RA Family

External Flash Library

Introduction

This application note describes a library for controlling external flash memory connected to an RA microcontroller (hereafter referred to as “MCU”). By using the library introduced in this document, you can perform write and debug operations on the external flash memory through an on-chip debugging emulator.

Please note that the sample project described in this application note is provided for reference purposes only, and its operation is not guaranteed by our company. If you choose to use the sample project, ensure that you implement the appropriate control processes for the target MCU and external flash memory, and perform sufficient evaluation in a suitable environment before deployment.

Target Device

RA8 Series

Target Software

The sample project described in this application note is listed below:

(1) r20an0822ref0100-ra-refl.zip

Contents

1. Overview

This application note provides an overview of the “External Flash Library,” which controls external flash memory connected to RA family microcontrollers (hereafter referred to as “MCU”). By using the External Flash Library, it is possible to perform write and debug operations on external flash memory through an on-chip debugging emulator or a flash memory programmer.

In addition to this document, please make sure to review the user manuals for each emulator unit and Renesas tools listed in **Table 1.1**, as well as the hardware section of the user manual for the target MCU.

Table 1.1. Related Documents

Document Name	Document Number	
	Japanese	English
E2 Emulator RTE0T00020KCE00000R User's Manual	R20UT3538JJ	R20UT3538EJ
E2 Emulator Lite RTE0T0002LKCE00000R User's Manual	R20UT3240JJ	R20UT3240EJ
Renesas Flash Programmer Vx.xx Flash memory programming software User's Manual	*	*
E2 Emulator, E2 Emulator Lite Additional Document for User’s Manual (Notes on Connection of RA Devices)	R20UT4686JJ	R20UT4686EJ
Renesas Boot Firmware for RA8M1 MCU Group	-	R01AN7140EU
Renesas Boot Firmware for RA8D1 MCU Group	-	R01AN7290EU
Renesas Boot Firmware for RA8T1 MCU Group	-	R01AN7291EU
Renesas Boot Firmware for RA8E1 MCU Group	-	R01AN7535EU
Renesas Boot Firmware for RA8E2 MCU Group	-	R01AN7547EU
Renesas Boot Firmware for RA8P1 MCU Group	-	R01AN7823EU

* Note: Varies by software version.
For documentation on Renesas Flash Programmer, please refer to the following webpage:
https://www.renesas.com/software-tool/renesas-flash-programmer-programming-gui#document

2. Glossary

In this document, the main terms used, and their explanations are shown below.

Table 2.1. Glossary

Term	Description
RA Family	A collective term for Renesas Advanced (RA) 32-bit microcontroller (MCU) products.
User System	The customer's application system that uses the target MCU.
Loader file	A file generated by building the External Flash Library and converting it into a format that can be loaded into the tool.
E2	E2 emulator (RTE0T00020KCE00000R)
E2 Lite	E2 emulator Lite (RTE0T0002LKCE00000R)
Emulator	In this document, refers to E2 and E2 Lite.
Debugger	In this document, software used for on-chip debugging of MCUs with e² studio and E2/E2 Lite.
Programming Software	In this document, refers to Renesas Flash Programmer.
Tool	In this document, refers to the debugger and programming software.
Renesas Boot Firmware	A program included in Renesas MCUs for erasing, writing, and verifying flash memory and MRAM.

3. Hardware Configuration

The hardware configuration for using the External Flash Library is shown in Figure 3-1.

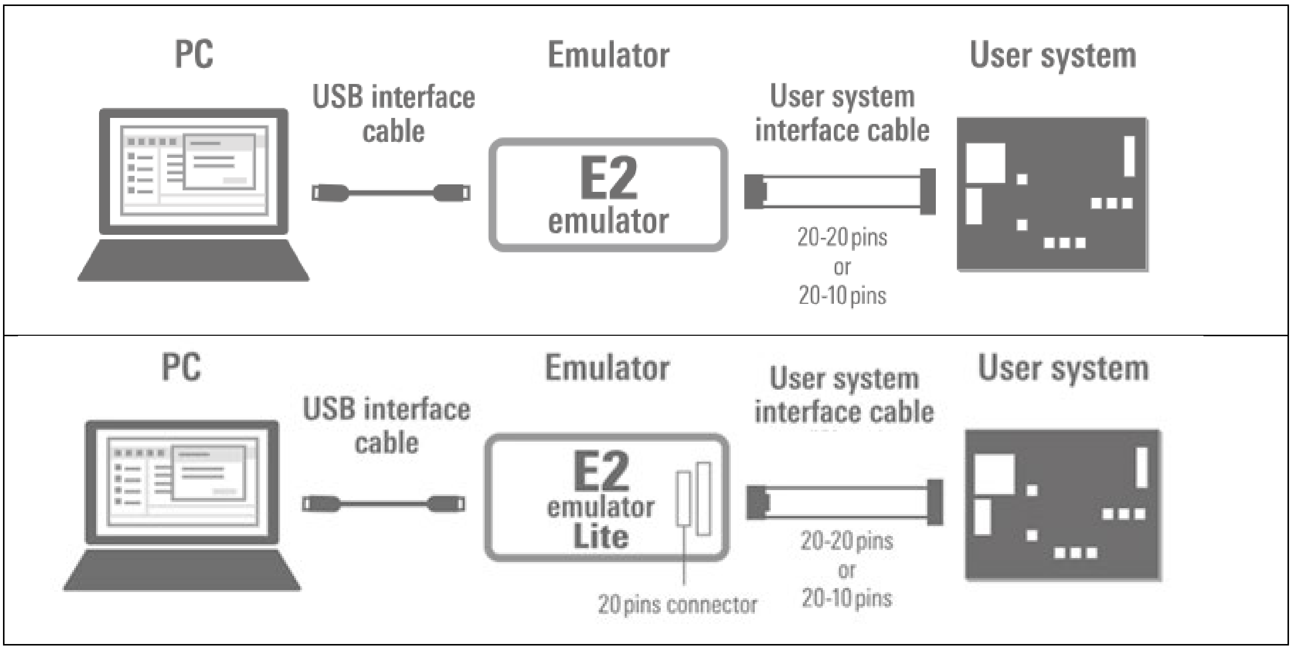


Figure 3-1 Hardware Configuration Using External Flash Library

4. Library Specifications

This chapter describes the specifications of the External Flash Library (hereafter referred to as “the library”). The library defines the interface and control processes that provide basic functions—such as erase, write, verify, and read—for external flash memory through the tool.

