RX Family

R01AN2239EJ0145
Rev.1.45
May 15, 2026

Sample Program using using USB Peripheral Mass Storage Class Driver (PMSC) to communicate via USB with USB Host Firmware Integration Technology

Introduction

This document describes the following sample firmware: USB Peripheral Mass Storage Class Driver using Firmware Integration Technology. The sample firmware is referred to below as the PMSC.

When developing an actual software, be sure to use the “USB Basic Host and Peripheral Driver Firmware Integration Technology Application Note” (Document number: R01AN2025) together with the user’s manual for each MCU (Hardware). In addition, also refer to the " USB Peripheral Mass Storage Class Driver (PMSC) Firmware Integration Technology Application Note” (Document number.R01AN2029), if necessary. “USB Basic Host and Peripheral Driver Firmware Integration Technology Application Note” (Document number: R01AN2025) is located in the "reference_documents" folder within the package.

Target Device

RX65N/RX651 Group
RX64M Group
RX71M Group
RX66T Group
RX72T Group
RX72M Group
RX66N Group
RX72N Group
RX671 Group

The operation of this program has been confirmed using the Renesas Starter Kits (RSK).

Contents

1. **Specifications** 2
2. **Software Configuration** 4
3. **Setup** 5
4. **Sample Application** 9
5. **Class Driver Overview** 15
6. **Using RI600V4 project with CS+** 16
7. **Using the e^2 studio project with CS+** 21

1. Specifications

1.1 Functions

The PMSC performs communication with the USB host PC in conformance with the bulk-only transfer (BOT) portion of the USB mass storage class specification (MSC).

The PMSC implements the following functions:

- It is recognized as a mass storage class device when connected to the USB host PC.
- It performs file transfers to and from the USB host PC.

1.2 PMSC Configuration Elements

The PMSC comprises the following FIT modules and a sample application:

Table 1-1 PMSC Configuration Elements

FIT Module	Folder Name
RX Family Board Support Package Module Firmware Integration Technology	r_bsp
RX Family USB Basic Host and Peripheral Driver Firmware Integration Technology	r_usb_basic
RX Family USB Peripheral Mass Storage Class Driver (PMSC) Firmware Integration Technology	r_usb_pmsc
RX Family DTC Module Using Firmware Integration Technology	r_dtc_rx
RX Family DMA Controller DMACA Control Module Firmware Integration Technology	r_dmaca_rx

Refer to the related documentation for details of each FIT module. Note that the latest versions of the FIT modules used by the sample firmware are available for download from the following website:

Renesas Electronics website: http://www.renesas.com/

1.3 Note

This driver is not guaranteed to provide USB communication operation. The customer should verify operation when utilizing it in a system and confirm the ability to connect to a variety of different types of devices.

1.4 Operating Confirmation Environment

The following is the operating confirmation environment of this program.

Table 1-2 Operating Confirmation Environment

Item	Contents
C compiler	Renesas Electronics C/C++ compiler for RX Family V.3.07.00
	Compile Option：-lang = c99
Real-Time OS	FreeRTOS V.10.0.0 RI600V4
Endian	Little Endian, Big Endian
USB Driver Revision Number	Rev.1.45
Using Board	Renesas Starter Kits for RX64M Renesas Starter Kits for RX71M Renesas Starter Kits for RX65N, Renesas Starter Kits for RX65N-2MB Renesas Starter Kits for RX72T Renesas Starter Kits for RX72M Renesas Starter Kits for RX72N Renesas Starter Kits for RX671
Host Environment	The operation of this USB Driver module connected to the following OSes has been confirmed. 1. Windows® 11

1.5 Terms and Abbreviations

APL	:	Application program
BOT	:	Bulk Only Transport.
DDI	:	Device Driver Interface, or PMSDD API.
Non-OS	:	USB Driver for OS-less
PCD	:	Peripheral Control Driver for USB-BASIC-FW
PCI	:	PCD Interface
PMSCD	:	Peripheral Mass Storage Class Driver (PMSCF + PCI + DDI)
PMSCF	:	Peripheral Mass Storage Class Function
PMSDD	:	Peripheral Mass Storage Device Driver (ATAPI driver)
RSK	:	Renesas Starter Kits
RTOS	:	USB Driver for the real-time OS
USB-BASIC-FW	:	USB Basic Host and Peripheral Driver

2. Software Configuration

2.1 Module Configuration

The Peripheral Device Class Driver (PDCD) comprises a Peripheral Mass Storage Class Driver (PMSCD) and Peripheral Mass Storage Device Driver (PMSDD). The PMSCD comprises a Peripheral Mass Storage Class Function (PMSCF), which performs BOT protocol control and data transmission/reception, as well as functions (DDI) for interfacing with the PMSDD and functions (PCI) for interfacing with the PCD. The PMSCD performs BOT protocol communication with the host via the PCD. The PMSDD analyzes and processes storage commands received from the PMSCD, and accesses media data via the media driver.

Figure 2-1 shows the module configuration of the PMSC, and Table 2-1 lists the functions of the modules.

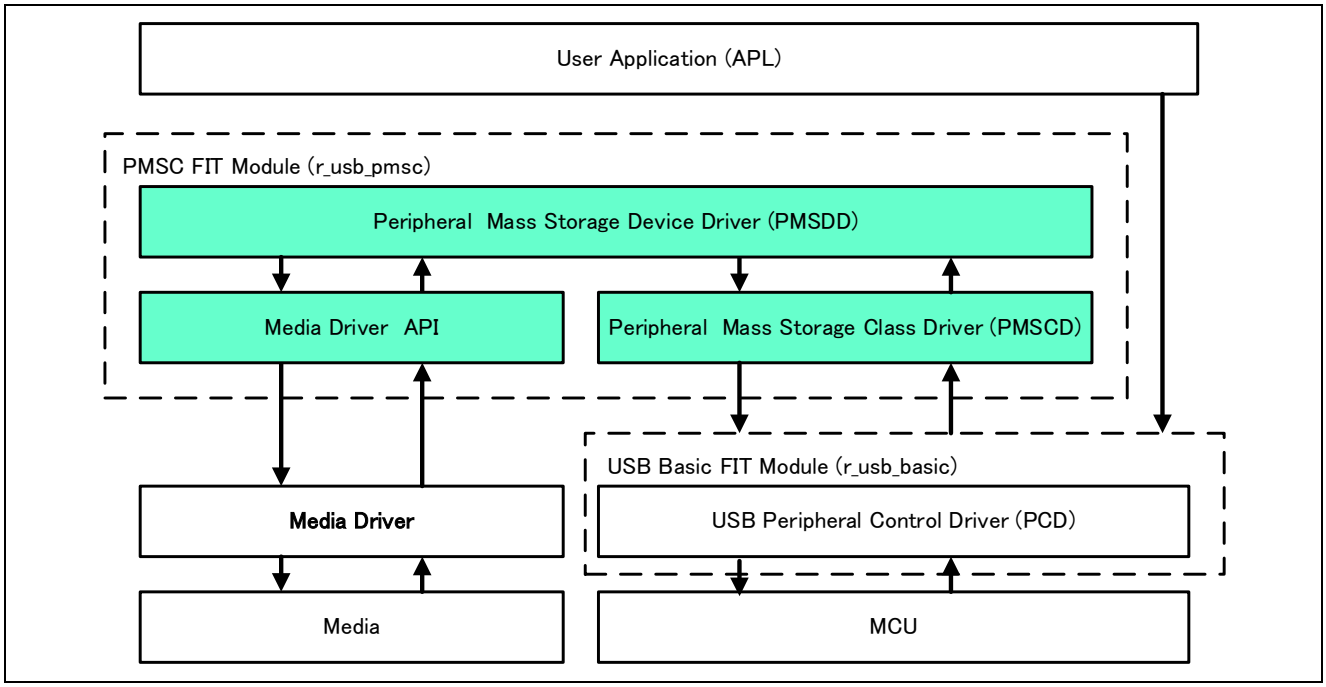


Figure 2-1 Module Configuration

Table 2-1 Functions of Modules

Module Name	Function
PMSDD	Mass storage device driver ・Processes storage commands from the PMSCD. ・Accesses media via the media driver.
PMSCD	Mass storage class driver ・Controls BOT protocol data and responds to class requests. ・Analyses CBWs and performs data transmission/reception. ・Interoperates with the PMSDD and PCD to transmit CSW.
PCD	USB Basic Driver
Media Driver	Block media storage device control driver

3. Setup

3.1 Hardware

3.1.1 Example Operating Environment

Figure 3-1 shows an example operating environment for the PMSC. Refer to the associated instruction manuals for details on setting up the evaluation board and using the emulator, etc

