RX Family

Interface conversion module for Ethernet Driver and Embedded system M3S-T4-Tiny Firmware Integration Technology

Introduction

This application note explains the software information about to convert the Embedded TCP/IP Library M3S-T4-Tiny (T4) user defined functions to the RX Family Ethernet Driver Interface (This module or T4 Driver FIT module).

T4 supports Ethernet communication. T4 is divided TCP/IP process and Ethernet control, and user can custom the parts of Ethernet control. This module provides the source code fitting T4 to RX Family Ethernet driver.

For about T4, please refer to the following URL.

https://www.renesas.com/mw/t4

This module and T4 are provided as FIT Module. Please refer to the URL to understand FIT outline.

FIT: Firmware Integration Technology.

https://www.renesas.com/jp/ja/software-tool/fit

Target Device

RX Family

Target Compilers

- Renesas Electronics C/C++ Compiler Package for RX Family
- GCC for Renesas RX
- IAR C/C++ Compiler for Renesas RX

For details of the confirmed operation contents of each compiler, refer to "4.1 Confirmed Operation Environment".

Related Documents

- Firmware Integration Technology User's Manual (R01AN1833)
- RX Family Board Support Package Module Using Firmware Integration Technology (R01AN1685)
- RX Family Adding Firmware Integration Technology Modules to Projects (R01AN1723)
- RX Family Adding Firmware Integration Technology Modules to CS+ Projects (R01AN1826)
- RX Smart Configurator User's Guide: e² studio (R20AN0451)
- RX Smart Configurator User's Guide: CS+ (R20AN0470)
- RX Smart Configurator User's Guide: IAREW (R20AN0535)
- RX Family TCP/IP for Embedded system M3S-T4-Tiny Introduction Guide Firmware Integration Technology (R20AN0051)
- RX Family Ethernet Module Using Firmware Integration Technology (R01AN2009)
- RX Family System Timer Module Firmware Integration Technology (R20AN0431)
- RX Family CMT Module Using Firmware Integration Technology (R01AN1856)
- TSIP (Trusted Secure IP) Module Firmware Integration Technology (Binary version) (R20AN0548)
- TCP/IP for Embedded system M3S-T4-Tiny User's Manual (R20UW0031)
- TCP/IP for Embedded system M3S-T4-Tiny: Ethernet Driver Interface Specification (R20UW0032)

Contents

1. Overview

1.1 T4 Driver FIT Module

The T4 Driver FIT module can be used by being implemented in a project as an API. See section 2.12 Adding the FIT Module to Your Project for details on methods to implement this FIT module into a project.

1.2 Overview of the T4 Driver FIT Module

This module converts the user-defined function called by T4 to the API of the Ethernet driver module.

This module is a software that can combine T4 and Ethernet driver.

When using the device that equips TSIP (Trusted Secure IP), the T4 Driver FIT Rev.1.09 realizes more secure Initial sequence number (ISN) generation than before. We recommend using the device with TSIP from the viewpoint of enhancing security. Please refer to the following web page for specific devices.

https://www.renesas.com/software-tool/trusted-secure-ip-driver

This figure shows 4 cases of T4 software stack. The software stack depends on the following conditions:

- ITRON TCP/IP APIs Style or Socket APIs style
- With TSIP (Note) or Without TSIP