4.1 Scope of Application

The library is valid for the system configurations of the tools shown below. By specifying the loader file of the library for each tool, you can use the functions for erasing, writing, verifying, and reading external flash memory.

Table 4.1. System Configuration List (Debugger)

Name	Product Name
Hardware tools	E2 Emulator
	E2 Emulator Lite
Software tools	e^2 studio 2026-04 and later
	RA Flexible Software Package (FSP) v6.3.0 and later
User Interface Cable	User Interface Cable for Emulator
User System	Equipped with RA8 Series MCU and External Flash Memory (connected to the MCU via xSPI)

Table 4.2. System Configuration List (Flash memory programming software)

Name	Product Name
Hardware tool	Same as in **Table 4.1** (Debugger)
Software tool	Renesas Flash Programmer V3.22 and later
User Interface Cable	Same as in **Table 4.1** (Debugger)
User System	Same as in **Table 4.1** (Debugger)

4.2 Supported Formats

The loader file of the library supports the following standard formats as input file formats when loading into the tool:

✓ Intel HEX Format

4.3 Library Configuration

The loader file of this library is organized into the following two sections according to their functional roles:

(1) Information Definition Section

Defines specific information related to the target external flash memory, as well as configuration data required by the tool to control the external flash memory.

(2) Driver Function Data Section

Contains data for a set of driver functions that provide basic operations such as initialization, erase, and write for the target external flash memory.
The tool uses these driver functions to control the external flash memory.

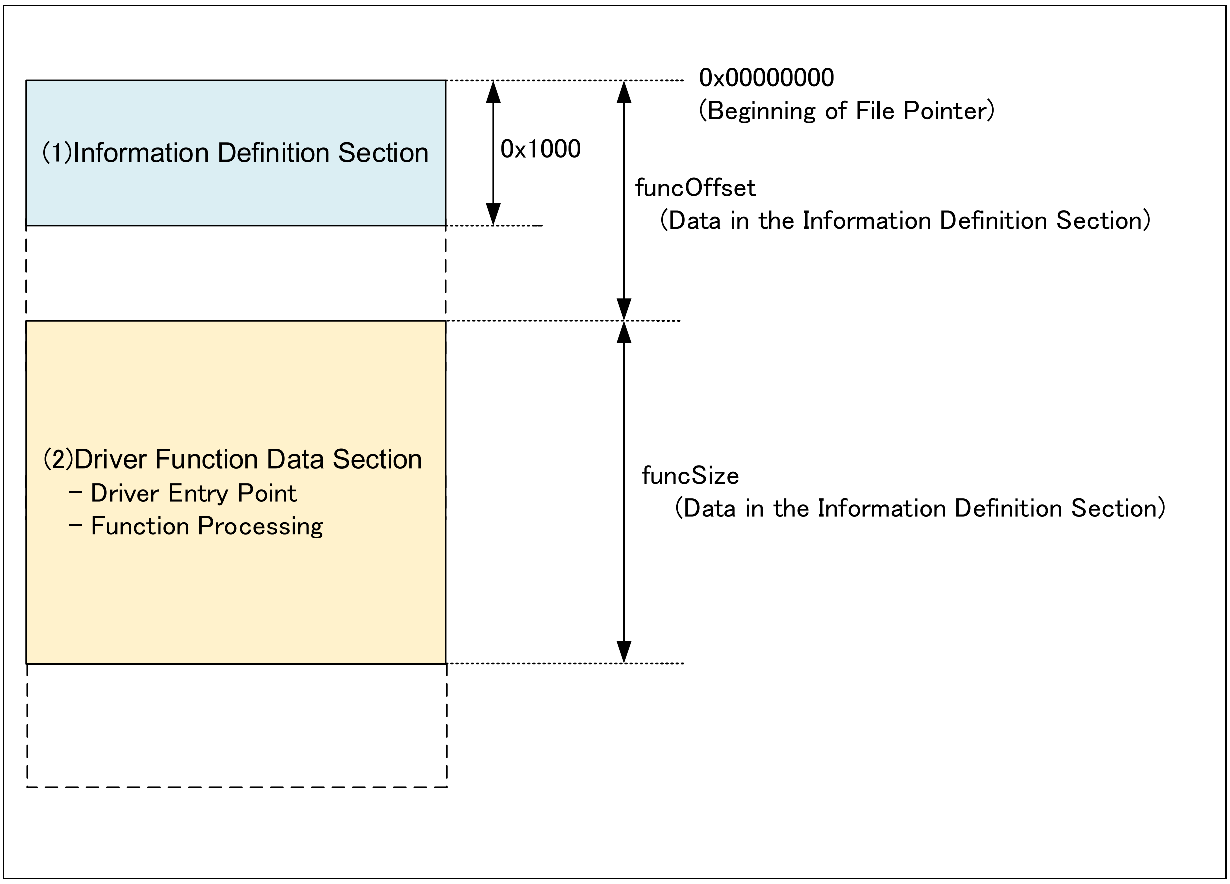


Figure 4-1 Data Structure of the Loader File

4.4 Information Definition Section

This section explains the Information Definition Section of the library. Starting from the beginning of the loader file of the library, define the items in the order shown below.

All data sequences, except for strings (char type), must be in little-endian format.

For string (char type) definitions, use only half-width alphanumeric characters (A–Z, a–z, 0–9) and the symbols “-” and “_”; do not use any other characters. In addition, always append a null character (\0) at the end of the string.