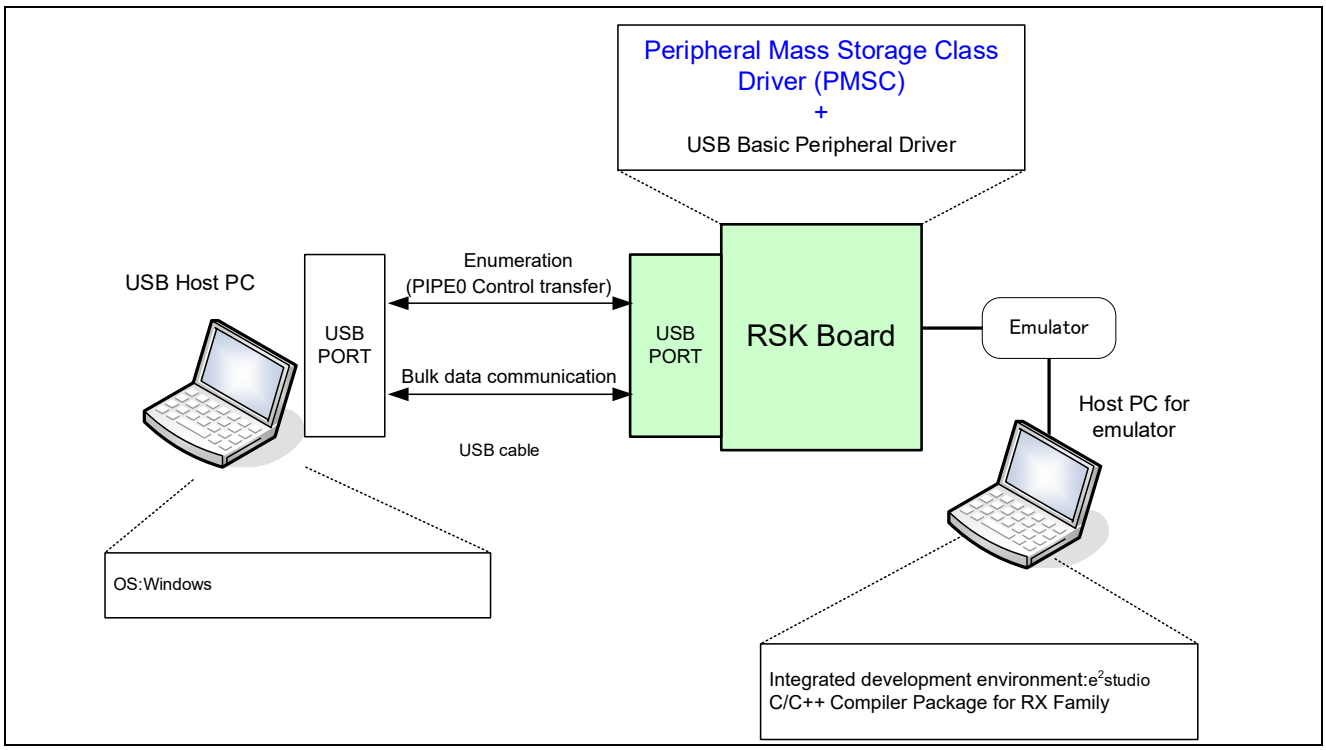


Figure 3-1 Example Operating Environment

Table 3-1 shows the evaluation board on which operation has been confirmed.

Table 3-1 Evaluation Board on Which PMSC Operation Has Been Verified

MCU	Evaluation Board
RX65N	RSK+RX65N, RSK+RX65N-2MB
RX64M	RSK+RX64M
RX71M	RSK+RX71M
RX72T	RSKRX72T
RX72M	RSK+RX72M
RX72N	RSK+RX72N
RX671	RSK+RX671

3.1.1 RSK Setting

It is necessary to set RSK to operate in the peripheral mode. Please refer to the following.

Table 3-2 **RSK Setting**

RSK	Jumper Setting
RSK+RX65N	J8: Shorted Pin2-3
RSK+RX65N_2MB	J7: Shorted Pin2-3 J16: Shorted Pin1-2
RSK+RX64M (USB0)	J2: Shorted Pin2-3 J6: Shorted Pin1-2
RSK+RX64M (USBH)	J7: Shorted Pin2-3 J9: Shorted Pin1-2
RSK+RX71M (USB0)	J1: Shorted Pin2-3 J3: Shorted Pin1-2
RSK+RX71M (USBA)	J4: Shorted Pin2-3 J7: Shorted Pin1-2
RSKRX72T	J13: Shorted Pin2-3
RSK+RX72M	J8: Shorted Pin1-2 J10: Shorted Pin1-2
RSK+RX72N	J7: Shorted Pin1-2 J8: Shorted Pin1-2
RSK+RX671	J8: Shorted Pin1-2 J13: Shorted Pin1-2

Note:

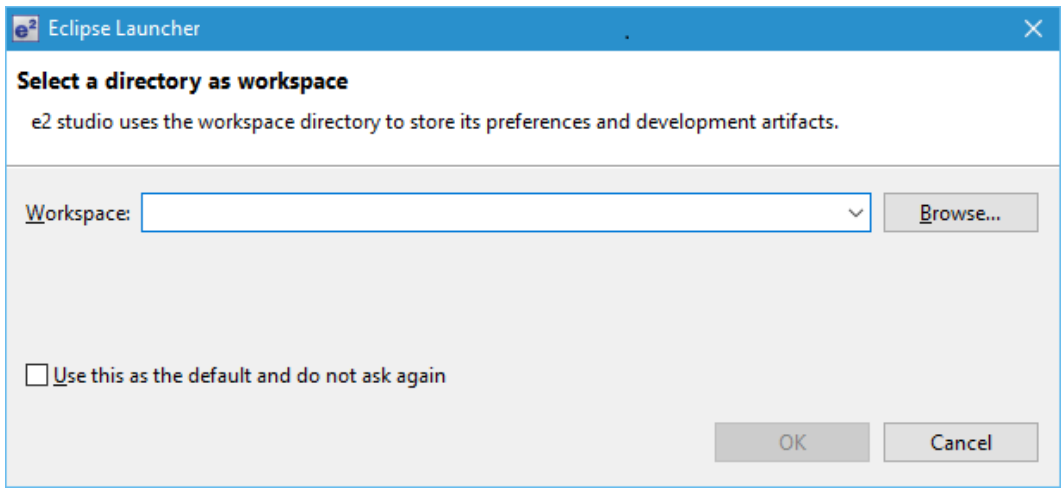
For the detail of RSK setting, refer to the user's manual of RSK.

3.2 Software

1) Setup e² studio

a) Start e² studio

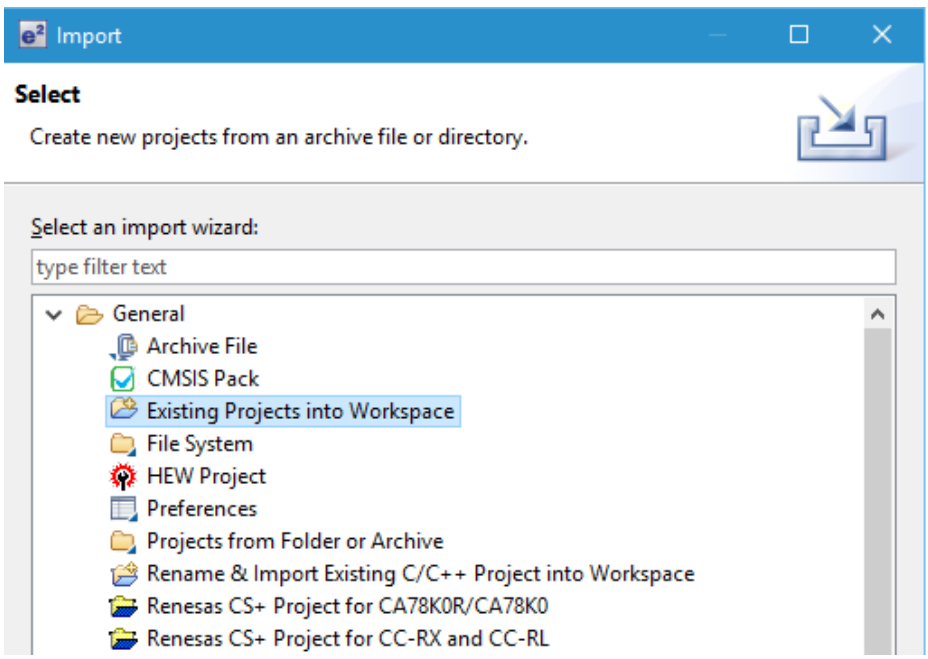
b) If you start up e² studio at first, the following dialog is displayed. Specify the folder to store the project in this dialog.