  Note: In case of using MCU equipped TSIP

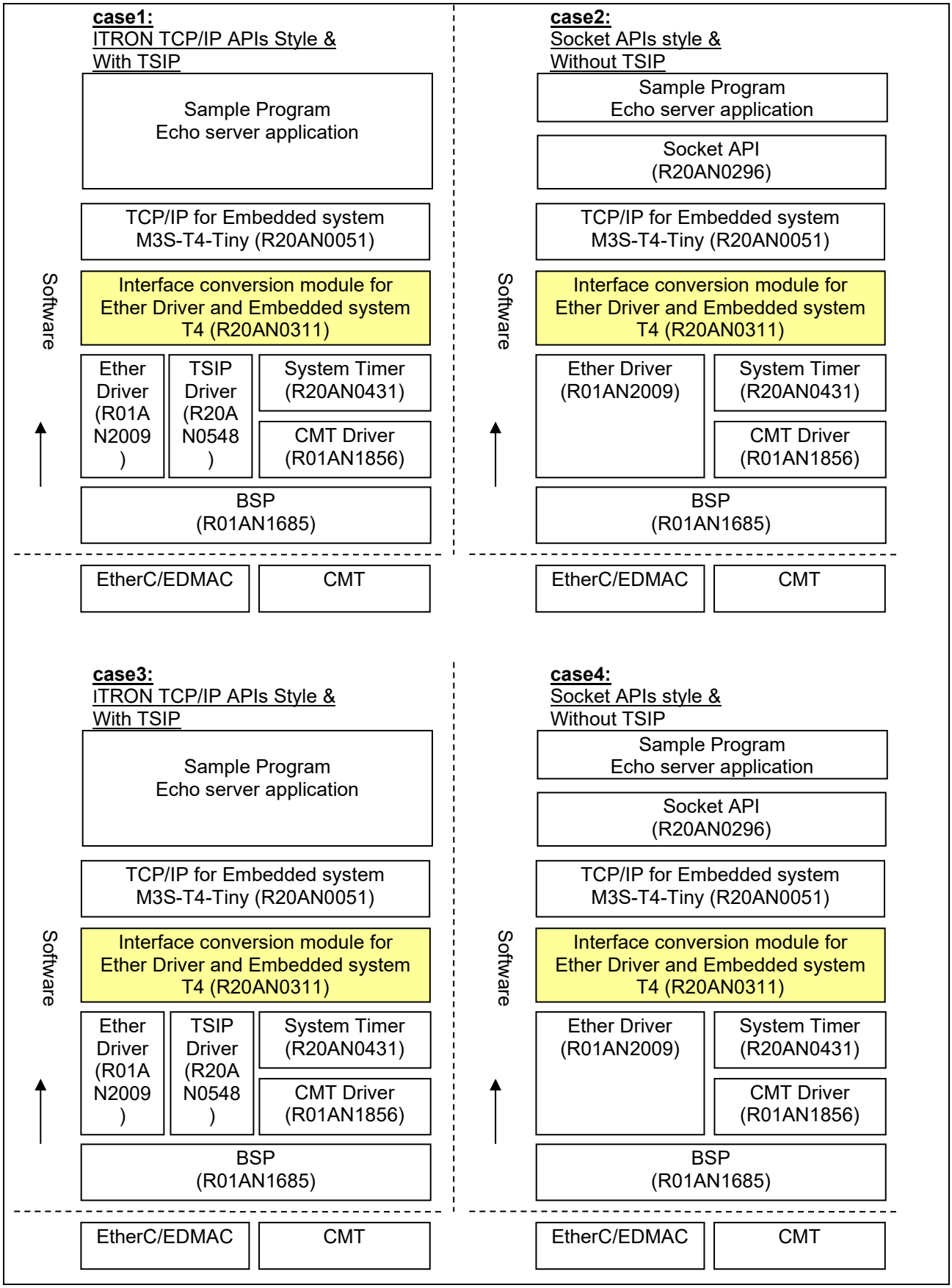

Figure 1 Software stack about T4 Driver FIT Modules

1.3 File Structure

This application note includes following files.

Table 1.1 File Structure 1

File/Directory name	detail
r20an0311xx0109-rx-t4-connectivity.zip	
r20an0311jj0109-rx-t4.pdf	Application Note (Japanese)
r20an0311ej0109-rx-t4.pdf	Application Note (English this document)
FITModules	
r_t4_driver_rx_v1.09.xml	xml file
r_t4_driver_rx_v1.09.zip	FIT modules

The folder to which the contents of r_t4_driver_rx_v1.09.zip is extracted will contain the files listed in table Table 1.2 Structure of the T4 Driver FIT Modules below.

Table 1.2 Structure of the T4 Driver FIT Modules

File/Directory name	Description
T4 Driver FIT Module (r_t4_driver_rx_v1.09.zip)	
T4 Driver config (r_config)	
r_t4_driver_rx_config.h	T4 Driver Config header
T4 Driver FIT Module body (r_t4_driver_rx)	
Document (doc)	
ja	
r20an0311jj0109-rx-t4.pdf	Application Note (Japanese)
en	
r20an0311ej0109-rx-t4.pdf	Application Note (English)
T4 config reference (ref)	
r_t4_rx_config_reference.h	T4 Config header(reference)
Source (src)	
t4_driver.c, etc.	T4 Source file
readme.txt	readme

2. API Information

This FIT module has been confirmed to operate under the following conditions.

2.1 Hardware Requirements

None.

2.2 Software Requirements

This driver is dependent upon the following FIT module:

- r_bsp
- r_ether_rx
- r_sys_time_rx
- r_t4_rx
- r_tsip_rx (In case of using the device with TSIP)

2.3 Supported Toolchain

This driver has been confirmed to work with the toolchain listed in 4.1, Confirmed Operation Environment.

2.4 Interrupt Vector

None.

2.5 Header Files

All API calls and their supporting interface definitions are located in r_t4_itcpip.h to T4 FIT Module (R20AN0051).

2.6 Integer Types

This project uses ANSI C99. These types are defined in stdint.h.