Table 4.3. List of Information Definition Section Items

Definition name	Data Type	Size(Byte)	Contents
fileVersion	uint16_t	2	File version information of the loader file
formatVersion	uint16_t	2	Format version of the loader file
funcOffset	uint32_t	4	Offset address of the driver function data section within the loader file
funcSize	uint32_t	4	Size of the driver function data section within the loader file
name	char	64 (Up to)	Name of the loader file
devName	char	64 (Up to)	Target MCU part numbering name
extFlashDevName	char	64 (Up to)	Product name of the external flash memory
devAdr	uint32_t	4	Start address of the external flash memory region (based on the target MCU’s memory map)
extFlashSize	uint32_t	4	Size of the external flash memory region (in bytes)
pageSize	uint32_t	4	Write unit size of the external flash memory (in bytes)
paramType *1	uint32_t	4	Parameter type for the tool
parameter *1	uint32_t	4 * 8 (fixed)	Parameters corresponding to paramType
timeoutProg	uint32_t	4	Fixed to 8 entries; unused entries must be set to 0xFFFFFFFF.
timeoutErase	uint32_t	4	Timeout period (in ms) for the memory write function in the driver function data section
reserved	uint32_t	4 * 4 (fixed)	Timeout period (in ms) for the sector erase function in the driver function data section
sectors(Size)	uint32_t	(4 , 4) * n n= up to 32	Sector configuration information of the external flash memory. For sectors(Size), specify the size of one sector (in bytes); for sectors(Num), specify the number of sectors. Specify up to 32 sets of this configuration starting from the external flash memory region’s start address (devAdr). The termination entry must be all 0xFFFFFFFF. Example: For S28HL512T (mounted on EK-RA8M1) {{ 0x1000, 32 }, { 0x20000, 1 }, { 0x40000, 255 }, { 0xFFFFFFFF, 0xFFFFFFFF } }
sectors(Num)	uint32_t		

*1 Note:

For RA8 Series Microcontrollers, specify as follows:

- ✓ paramType = 0x00000001
- ✓ ‘parameter’ values correspond to the settings for OCTACLK frequency (OCK) and VCC2 voltage (VCC) in the *External Flash Memory Setting Command* of Renesas Boot Firmware.
    - ➢ parameter[0] = 0x00000001 // For RA8P1, OCTACLK = 100 MHz
    - ➢ parameter[1] = 0x00000000 // For RA8P1, VCC2 voltage ≥ 2.7 V
    - ➢ parameter[2] = 0xFFFFFFFF // Unused

      :
    - ➢ parameter[7] = 0xFFFFFFFF // Unused

For details, refer to the application note “Renesas Boot Firmware” for the target MCU.

4.5 Driver Function Data Section

This section explains the driver function data area of the library. The driver function data section is placed in the internal SRAM region of the target MCU and executed by the tool you are using.

For RA8 Series MCU, the placement address is fixed according to the specifications of Renesas Boot Firmware. For details, please refer to the application note “Renesas Boot Firmware” for the target microcontroller.

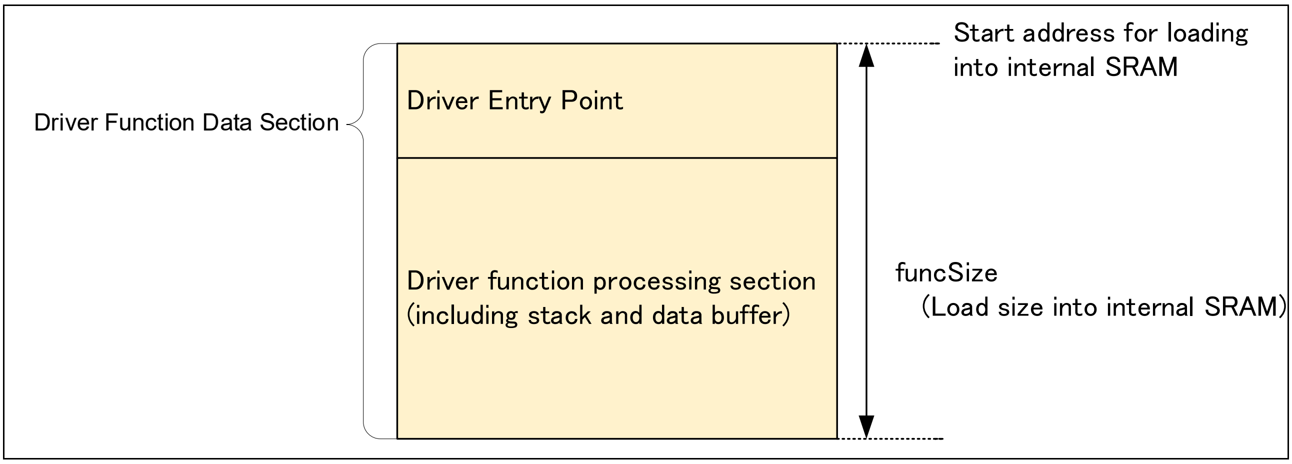


Figure 4-2 Structure of the driver function data section

4.5.1 Driver Entry Point

The driver entry point is placed at the beginning of the driver function data section and is loaded into the internal SRAM of the target microcontroller together with the driver function processing section.

This area contains entry point functions for controlling external flash memory, which are used by the tool to perform basic operations such as initialization, erase, and write.

The entry point functions and their offset addresses are as follows:

Table 4.4. List of Entry Point Functions

Function name	Offset Address	Contents
R_Flash_Initialize	0x0000	Initialization
R_Flash_EraseSector	0x0040	Sector erase
R_Flash_EraseChip	0x0080	Chip erase
R_Flash_ProgramData	0x00C0	Memory write
R_Flash_Uninitialize	0x0100	(Reserved) Un-Initialization
R_Flash_ReadData	0x0140	(Reserved) Memory read
R_Flash_CalcCRC	0x0180	(Reserved) CRC calculation

Note: (Reserved) is not used for RA8 Series MCU.