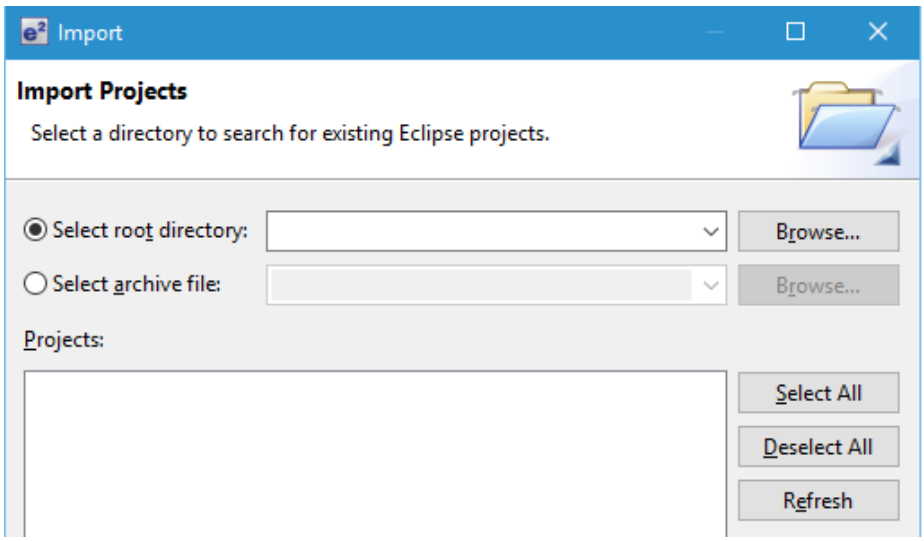
2) Import the project to the workspace

a) Select [File] > [Import]

b) Select [General] => [Existing Projects into Workspace]



c) Select the root directory of the project, that is, the folder containing the ".cproject" file.



d) Click "Finish".

You have now imported the project into the workspace. Note that you can import other projects into the same workspace.

3) Generate the binary target program by clicking the "Build" button.

4) Connect the target board to the debug tool and download the executable. The target is run by clicking the "Run" button.

4. Sample Application

4.1 Application Specifications

The PMSC sample application (APL) runs on the RSK. When the RSK is connected to the host PC it is recognized as a removable disk, and data transfers, such as reading and writing files, can be performed.

Figure 4-1 shows an example PMSC operating environment, and Figure 4-2 shows a PMSC operation example.

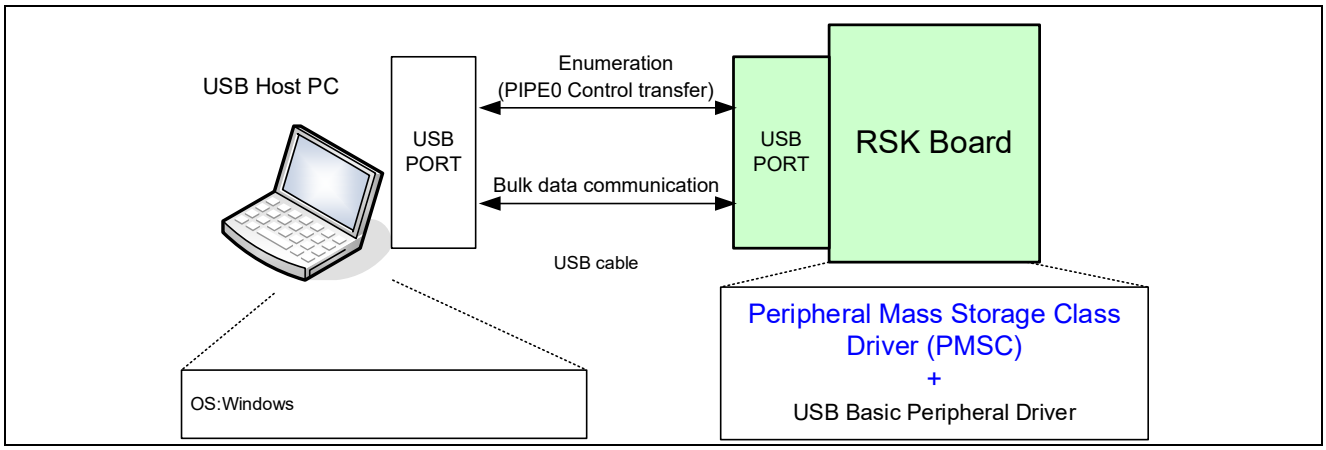


Figure 4-1 PMSC Operation Environment

4.1.1 Media Area on Removable Disk

The APL uses the SDRAM area on the RSK as the media area of a removable disk.

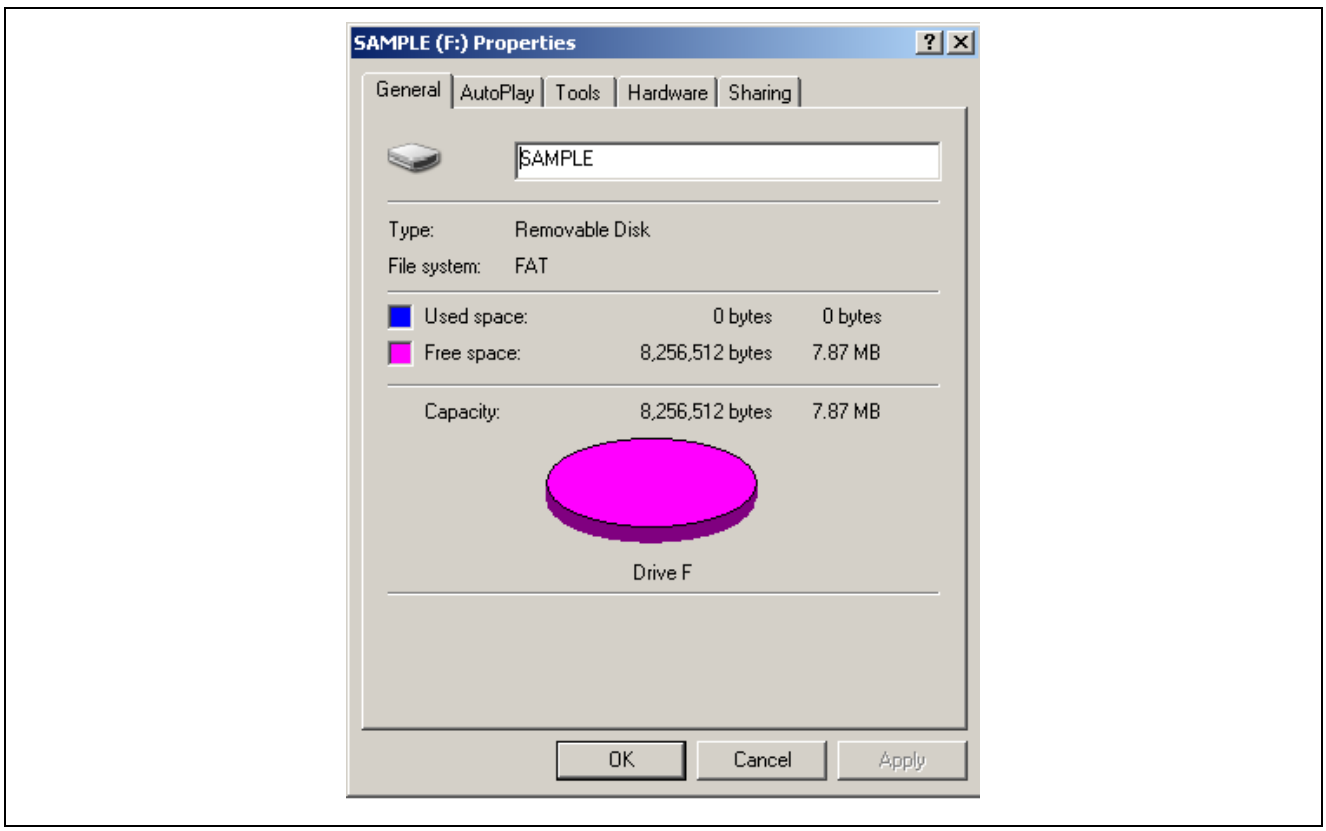


Figure 4-2 PMSC Operation Example

Note:

The internal RAM is used as the storage media when using RX72T.

4.2 Application Processing (for Non-OS)

The application comprises two parts: initial settings and main loop.

Initial setting : Makes MCU pin settings, initializes the USB controller, and initializes the USB driver.

Main loop : Calls *R_USB_GetEvent* function in the loop. If a Suspend request is received from the USB host or the USB host is detached while the loop is being processed, the APL transitions the MSC device (RSK) to low-power mode. For details of low-power mode, see 4.5, MCU Low power consumption processing

PMSC controls processing by a mass storage class driver (MSCD) and mass storage device driver (MSDD) in response to requests from the USB host (PC). Therefore, the PMSC APL does not perform any processing on data transferred from the host. Aside from initialization processing, the only thing performed within the loop is calling the *R_USB_GetEvent* function. The APL does not write files to or read files from the PMSC storage area; this processing is all performed by the PMSC USB driver.

