

RX Family

Adding Firmware Integration Technology Modules to CS+ Projects

R01AN1826EJ0110
Rev.1.10
Mar 23, 2017

Summary

This document describes the procedure to obtain the firmware integration technology (FIT) module and add it to CS+ projects.

Products

Supported devices depend on the FIT modules.

Refer to the device list in the readme.txt file provided with the FIT module to see the supported devices.

Contents

1. Overview	3
1.1 Terminology.....	3
1.2 FIT Structure.....	4
1.3 File Structure of the FIT Module Application Note	5
2. Manual Addition Procedure	7
2.1 Creating a CS+ Project	8
2.1.1 Create a New Project	8
2.1.2 Delete Unused Files.....	9
2.2 BSP Addition	11
2.2.1 Download the BSP	11
2.2.2 Add the BSP	11
2.2.3 Select the Platform	12
2.2.4 Select the Microcontroller.....	13
2.2.5 Set Up the Header File.....	13
2.2.6 Create the BSP Setup File (r_bsp_config.h)	14
2.2.7 Create the BSP Setup File (r_bsp_interrupt_config.h).....	14
2.3 Adding the FIT Module	15
2.3.1 Download the FIT Module	15
2.3.2 Add the FIT Module.....	15
2.3.3 Select the FIT Module	16
2.3.4 Create the FIT Module Setup File	17
2.4 FIT Module Registration	18
2.5 CC-RX Setup.....	19
2.5.1 Verify the Include Path	19
2.5.2 Change the C Language Standard	20
2.5.3 Set the Variable Vector (Interrupt Vector) Empty Area	21
2.5.4 Change the Section Address	22
2.6 Build the Project.....	24
3. Troubleshooting	25
4. Notes	27
5. FIT Module	27
6. Reference Documents	27

1. Overview

This application note describes how to manually add a FIT module in a CS+ project. Refer to the CS+ user's manual for information on how to use CS+.

1.1 Terminology

Term	Meaning
FIT	Acronym for firmware integration technology.
CS+	Integrated development environment CS+ (CS plus) offers the ultimate in simplicity, usability, and security for the repetitive editing, building and debugging that typifies software development.
BSP	Acronym for board support package. The BSP is a collection of programs that perform basic MCU settings like clock and pin settings. The BSP is the foundation of any project that uses FIT modules. The BSP includes source files for the various microcontroller groups and specific boards. The FIT module for the BSP is "r_bsp".
CMTW	Acronym for compare match timer W. The FIT module for the RX CMTW is "r_cmtw_rx".
DMACA	Acronym for direct memory access controller A. The FIT module for the RX DMACA is "r_dmaca_rx".
Platform	Means the same as target board.

1.2 FIT Structure

The FIT consists of BSP, peripheral, middleware, and interface modules.

- BSP: Module that performs microcontroller initialization, clock setup, and board setup.
- FIT peripheral function module: Driver that controls the RX microcontroller peripheral functions.
- FIT middleware module: Middleware that implements TCP/IP, file system, and other functions.
- FIT interface module: Interface that provides the Socket API and other functions.

Software development is made easier by using these components.

Figure 1.1 shows the FIT Structure.