When the tool calls an entry point function, it uses the common interface shown below.

Table 4.5. Entry Point Function Arguments / Return value

Items	General Purpose Register name
Argument #1	R0
Argument #2	R1
Argument #3	R2
Return	R3

4.5.2 Entry Point Function Specifications

This section describes the function interface specifications between the tool and the entry point functions.

(1) Initialization

int32_t R_Flash_Initialize (uint32_t rfu)	
Argument #1 rfu [in]	(Reserved)
Return	Success: 00000000h Failure: FFFFFFFFh
Description	Initializes the target MCU and external flash memory settings necessary for external flash memory operations. (For RA8 Series MCUs) It is used as an External Flash Memory Access Driver by Renesas Boot Firmware. Please refer to the application note “Renesas Boot Firmware” for the target MCU and take care to avoid conflicts or duplication in the initialization process.

(2) Sector erase

int32_t R_Flash_EraseSector (uint32_t addr)	
Argument #1 addr [in]	Start address of the erase sector. The specified address must be within the region assigned to the external flash memory in the address map of the target MCU.
Return	Success: Erased data size(in byte) Failure: FFFFFFFFh
Description	Erases one sector of the external flash memory. (For RA8 Series MCUs) It is used as an External Flash Memory Access Driver by Renesas Boot Firmware.

(3) Chip erase

int32_t R_Flash_EraseChip (void)	
Argument #1	None
Return	Success: 00000000h Failure: FFFFFFFFh
Description	Erases the entire external flash memory region. (For RA8 Series MCUs) It is used as an External Flash Memory Access Driver by Renesas Boot Firmware.

(4) Memory write

int32_t R_Flash_ProgramData(uint32_t addr, const void *data, uint32_t cnt)	
Argument #1 addr [in]	Start address for writing
Argument #2 data [in]	Pointer to the write data buffer
Argument #3 cnt [in]	Size of the write data (in bytes)
Return	Success: Written data size(in bytes) Failure: FFFFFFFFh
Description	Writes data to the external flash memory starting from the specified start address for the specified size (cnt). (For RA8 Series MCUs) It is used as an External Flash Memory Access Driver by Renesas Boot Firmware.

(5) Un-Initialization

int32_t R_Flash_Uninitialize(uint32_t rfu)	
Argument #1 rfu [in]	(Reserved)
Return	Success: Written data size(in bytes) Failure: FFFFFFFFh
Description	Releases the initial settings to complete the external flash memory operation. (For RA8 series MCUs) Not used.

(6) Memory read

int32_t R_Flash_ReadData(uint32_t addr, void *data, uint32_t cnt)	
Argument #1 addr [in]	Start address for reading
Argument #2 data [out]	Pointer to the read data buffer
Argument #3 cnt [in]	Size of the read data (in bytes)
Return	Success: Read data size(in bytes) Failure: FFFFFFFFh
Description	Reads data to the external flash memory starting from the specified start address for the specified size (cnt). (For RA8 series MCUs) Not used.

(7) CRC calculation

uint32_t R_Flash_CalcCRC(uint32_t crc, uint32_t addr, uint32_t cnt, uint32_t polynom)	
Argument #1 crc [in]	Initial CRC value
Argument #2 addr [in]	Start address of the data for CRC calculation
Argument #3 cnt [in]	Data length (in bytes)
Argument #4 polynom [in]	Polynomial defined for each CRC type
Return	CRC value
Description	Calculates the CRC value of the data in the external flash memory starting from the specified start address. (For RA8 series MCUs) Not used.

5. Sample Project

A sample project named “r20an0822ref0100-ra-refl.zip” (hereinafter referred to as “the Sample Project”) is available to facilitate the use of External Flash Library.

The following chapter describes its overview and usage.

5.1 Development Environment

The Sample Project has been verified to operate in the development environments shown in **Table 5.1** and **Table 5.2** below.

Table 5.1. Hardware Development Environment

Microcontroller / CPU Board Model	External Flash Memory Product
RA8P1 (R7KA8P1KFLCAC) / RTK7EKA8P1S01001BE	MX25LW51245GXDI00

Table 5.2. Software Development Environment

IDE Version	Toolchain Version
e² studio 2025-07	Arm GNU Toolchain 13.2.rel1

5.2 File and Folder Structure

This section describes the file and folder structure of the Sample Project.

```
sample_project
 ├──── .cproject                          # e² studio C/C++ project configuration file
 ├──── .project                           # e² studio basic project information file
 ├──── .settings
 │     ├──── com.renesas.cdt.managedbuild.gnuarm.prefs      # e² studio setting file for Managed Build
 │     └──── language.settings.xml        # e² studio setting file for language, include paths, and macros
 ├──── script
 │     └──── target.ld                    # Linker script (memory layout definition)
 └──── src
       ├──── LoaderInfo.h                 # Header file for information definitions
       ├──── LoaderInfo.c                 # Source file for information definitions
       ├──── LoaderFunc.h                 # Header file for information definitions
       ├──── LoaderFunc.c                 # Source file for driver function data
       ├──── MX25LW51245G.h               # Header file for external flash (MX25LW51245G)
       └──── RA8P1.h                      # Header file for RA8P1
```

Figure 5-1 File/Folder Structure Diagram of the Sample Project

5.3 Build Procedure

This section describes the steps to import the Sample Project into the e² studio and build it.

5.3.1 Importing the Project

(1) Extract the zip file of the Sample Project and place it in any desired folder.

(2) Launch e² studio and select [Import] from the [File] menu on the menu bar.