[Note]

1. For a list of the storage commands supported by the PMSC, see 5.2,Storage Commands.
2. Make sure to call the *R_USB_GetEvent* function from within the application program loop processing.

An overview of the processing performed by the APL is shown below:

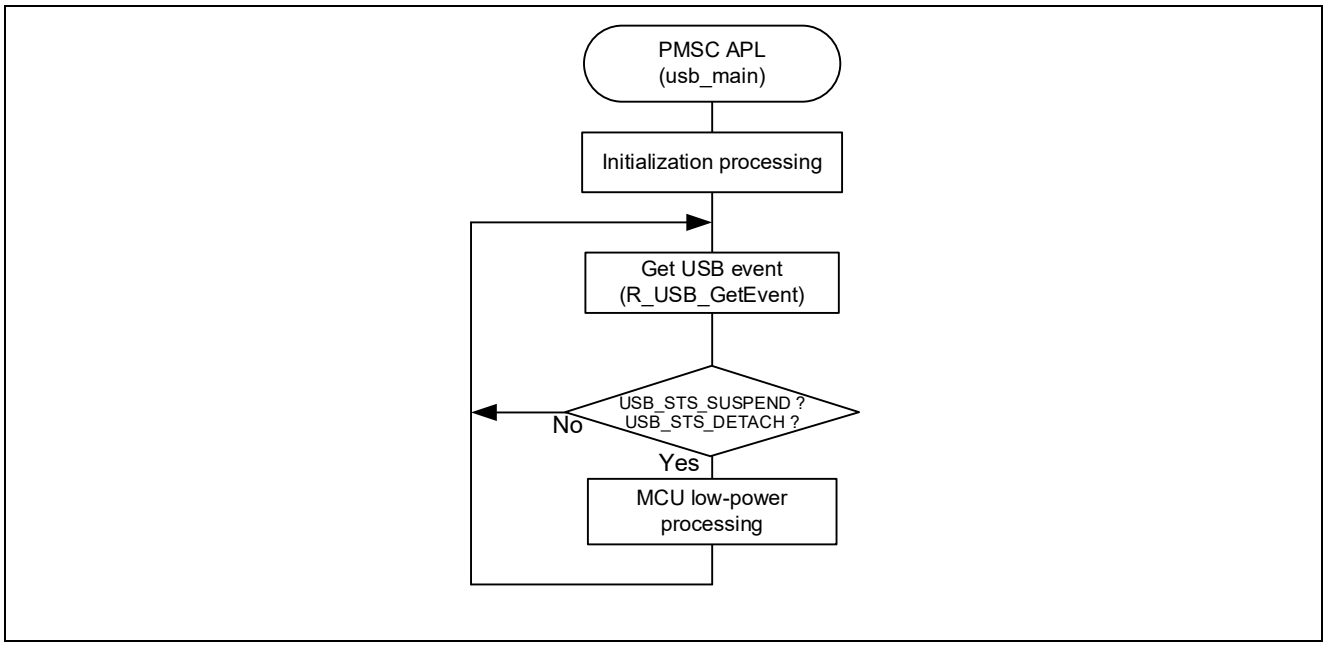


Figure 4-3 APL Processing Overview

4.3 Application Processing (RTOS)

The application comprises two parts: initial settings and main loop.

Initial setting : Makes MCU pin settings, initializes the USB controller, and initializes the USB driver.

Main loop : The reception processing of USB completion event is performed in the loop. If a Suspend request is received from the USB host or the USB host is detached while the loop is being processed, the APL transitions the MSC device (RSK) to low-power mode. For details of low-power mode, see 4.5, MCU Low power consumption processing

PMSC controls processing by a mass storage class driver (MSCD) and mass storage device driver (MSDD) in response to requests from the USB host (PC). Therefore, the PMSC APL does not perform any processing on data transferred from the host. Aside from initialization processing, only the reception processing of USB completion event is performed within the loop. The APL does not write files to or read files from the PMSC storage area; this processing is all performed by the PMSC USB driver.

[Note]

For a list of the storage commands supported by the PMSC, see 5.2,Storage Commands.

An overview of the processing performed by the APL is shown below:

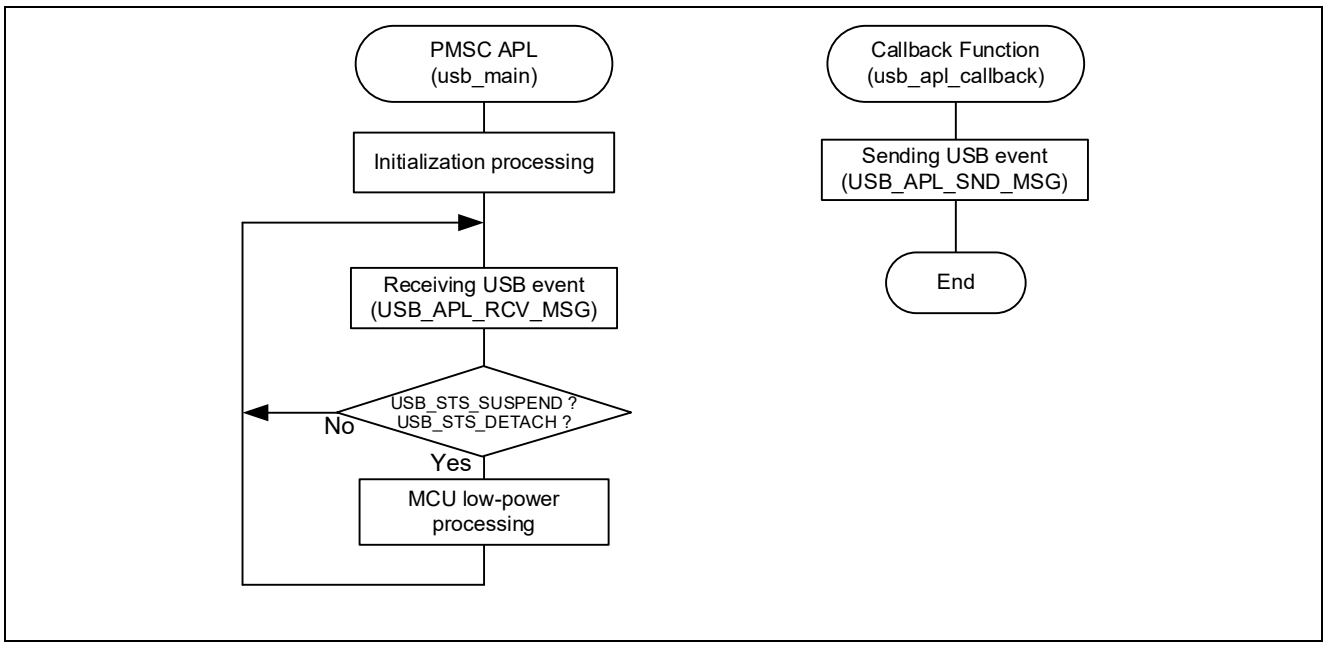


Figure 4-4 APL Processing Overview

4.4 Descriptor

The PMSC's descriptor information is contained in *r_usb_pmsc_descriptor.c*. Also, please be sure to use your vendor ID.

4.5 MCU Low power consumption processing

MCU low-power processing occurs when the conditions in Table 4-1 or Table 4-2 are met, causing a transition to low-power mode. To enable this processing, specify *USB_APL_ENABLE* to *USB_SUPPORT_LPW* definition in the *r_usb_pmsc_apl_config.h file*.

1. Non-OS

Table 4-1 Conditions for Transition to Low-Power Mode

Transition Condition		Transition Status
VBUS	USB State	
OFF	—	Software standby mode
ON	Suspend Configured	Sleep mode
ON	Other than Suspend Configured	Normal mode (program running)

(1). When the MSC device (RSK) detaches from the USB host (VBUS OFF), the APL performs processing to transition the MCU to software standby mode. Recovery from software standby mode occurs when the MSC device (RSK) attaches to the USB host.

(2). When a suspend signal sent by the USB host is received while the MSC device (RSK) is connected to the USB host, the APL performs processing to transition the MCU to sleep mode. Note that recovery from sleep mode occurs when a resume signal is received from the USB host.