2.7 Configuration Overview

The configuration option settings of this module are located in r_t4_driver_rx_config.h. The option names and setting values are listed in the table below:

Configuration options in r_t4_driver_rx_config.h	
None	-

2.8 Code Size

The sizes of ROM, RAM and maximum stack usage associated with this module are listed below.

The ROM (code and constants) and RAM (global data) sizes are determined by the build-time configuration options described in 2.7, Configuration Overview.

The values in the table below are confirmed under the following conditions.

Module Revision: r_t4_driver_rx Rev.1.09

Compiler Version: Renesas Electronics C/C++ Compiler Package for RX Family V3.02.00

(The option of “lang = c99” is added to the default settings of the integrated development environment.)

GCC for Renesas RX 4.8.4.201803

(The option of “-std=gnu99” is added to the default settings of the integrated development environment.)

IAR C/C++ Compiler for Renesas RX version 4.14.1

(The default settings of the integrated development environment.)

Configuration Options: Default settings

ROM, RAM and Stack Code Sizes				
Device	Category	Memory Used		
		Renesas Compiler	GCC	IAR Compiler
RX65N (without TSIP)	ROM	**1,588 bytes**	**3,921 bytes**	**2,472 bytes**
	RAM	**280 bytes**	**280 bytes**	**279 bytes**
	STACK (Note 1, 2)	**32 bytes**	**-**	**556 bytes**

Note 1: in executing timer_interrupt().

Note 2: most of T4 driver’s functions are called by T4’s API, so the stack code size tends to vary depends on the stack code size calculation method of each compiler.

2.9 Parameters

Please refer to the T4 user's manual (r20uw0031ejxxxx_t4tiny.pdf) and Ethernet Driver Interface Specification (r20uw0032ejxxxx_t4tiny.pdf). These documents are included to T4 FIT Module (R20AN0051). The structure is located in r_t4_itcpip.h as are the prototype declarations of API functions.

2.10 Return Values

Please refer to the T4 user's manual (r20uw0031ejxxxx_t4tiny.pdf) and Ethernet Driver Interface Specification (r20uw0032ejxxxx_t4tiny.pdf). These documents are included to T4 FIT Module (R20AN0051).

2.11 Callback Function

Please refer to the T4 user's manual (r20uw0031ejxxxx_t4tiny.pdf) and Ethernet Driver Interface Specification (r20uw0032ejxxxx_t4tiny.pdf). These documents are included to T4 FIT Module (R20AN0051).

2.12 Adding the FIT Module to Your Project

This module must be added to each project in which it is used. Renesas recommends the method using the Smart Configurator described in (1) or (3) or (5) below. However, the Smart Configurator only supports some RX devices. Please use the methods of (2) or (4) for RX devices that are not supported by the Smart Configurator.

(1) Adding the FIT module to your project using the Smart Configurator in e^2 studio
By using the Smart Configurator in e^2 studio, the FIT module is automatically added to your project. Refer to "Renesas e^2 studio Smart Configurator User Guide (R20AN0451)" for details.

(2) Adding the FIT module to your project using the FIT Configurator in e^2 studio
By using the FIT Configurator in e^2 studio, the FIT module is automatically added to your project. Refer to "Adding Firmware Integration Technology Modules to Projects (R01AN1723)" for details.

(3) Adding the FIT module to your project using the Smart Configurator in CS+
By using the Smart Configurator Standalone version in CS+, the FIT module is automatically added to your project. Refer to "Renesas e^2 studio Smart Configurator User Guide (R20AN0451)" for details.

(4) Adding the FIT module to your project in CS+
In CS+, please manually add the FIT module to your project. Refer to "Adding Firmware Integration Technology Modules to CS+ Projects (R01AN1826)" for details.

(5) Adding the FIT module to your project using the Smart Configurator in IAR Embedded Workbench
By using the Smart Configurator Standalone version in IAR Embedded Workbench, the FIT module is automatically added to your project. Refer to "Renesas e^2 studio Smart Configurator User Guide (R20AN0535)" for details.

2.13 "for", "while" and "do while" statements

In this module, "for", "while" and "do while" statements (loop processing) are used in processing to wait for register to be reflected and so on. For these loop processing, comments with "WAIT_LOOP" as a keyword are described. Therefore, if user incorporates fail-safe processing into loop processing, user can search the corresponding processing with "WAIT_LOOP".

The following shows example of description.

```
while statement example :
/* WAIT_LOOP */
while(0 == SYSTEM.OSCOVFSR.BIT.PLOVF)
{
    /* The delay period needed is to make sure that the PLL has stabilized. */
}

for statement example :
/* Initialize reference counters to 0. */
/* WAIT_LOOP */
for (i = 0; i < BSP_REG_PROTECT_TOTAL_ITEMS; i++)
{
    g_protect_counters[i] = 0;
}

do while statement example :
/* Reset completion waiting */
do
{
    reg = phy_read(ether_channel, PHY_REG_CONTROL);
    count++;
} while ((reg & PHY_CONTROL_RESET) && (count < ETHER_CFG_PHY_DELAY_RESET)); /* WAIT_LOOP */
```

3. Specification about module

Please refer to the T4 user's manual (r20uw0031ejxxxx_t4tiny.pdf) and Ethernet Driver Interface Specification (r20uw0032ejxxxx_t4tiny.pdf). These documents are included to T4 FIT Module (R20AN0051).