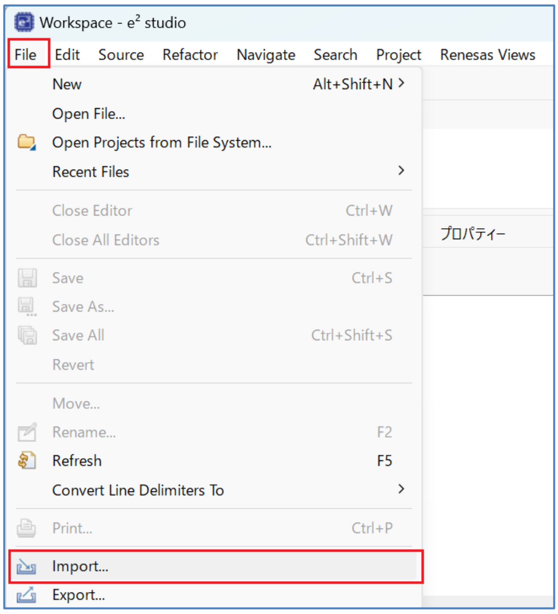
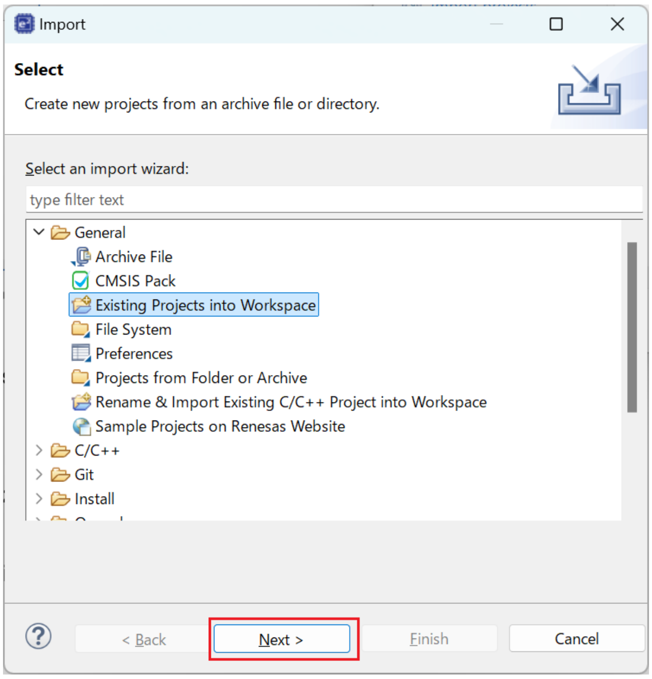


Figure 5-2 Selecting Project Import

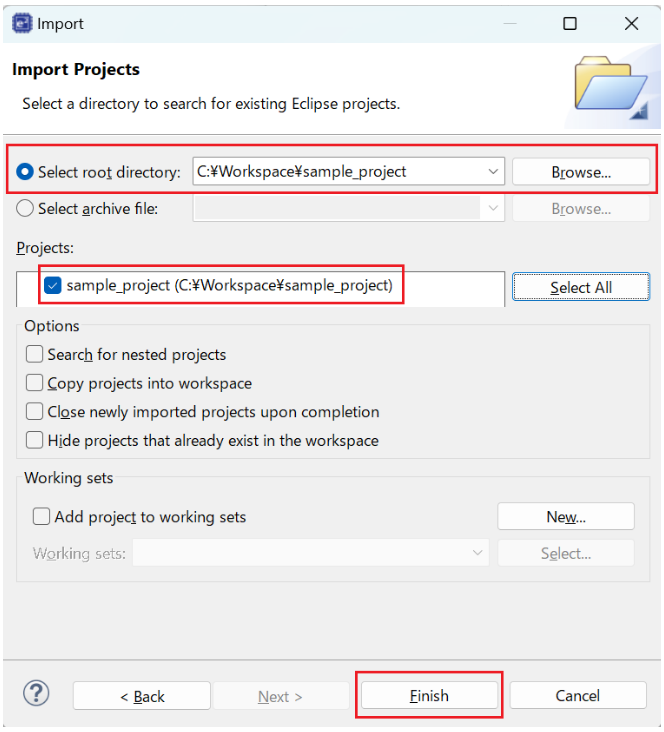
(3) In [General], select [Existing Projects into Workspace], and then click [Next].



Create new projects from an archive file or directory.

Figure 5-3 Selection in the Import Wizard

(4) In [Select root directory], enter the directory path of the Sample Project.
Next, in [Projects:], select the target project displayed, and then click [Finish].



Import Projects

Select a directory to search for existing Eclipse projects.

● Select root directory: C:¥Workspace¥sample_project

○ Select archive file:

Projects:

☑ sample_project (C:¥Workspace¥sample_project)

Options

☐ Search for nested projects

☐ Copy projects into workspace

☐ Close newly imported projects upon completion

☐ Hide projects that already exist in the workspace

Working sets

☐ Add project to working sets

Working sets:

Figure 5-4 Selection of Project to Import

(5) In [Project Explorer], verify that the project has been successfully imported.

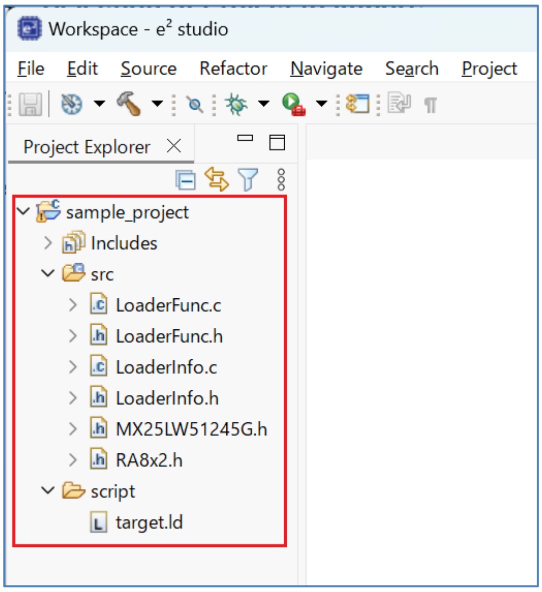


Figure 5-5 Selection of Project to Import

Note:

If [Project Explorer] is not displayed, select [Project Explorer] from [Show View] under the [Window] menu on the menu bar to display it.