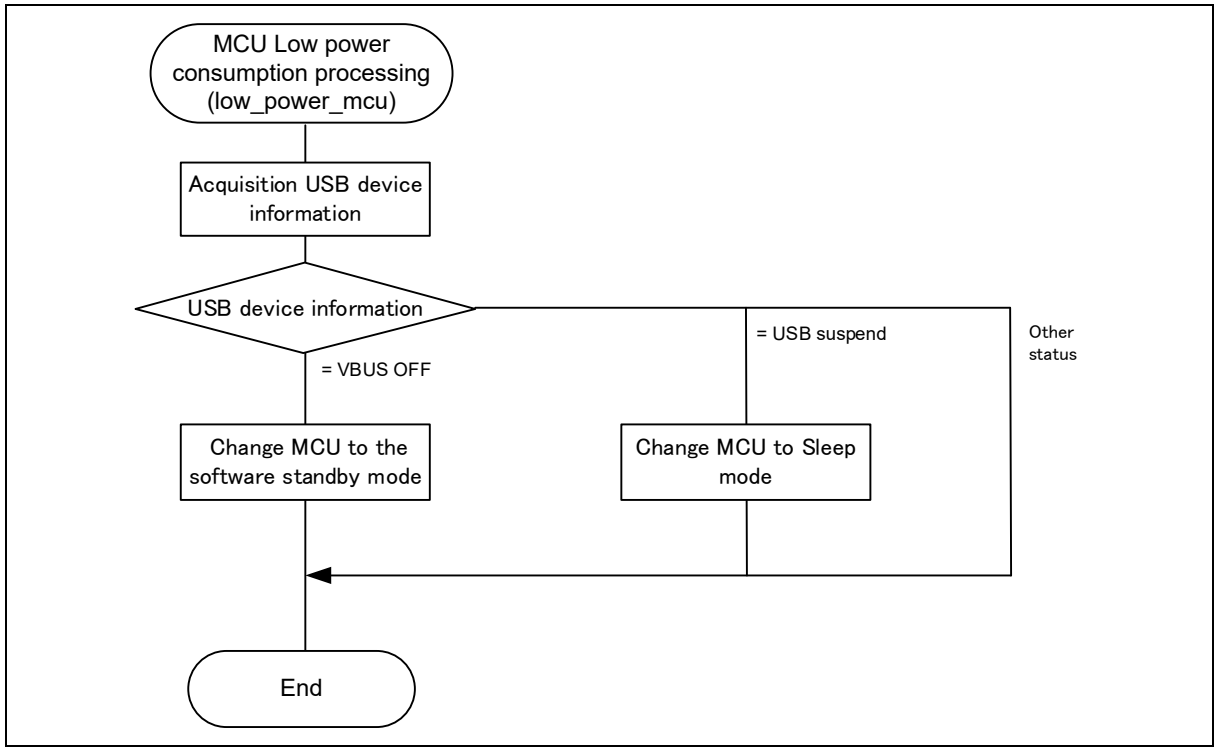


Figure 4-5 Flowchart of MCU Low Power Consumption Processsing

2. RTOS (FreeRTOS only)

Table 4-2 Conditions for Transition to Low-Power Mode

Transition Condition		Transition Status
VBUS	USB State	
OFF	—	Software standby mode
ON	Suspend Configured	Software standby mode
ON	Other than Suspend Configured	Normal mode (program running)

(1). When the CDC device (RSK) detaches from the USB host (VBUS OFF), the APL performs processing to transition the MCU to software standby mode. Recovery from software standby mode occurs when the CDC device (RSK) attaches to the USB host.

(2). When a suspend signal sent by the USB host is received while the CDC device (RSK) is connected to the USB host, the APL performs processing to transition the MCU to software standby mode. Note that recovery from softwre standby mode occurs when a resume signal is received from the USB host.

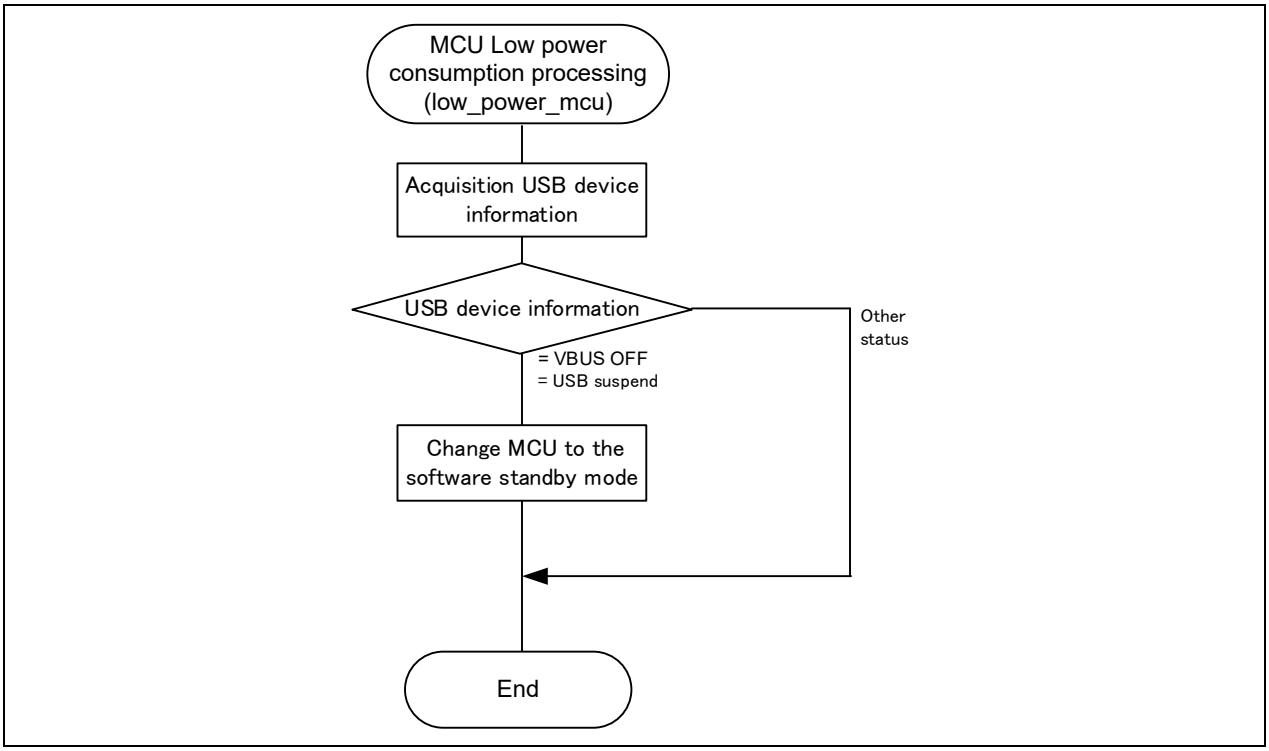


Figure 4-6 Flowchart of MCU Low Power Consumption Processsing

4.6 Configuration File for the application program (r_usb_pcdc_apl_config.h)

Make settings for the definitions listed below.

1. USE_USBIP Definition

Specify the module number of the USB module you are using. Specify one of the following settings for the *USE_USBIP* definition.

```
#define    USE_USBIP    USE_USBIP0    // Specify USB_IP0.
#define    USE_USBIP    USE_USBIP1    // Specify USB_IP1.
```

[Note]

You can specify *USE_USBIP1* when using RX64M or RX71M. Specify *USE_USBIP0* when using the MCU other than RX64M and RX71M.

2. USB_SUPPORT_SPEED Definition

Specify the USB operation speed (Hi-speed or Full-speed) which CDC device (RSK) supports.

```
#define    USB_SUPPORT_SPEED    HI_SPEED      // Hi-Speed setting
#define    USB_SUPPORT_SPEED    FULL_SPEED    // Full-Speed setting
```

[Note]

You can specify *HI_SPEED* when using RX71M. Specify *FULL_SPEED* when using the MCU other than RX71M.

3. Low-Power Function Definition

Specify whether or not the low-power function will be used. If the low-power function will be used, speify *USB_APL_ENABLE* to *USB_SUPPORT_LPW* definition.

```
#define    USE_SUPPORT_LPW    USB_APL_DISABLE    // No use the low-power function
#define    USB_SUPPORT_LPW    USB_APL_ENABLE     // Use the low-power function
```

4. USB_SUPPORT_RTOS Definition

Please specify *USB_APL_ENABLE* to *USB_SUPPORT_RTOS* definition when using the real-time OS.

```
#define    USB_SUPPORT_RTOS    USB_APL_DISABLE    // No use the real-time OS
#define    USB_SUPPORT_RTOS    USB_APL_ENABLE     // Use the real-time OS
```

5. Note

The above configuration settings apply to the application program. USB driver configuration settings are required in addition to the above settings. For information on USB driver configuration settings, refer to the application note *USB Basic Host and Peripheral Driver Firmware Integration Technology* (Document number. R01AN2025EJ).

5. Class Driver Overview

5.1 Class Request

Table 5-1 lists the class requests supported by the PMSC.

Table 5-1 Supported MSC Class Requests

Request	Code	Description
Bulk-Only Mass Storage Reset	0xFF	Resets the connection interface to the mass storage device.
Get Max Lun	0xFE	Reports the logical numbers supported by the device.

5.2 Storage Commands

The PMSC supports storage commands necessary to access an USB mass storage device as well as sample storage commands. The MSDD of the PMSC supports the SFF-8070i USB mass storage subclass.

