://www.renesas.com/jp/en/products/software-tools/software-os-middleware-driver/software-package/fit.html#fit>

[RX Family products webpage]

<https://www.renesas.com/jp/en/products/microcontrollers-microprocessors/rx.html>

The RX Driver Package is available for download on the RX Driver Package webpage.

[RX Driver Package webpage]

<https://www.renesas.com/jp/en/products/software-tools/software-os-middleware-driver/software-package/rx-driver-package.html>

For information on how to manually import the downloaded FIT module, refer to Application Note "RX family Manually Importing Firmware Integration Technology Modules (R01AN5967)"

For the procedure for downloading FIT modules from the webpages of development tools, refer to the RX Smart Configurator User's Guide.

### 4.3 Creating a FIT Project and Importing FIT Modules

The Smart Configurator is used to create a FIT project and import FIT modules. It can be used both to import and to configure FIT modules. The Smart Configurator lets you use a GUI to make settings for clocks, peripheral functions, pins, interrupts, and more.

The procedure for creating a FIT project and importing FIT modules differs somewhat depending on the IDE used. For a detailed description of the procedure, refer to the RX Smart Configurator User's Guide.

## 4.4 Software Development

The steps after adding a FIT module to your project are described below.

1. Make sure to include `platform.h` and the module's interface header file in the source files accessing the module. The file `platform.h` is part of the board support package (`r_bsp`) and the interface header file is part of each peripheral FIT module.  
The file `platform.h` contains definitions used by user APIs, FIT modules, standard library functions, etc., so make sure to include it before the other header files.
2. Most FIT modules contain no processing for pin settings, so you should make pin settings yourself. The method of making pin settings and the timing of the settings differs depending on the FIT module. For details on pin settings, refer to the manual of the peripheral FIT module in question.
3. Call the API functions of the FIT module. Each FIT module has multiple API functions, and they can be used as needed. Most FIT modules are designed to allow use of Open, Close, Read, Write, and Control functions.
  - The Open function releases the peripheral module from the module stop state and performs initial settings.
  - The Close function halts operation of the peripheral module and puts it into the module stop state.
  - The Read function is used to obtain data from the module (e.g., reading the input level of an I/O port).
  - The Write function is used to send data to the module (e.g., specifying output for an I/O port).
  - The Control function is used to control the module in a particular way. If an API contains a large number of small functions, it can become difficult to keep straight what each individual function does. For this reason, `cmd` is provided as an argument, allowing control over functionality to be expanded without changing the module API. An example would be an I/O port driver that has commands that can be used to specify things like the pin mode or the output format.For specifications of API functions, refer to the manual of the FIT module in question.
4. FIT modules allow you to specify callback functions to detect interrupts. Callback functions enable user applications to detect interrupts.  
The callback function you specified in the FIT module is called when an interrupt request occurs. You just need to set the address of the specified function in the FIT module.  
When a callback function for an interrupt is called, the other interrupts that were pending when it was called are made to wait while the callback function runs. For this reason callback functions for interrupts must be small and should be used prudently.

## 4.5 Creating a Project That Uses a RTOS

If an RTOS is used it is necessary to make settings not only for any FIT modules but for the RTOS as well. For details, refer to the manual of the RTOS.

**Revision History**

Rev.	Date	Description	
		Page	Summary
1.00	Nov. 15, 13	—	First edition issued
2.00	Nov. 02, 19	—	Document updated to reflect the latest information
2.01	Aug. 31, 21	1	Added "RX family Manually Importing Firmware Integration Technology Modules (R01AN5967)" to related documents.
		5	Deleted the description of FIT configurator from "2. Tools and Software Supporting Use of FIT Modules"
		9	Added how to manually import downloaded FIT modules in "4.2 Downloading FIT module".

# 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](http://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/](http://www.renesas.com/contact/).