5.3.2 Executing the Build

(1) From the [Project] menu on the menu bar, select [Build All].

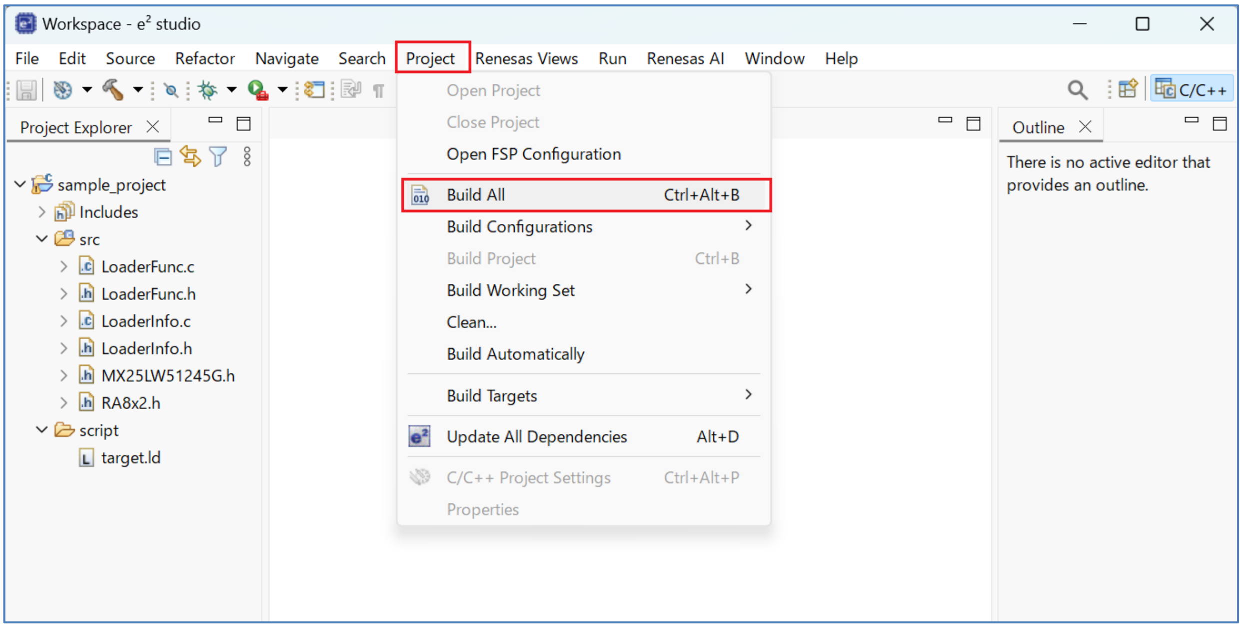
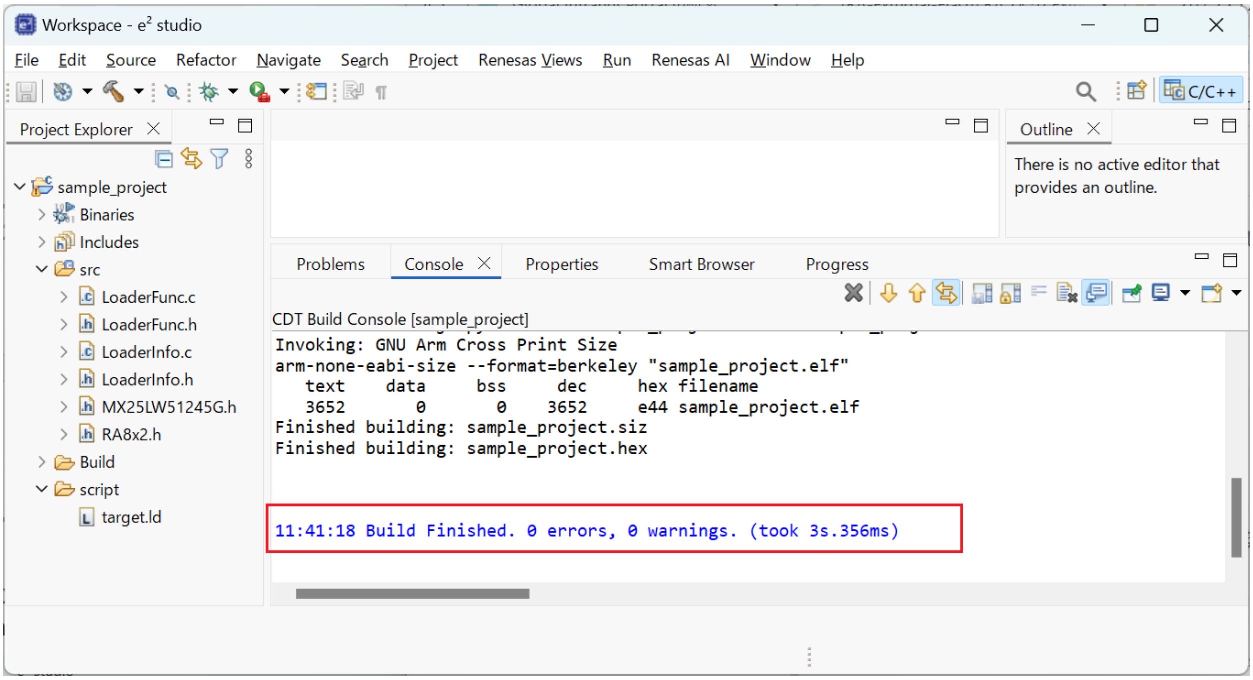


Figure 5-6 Executing the Build for the Sample Project

(2) In the [Console] view, verify that “Build Finished.” is displayed, confirming that the build has completed successfully.