Table 5-2 lists the storage commands supported by the PMSC.

Table 5-2 Supported Storage Commands

Command	Code	Description	Supported
TEST_UNIT_READY	0x00	Checks the state of the peripheral device.	○
REQUEST_SENSE	0x03	Gets the state of the peripheral device.	○
FORMAT_UNIT	0x04	Formats the logical unit.	×
INQUIRY	0x12	Gets the parameter information of the logical unit.	○
MODE_SELECT6	0x15	Specifies parameters.	×
MODE_SENSE6	0x1A	Gets the parameters of the logical unit.	×
START_STOP_UNIT	0x1B	Enables/disabled logical unit access.	×
PREVENT_ALLOW	0x1E	Enables/disabled media removal.	×
READ_FORMAT_CAPACITY	0x23	Gets the formattable capacity.	○
READ_CAPACITY	0x25	Gets the capacity information of the logical unit.	○
READ10	0x28	Reads data.	○
WRITE10	0x2A	Writes data.	○
SEEK	0x2B	Moves to a logical block address.	×
WRITE_AND_VERIFY	0x2E	Writes data with verification.	×
VERIFY10	0x2F	Verifies data.	×
MODE_SELECT10	0x55	Specifies parameters.	×
MODE_SENSE10	0x5A	Gets the parameters of the logical unit.	○

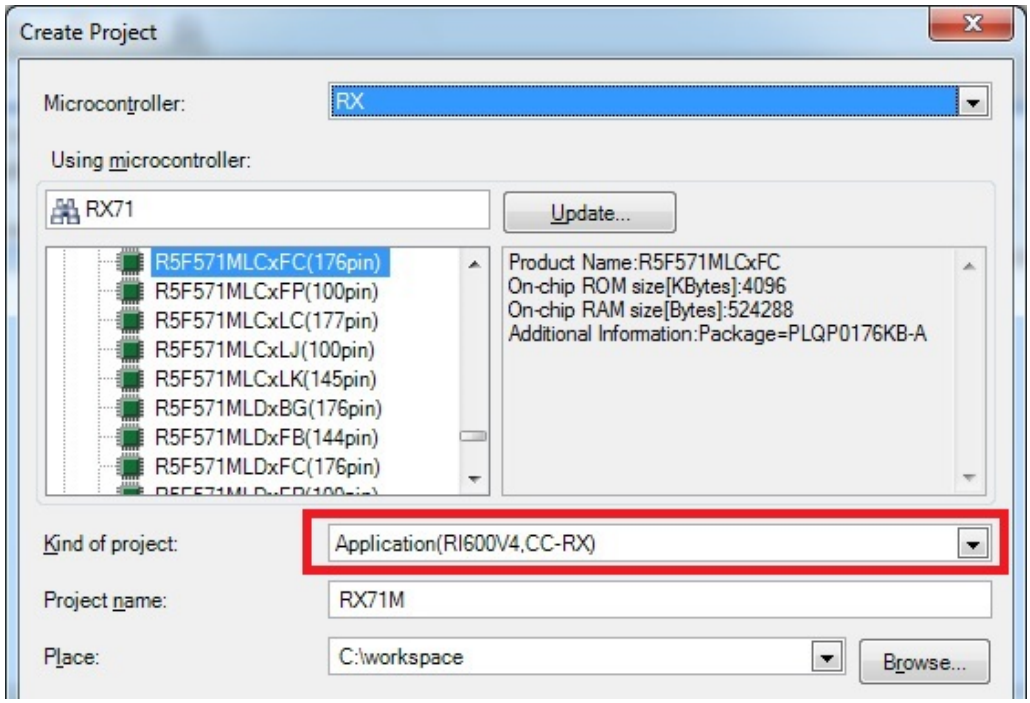
○ : Implemented × : Not implemented(Stall response)

6. Using RI600V4 project with CS+

The RI600V4 project in the package does not support CS+. The user needs to create a project for CS+ according to the following procedure when using RI600V4 project on CS+.

6.1 New Project Creation

Select "Application(RI600V4, CC-RX) for the Kind of project.



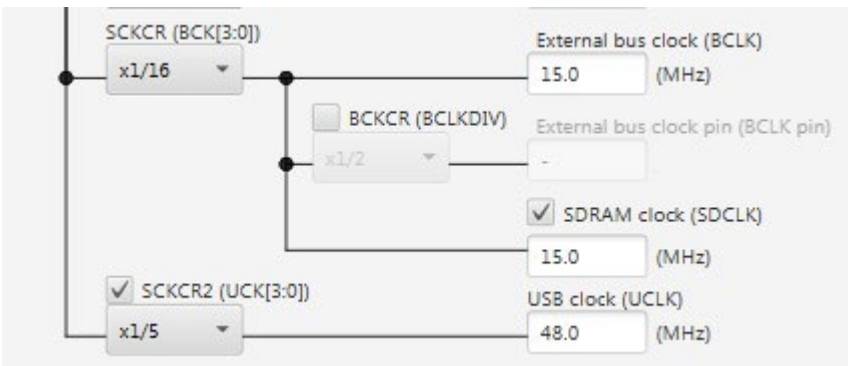
6.2 Launch Smart Configurator

1. Clock Setting (Select "Clocks" tab)

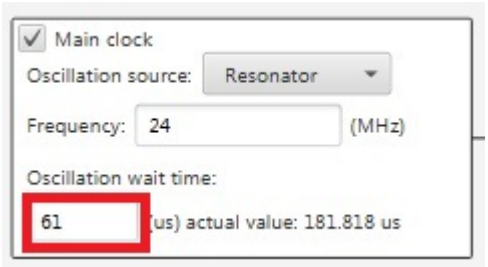
(1). Set the related clock so that "48MHz" is set to UCLK (USB clock).

Note:

Set the related clock so that "15MHz" is set to SDRAM (SDCLK) when using the RSK.



(2). Set "61" to the oscillation wait time for the main clock.



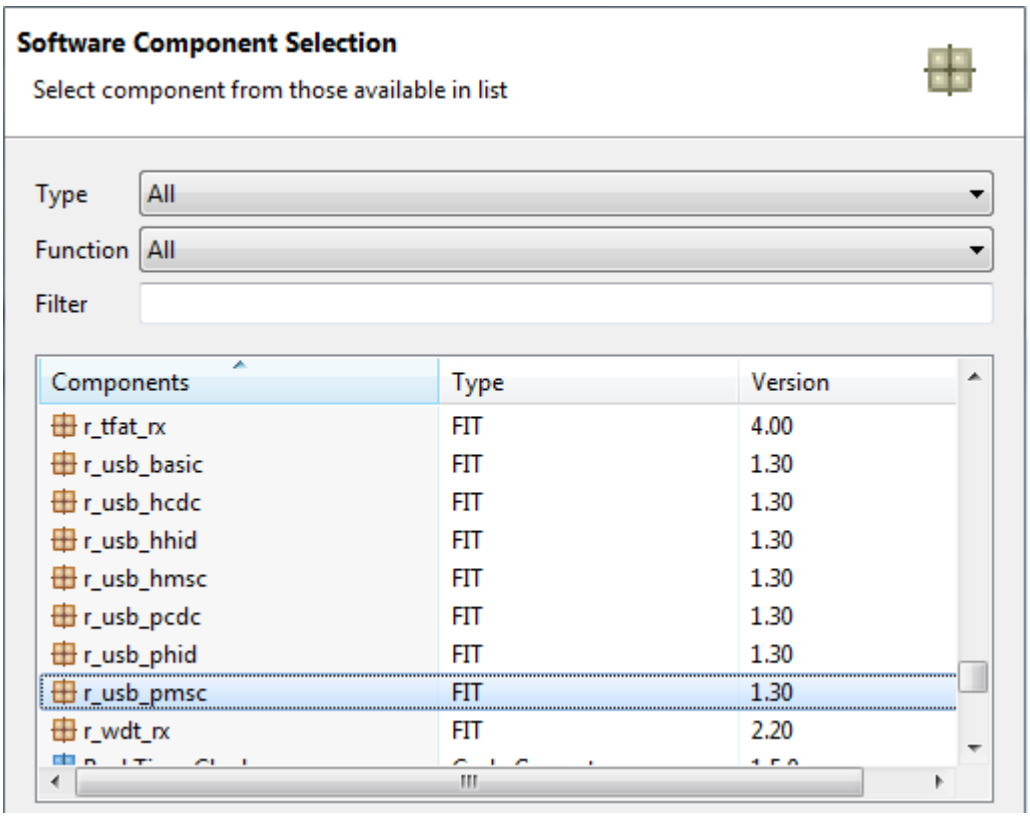
2. Component Setting (Select "Components" tab)

(1). Import the USB FIT module

Select the *r_usb_pmsc* module and press the "Finish" button. The *r_usb_basic* module is imported at the same time.