In case of using the TSIP feature, call the R_TSIP_OPEN() function before calling the lan_open() function.

4. Appendices

4.1 Confirmed Operation Environment

This section describes confirmed operation environment for the T4 Driver FIT module.

Table 4.1 Confirmed Operation Environment (Rev. 1.08)

Item	Contents
Integrated development environment	Renesas Electronics e^2 studio Version 7.3.0 IAR Embedded Workbench for Renesas RX 4.11.1
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V3.01.00 Compiler option: The following option is added to the default settings of the integrated development environment. -lang = c99
	GCC for Renesas RX 4.8.4.201801 Compiler option: The following option is added to the default settings of the integrated development environment. -std=gnu99
	IAR C/C++ Compiler for Renesas RX version 4.11.1 Compiler option: The default settings of the integrated development environment.
Endian	Big endian/little endian
Revision of the module	Rev.1.08
Board used	Renesas Starter Kit+ for RX65N-2MB (RTK50565Nxxxxxx) Renesas Starter Kit+ for RX64M (R0K50564Mxxxxxx)

Table 4.2 Confirmed Operation Environment (Rev. 1.09)

Item	Contents
Integrated development environment	Renesas Electronics e^2 studio V20.10.0 IAR Embedded Workbench for Renesas RX 4.14.1
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V3.02.00 Compiler option: The following option is added to the default settings of the integrated development environment. -lang = c99
	GCC for Renesas RX 4.8.4.201803 Compiler option: The following option is added to the default settings of the integrated development environment. -std=gnu99
	IAR C/C++ Compiler for Renesas RX version 4.14.1 Compiler option: The default settings of the integrated development environment.
Endian	Big endian/little endian
Revision of the module	Rev.1.09
Board used	Renesas Starter Kit+ for RX65N-2MB (RTK50565Nxxxxxx) Renesas Starter Kit+ for RX64M (R0K50564Mxxxxxx)

4.2 Troubleshooting

(1) Q: I have added the FIT module to the project and built it. Then I got the error: Could not open source file “platform.h”.

A: The FIT module may not be added to the project properly. Check if the method for adding FIT modules is correct with the following documents:

- Using CS+:

  Application note “Adding Firmware Integration Technology Modules to CS+ Projects (R01AN1826)”

- Using e^2 studio:

  Application note “Adding Firmware Integration Technology Modules to Projects (R01AN1723)”

When using this FIT module, the board support package FIT module (BSP module) must also be added to the project. Refer to the application note “Board Support Package Module Using Firmware Integration Technology (R01AN1685)”.

Revision History

Rev.	Date	Description	
		Page	Summary
1.09	Apr 01, 2021	—	Countermeasures for more security: - Changed get_random_number() and added get_hash_value() for countermeasures for more security. Changed specification: - Added the TSIP FIT module in software stack because of countermeasures for more security.
1.08	Jun 28, 2019	—	Added Target Compiler (GCC for Renesas RX, IAR C/C++ Compiler for Renesas RX)
1.07	Dec 10, 2018	—	Fixed a problem that IP can not be acquired when DHCP is enabled and multiple boards are restarted at the same time. Fixed an issue where IP addresses conflict when connecting a repeater hub.
1.06	Nov 30, 2016	—	Added lan_check_link(). Added register_callback_linklayer(). Dependence upon r_sys_time_rx module. Fixed Figure1.
1.05	Oct 01, 2016	—	Added Support MCUs(RX63N,RX65N).. Added the LinkProcess function execution (R_ETHER_LinkProcess()) during processing of the timer interrupt.
1.04	Apr 15, 2016	—	Fixed: count timing for 10ms tick software counter variable "tcpudp_time_cnt".
1.03	Dec 01, 2015	—	Added: get_random_number() Changed: - When transmit data size is lesser than 60 byte, lack size amount will be padded using zero-padding in lan_write().
1.02	Jan 05, 2015	—	Changed document Title. Added Support MCUs. Changed FITModule name.
1.01	Jul 01, 2014	—	Enabled multiplex interrupt in the following function, timer_interrupt() and lan_inthdr().
		1	Fixed Introduction.
		3	Fixed Section 2.2 , 2.3
		5	Fixed Figure 1 , and section 4
1.00	May 01, 2014	—	First edition issued

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

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

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

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/.