CDT Build Console [sample_project]

```
Invoking: GNU Arm Cross Print Size
arm-none-eabi-size --format=berkeley "sample_project.elf"
   text    data     bss     dec     hex filename
   3652       0       0    3652     e44 sample_project.elf
Finished building: sample_project.siz
Finished building: sample_project.hex

11:41:18 Build Finished. 0 errors, 0 warnings. (took 3s.356ms)
```

Figure 5-7 Sample Project Build Result

5.4 Usage

This section explains how to use the library for each tool. The tools use only the loader file generated by building this library.

5.4.1 Usage in e² studio

This subsection describes how to use the library in e² studio.
It explains the configuration steps required to use this library in the customer's target debug project.

(1) Open [Debug Configurations] from the [Run] menu on the menu bar.

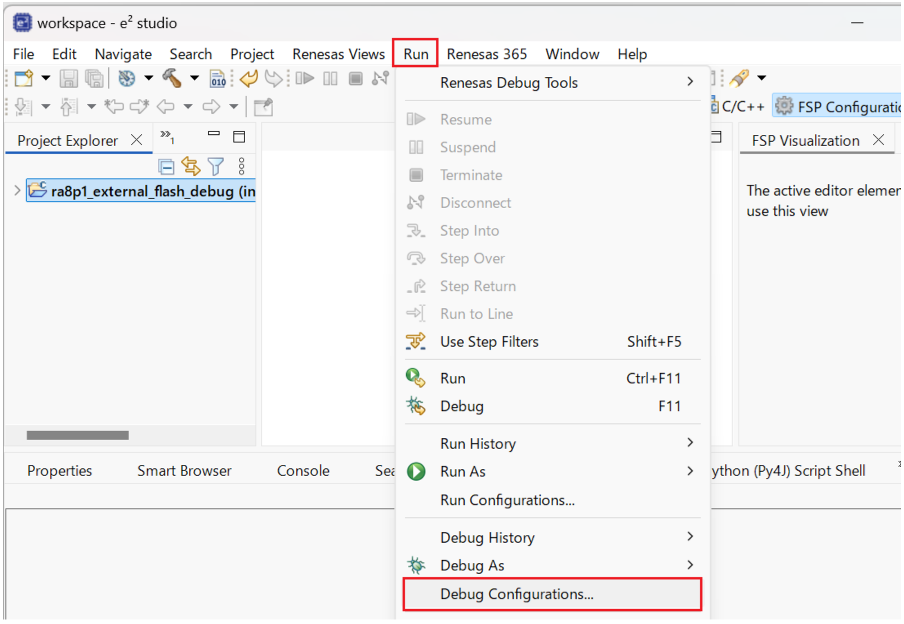
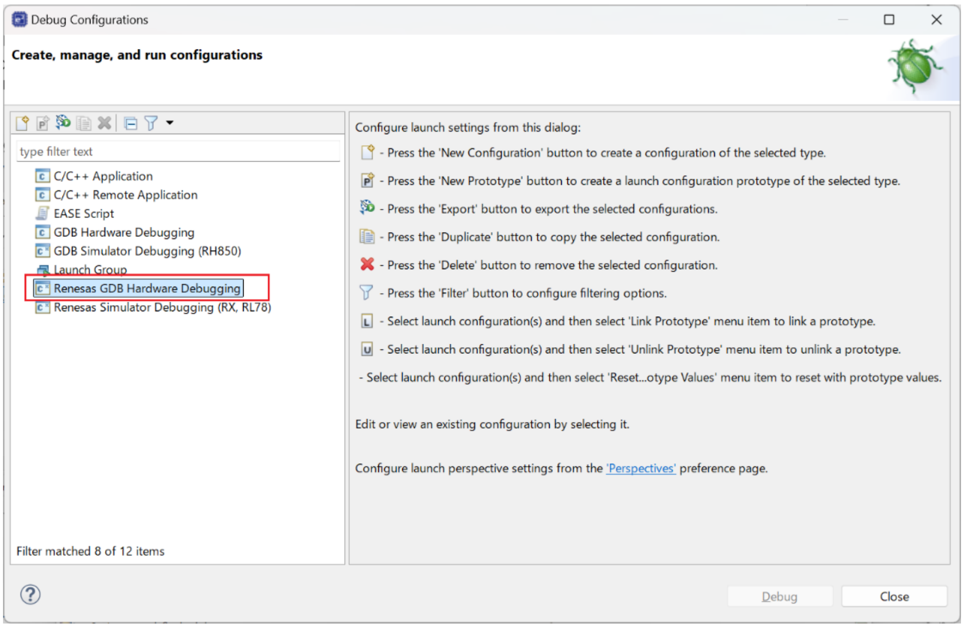


Figure 5-8 Open Debug Configurations

(2) In the Debug Configurations window, double-click [Renesas GDB Hardware Debugging].



Configure launch settings from this dialog:

- Press the 'New Configuration' button to create a configuration of the selected type.
- Press the 'New Prototype' button to create a launch configuration prototype of the selected type.
- Press the 'Export' button to export the selected configurations.
- Press the 'Duplicate' button to copy the selected configuration.
- Press the 'Delete' button to remove the selected configuration.
- Press the 'Filter' button to configure filtering options.
- Select launch configuration(s) and then select 'Link Prototype' menu item to link a prototype.
- Select launch configuration(s) and then select 'Unlink Prototype' menu item to unlink a prototype.
- Select launch configuration(s) and then select 'Reset...otype Values' menu item to reset with prototype values.

Edit or view an existing configuration by selecting it.

Configure launch perspective settings from the 'Perspectives' preference page.