Note:

Select the *r_dtc_rx* / *r_dmaca_rx* module when using the DTC/DMA.

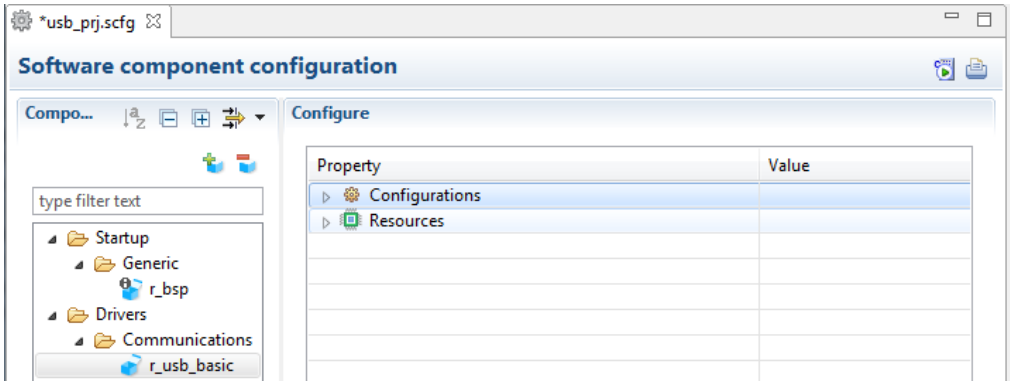


(2). Configuration Setting

a. r_bsp

Change the heap size when using DTC transfer. For the setting value, refer to the documentation for DTC FIT module.

b. r_usb_basic



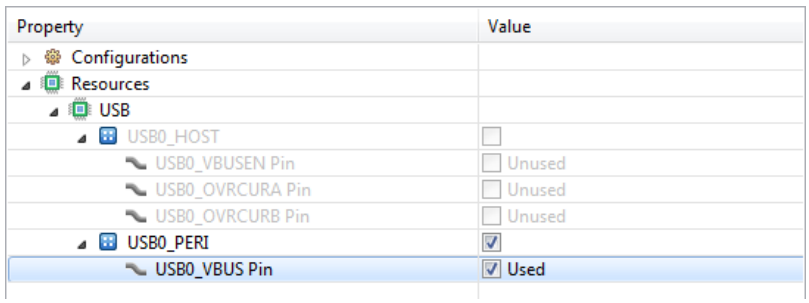
(a). Configurations

Set each item according to the user system.

For the detail of each item, refer to chapter "Configuration" in *USB Basic Host and Peripheral Driver Firmware Integration Technology* application note (Document number: R01AN2025).

(b). Resources

Check the check box for USBx_VBUS pin.

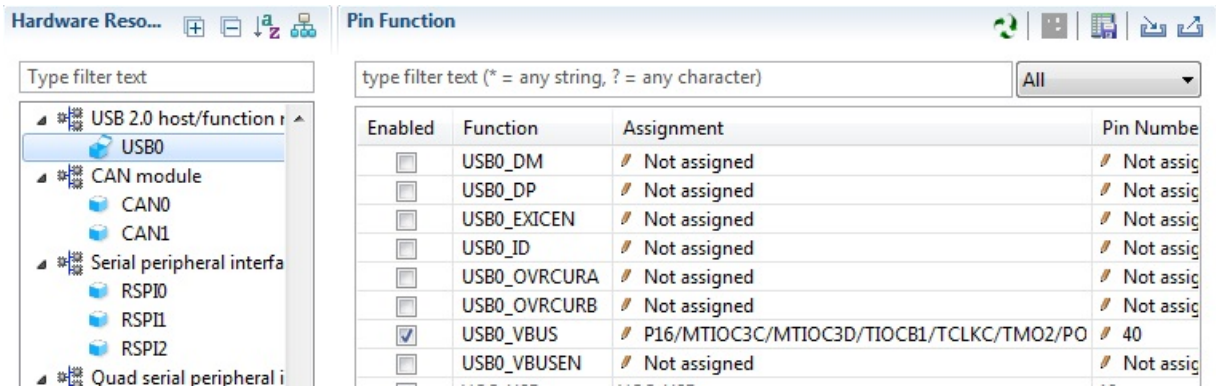


c. r_usb_pmsc

Refer to chapter "Configuration" in *USB Peripheral Mass Storage Class Driver (PCDC) Firmware Integration Technology* application note (Document number: R01AN2029).

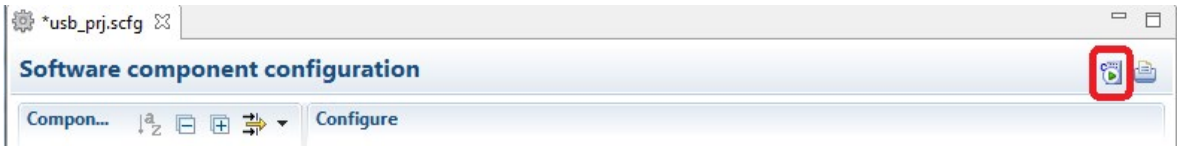
3. Pin Setting (Select "Pins" tab)

Select the port for USB pin match the user system.



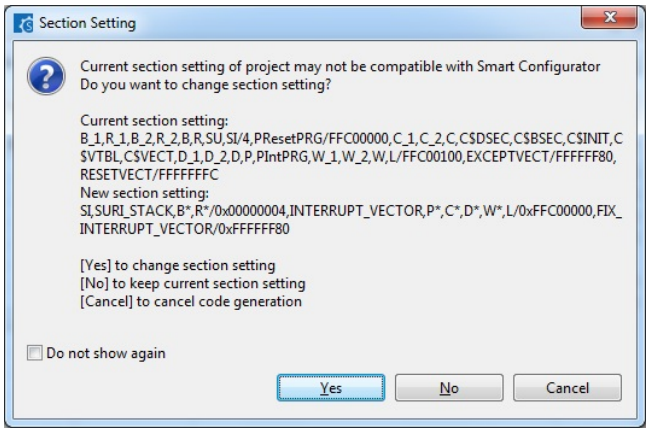
4. Generate Code

The Smart Configurator genrates source codes for USB FIT module and USB pin setting in "*<ProjectDir>¥src¥smc_gen*" folder by by clicking on the [(Generate Code)] button.



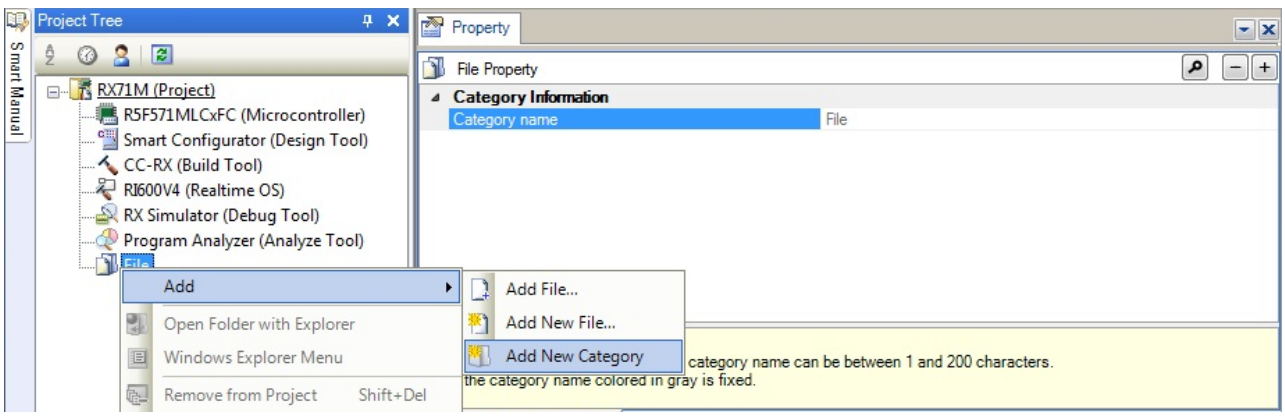
Note:

Select "Yes" if the following dialog box is displayed.



6.3 Add the application program and the configuration file

1. Copy the *demo_src* folder in this package to the *"<ProjectDir>¥src"* folder.
2. Copy the RI600V4 configuration file (.cfg file) to *"<ProjectDir>"* folder.
3. Select "File" in the "Project Tree" and click the right button. Select [Add] → [Add New Category] and create the category to store the application program. Then select [Add File] and register the application program and the configuration file which are copied at the above 2.



Note:

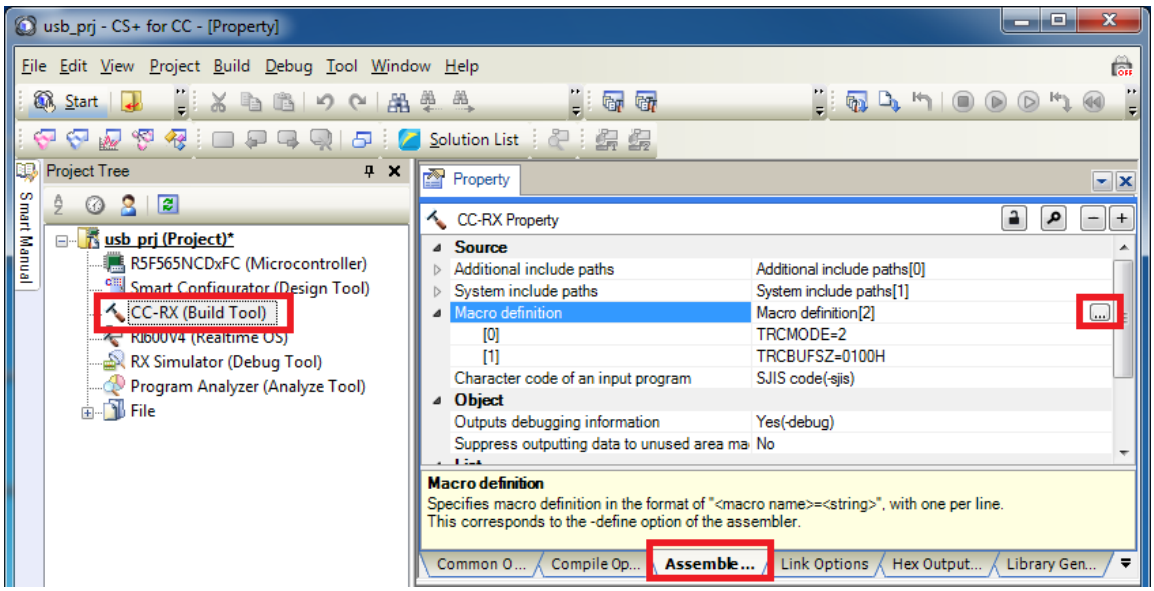
Remove the "task.c" file and "sample.cfg" created in "*<ProjectDir>*" folder by CS+.

6.4 Remote Macro Definition

Remove these macros since the following macros is defined in the new created project.

Select [CC-RX(Build Tool)] → [Assemble Options] tab, remove the following macros.

1. TRCMODE = 2
2. TRCBUFSZ = 0100H



6.5 Build Execution

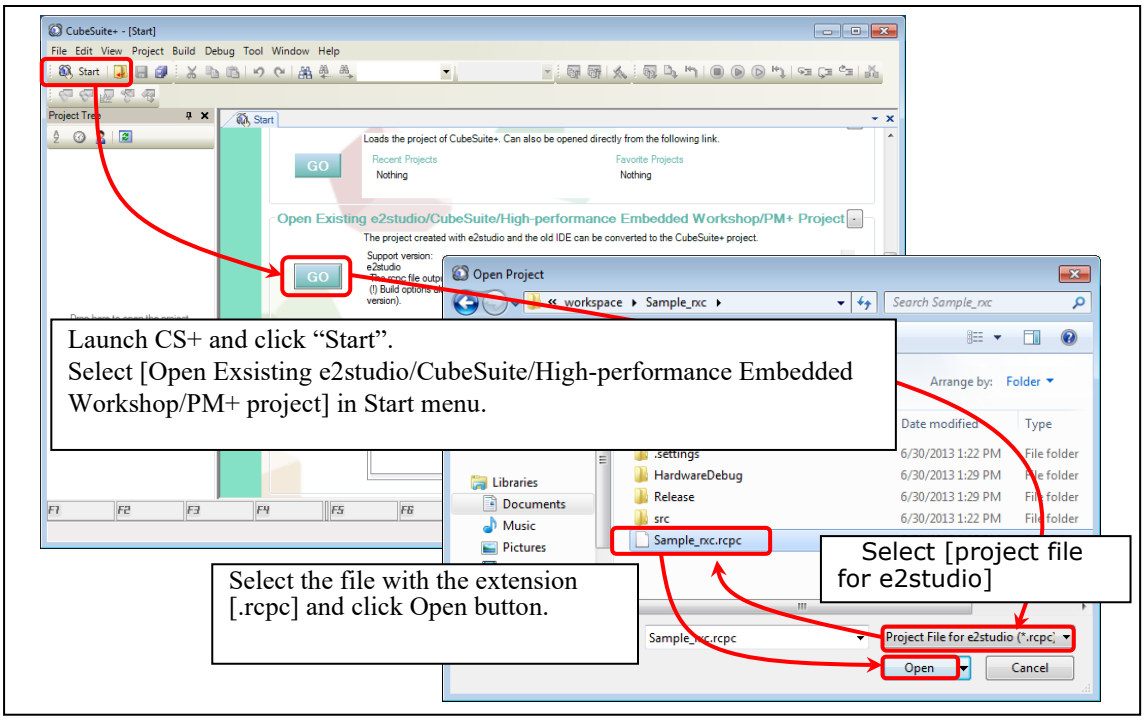
Excecute the build and generate the binary target program.

7. Using the e² studio project with CS+

The PMSC contains a project only for e² studio. When you use the PMSC with CS+, import the project to CS+ by following procedures.

[Note]

1. Uncheck the checkbox Backup the project composition files after conversion in Project Convert Settings window.
2. The following method is not supported when using RI600V4. Refer to chapter **6, Using RI600V4 project with CS+**.



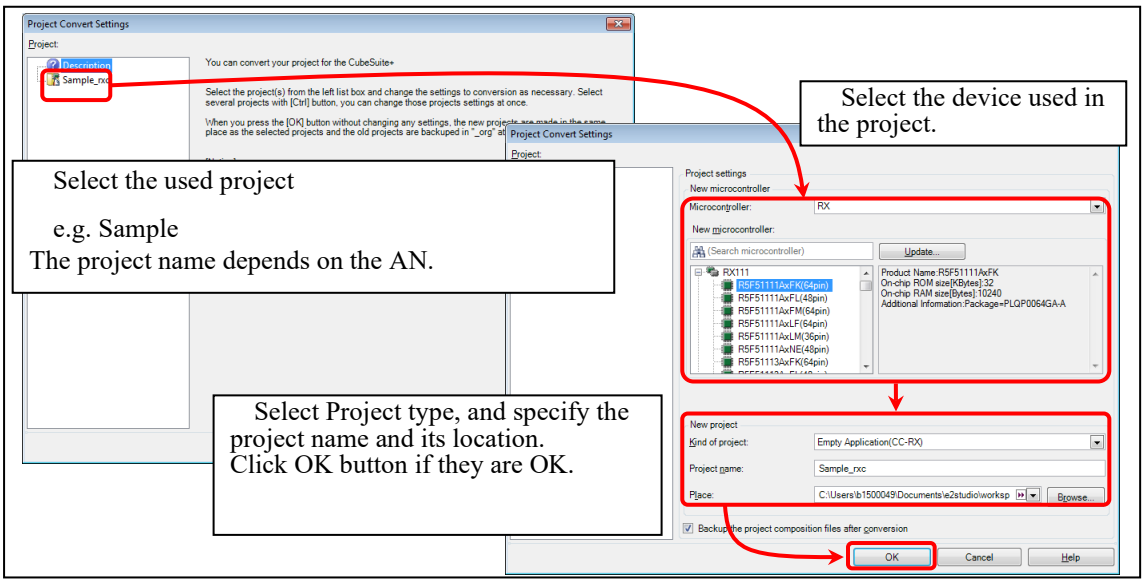


Figure 7-1 Using the e² studio project with CS+

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry/

All trademarks and registered trademarks are the property of their respective owners.

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Oct 16, 2014	—	First Edition Issued.
1.10	Dec 26, 2014	—	RX71M is added in Target Device
1.11	Sep 30, 2015	—	RX63N and RX631 are added in Target Device
1.20	Sep 30, 2016	—	1. RX65N and RX651 are added in Target Device. 2. Supporting DMA transfer. 3. Supporting USB Host and Peripheral Interface Driver application note
1.21	Mar 31, 2017	—	The revision of USB Basic driver has been updated.
1.22	Sep 30, 2017	—	Supporting RX65N/RX651-2M
1.23	Mar 31, 2018	—	The revision of USB Basic driver has been updated.
1.24	Dec 28, 2018	—	Supported the real-time OS.
1.25	Apr 16, 2019	—	Added RX66T/RX72T in Target Device.
1.27	Jul 31, 2019	—	1. RX72M is added in Target Device. 2. RX63N is removed from Target Device.
1.30	Mar 1, 2020	—	1. Supported the real time OS (uITRON:RI600V4). 2. Added RX72N/RX66N in Target Device.
1.31	Mar 1, 2021	—	Added RX671 in Target Device.
1.45	May 15, 2026	—	Included spdx file and json file in the package.

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/.

© 2026 Renesas Electronics Corporation. All rights reserved.