Figure 5-9 Open Debug Configurations

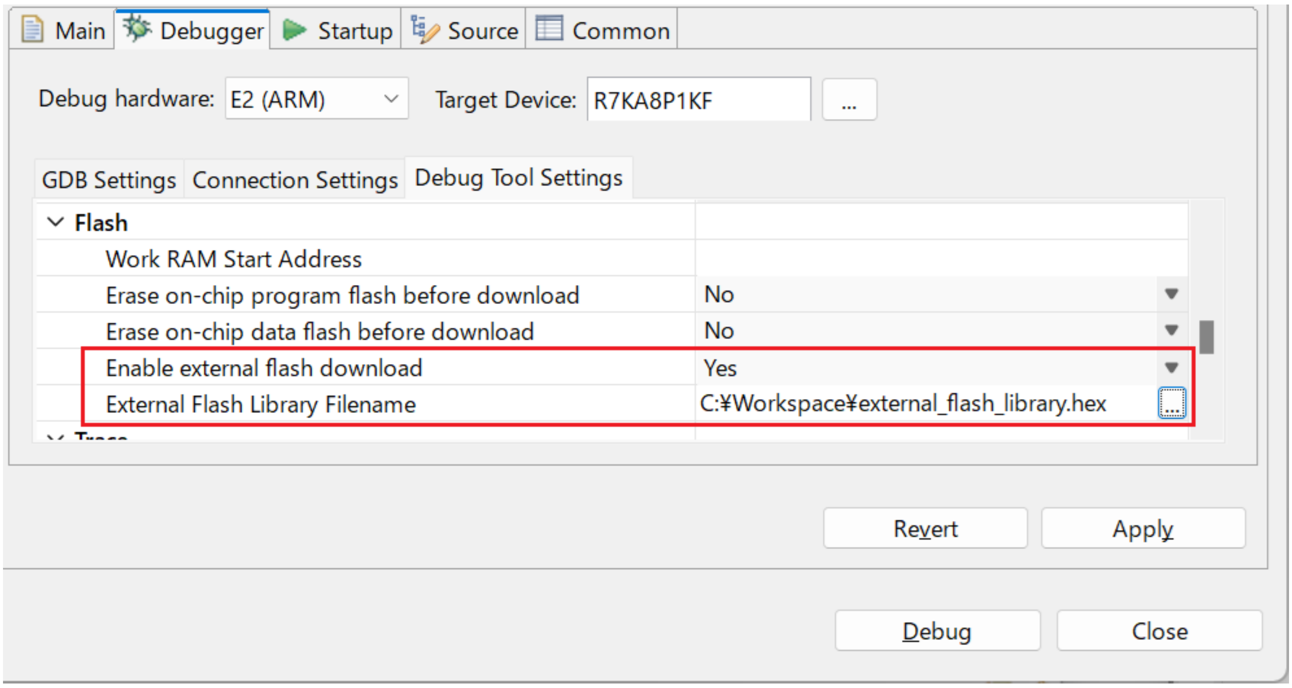
Note: If an existing debug configuration has already been created, it should be selected.

(3) In the [Debugger] tab, under [Debug Tool Settings], configure the following settings:

a) For [Enable external flash download], select [Yes].

b) For [External Flash Library Filename], specify the loader file of this library with the full path.

c) After completing steps a) and b), click [Apply].



Debug hardware: E2 (ARM)

Target Device: R7KA8P1KF

Flash	
Work RAM Start Address	
Erase on-chip program flash before download	No
Erase on-chip data flash before download	No
Enable external flash download	Yes
External Flash Library Filename	C:¥Workspace¥external_flash_library.hex

Figure 5-10 Selection of the Loader File (e² studio)

From this point onward, when debugging is started, downloading and debugging of programs and data assigned to the external flash memory will be possible.

5.4.2 Usage in Renesas Flash Programmer

This section describes how to use this library with Renesas Flash Programmer (hereafter referred to as “RFP”).

(1) Place the loader file of this library in the following folder:

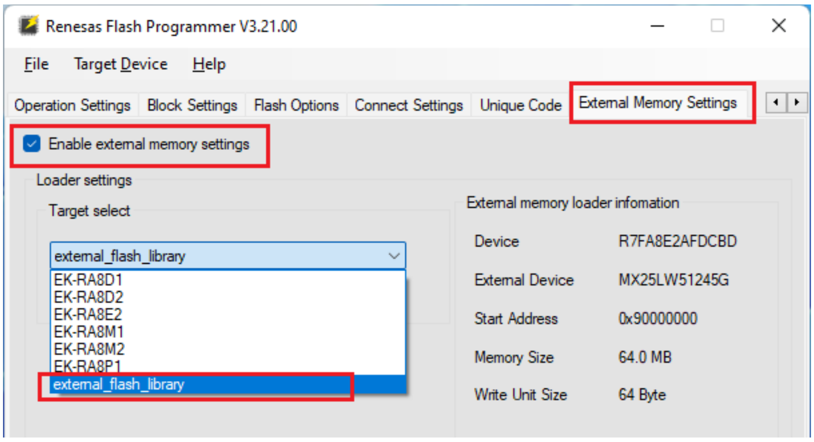
<RFP installation directory>\Resources\ExternalMemory\RA

Note: Administrator privileges may be required depending on the environment.

(2) Launch RFP and open the project to be used.

(3) In the [External Memory Settings] tab of RFP, check [Enable External Memory Settings].

(4) The loader file that was placed is to be selected from the dropdown list under [Loader Settings].



☑ Enable external memory settings

Loader settings

Target select

external_flash_library

EK-RA8D1
EK-RA8D2
EK-RA8E2
EK-RA8M1
EK-RA8M2
EK-RA8P1
external_flash_library

External memory loader infomation

Device	R7FA8E2AFDCBD
External Device	MX25LW51245G
Start Address	0x90000000
Memory Size	64.0 MB
Write Unit Size	64 Byte

Figure 5-11 Selection of the Loader File (RFP)

Note: The loader file name (without extension) is displayed in the dropdown list.

From this point onward, for instructions on how to perform write operations to the target MCU, please refer to the Renesas Flash Programmer Flash Programming Software User’s Manual.

6. Website and Support

Visit the following URLs to learn about key elements of the RA family, download components and related documentation, and get support:

RA Product Information	www.renesas.com/ra
RA Product Support Forum	https://community.renesas.com/mcu-mpu/ra/
RA Flexible Software Package	www.renesas.com/FSP
Renesas Support	www.renesas.com/support

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jan.20.26	—	Initial release

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/.