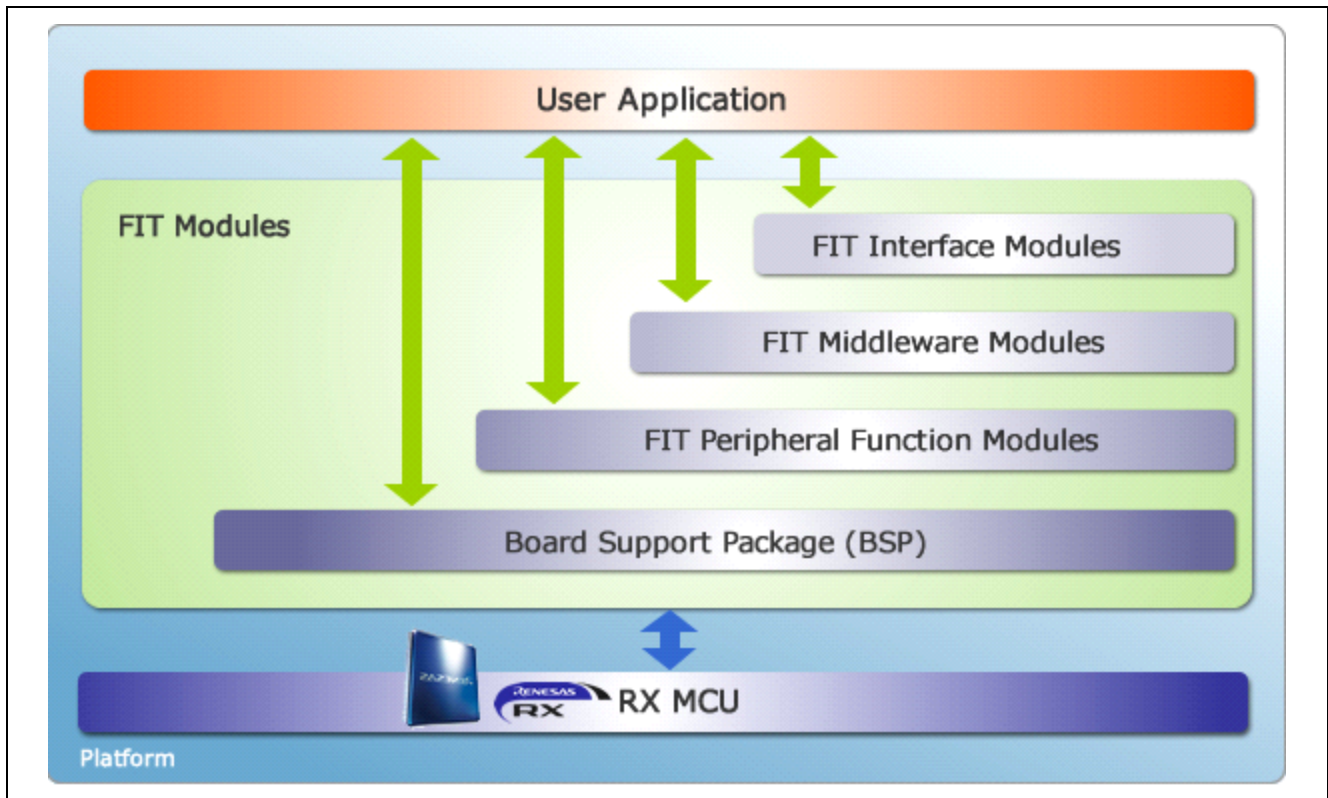


Figure 1.1 FIT Structure

1.3 File Structure of the FIT Module Application Note

The FIT module application note is a single ZIP file that can be downloaded from the Renesas website.

The FIT module application note includes common subsets for files and folders.

Figure 1.2 shows a dummy FIT module application note that includes the FIT module with the peripheral function “abc”. Figure 1.3 shows the Contents of the Dummy FIT Module “r_abc_rx”.

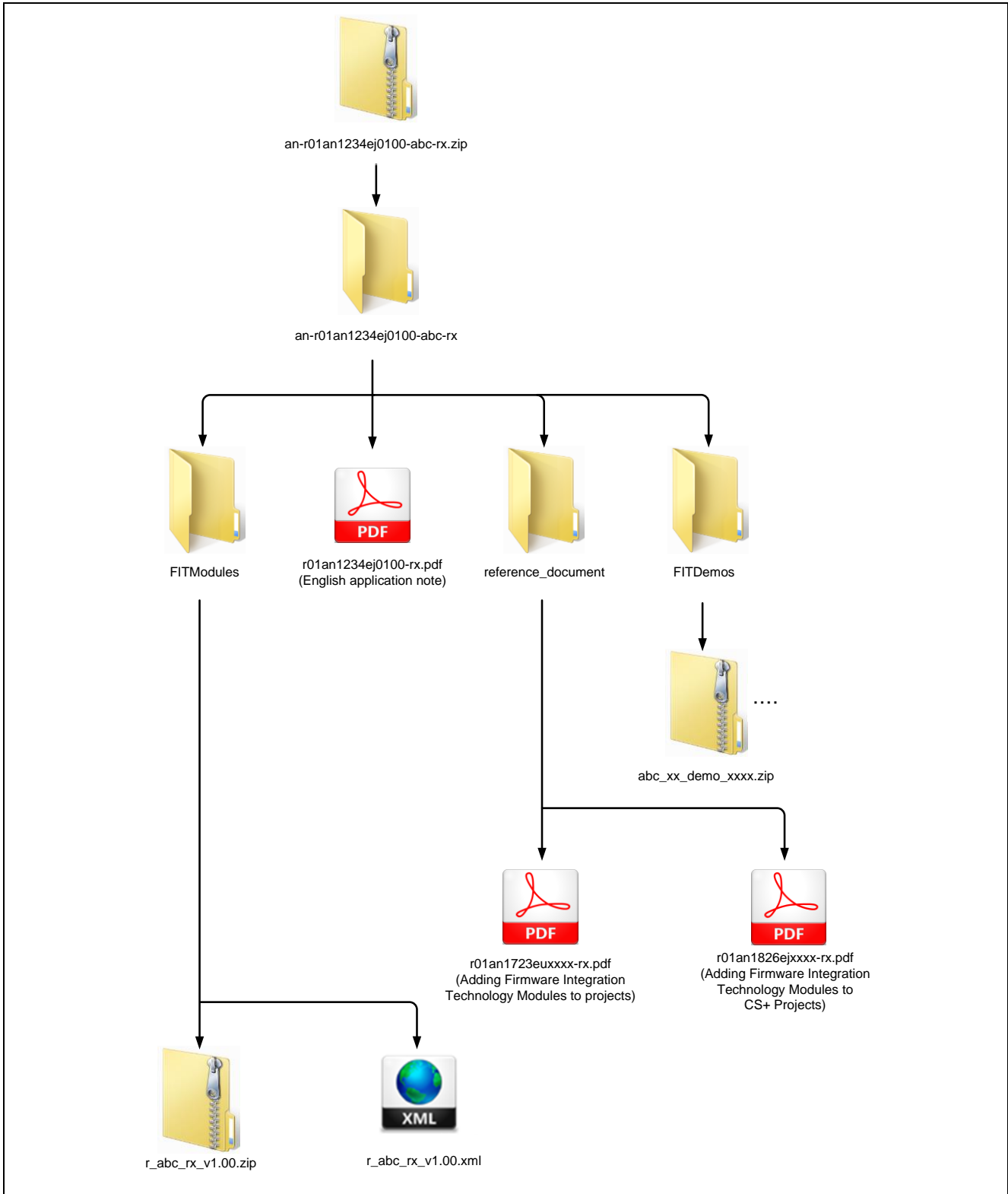


Figure 1.2 Contents of the FIT Module Application Note

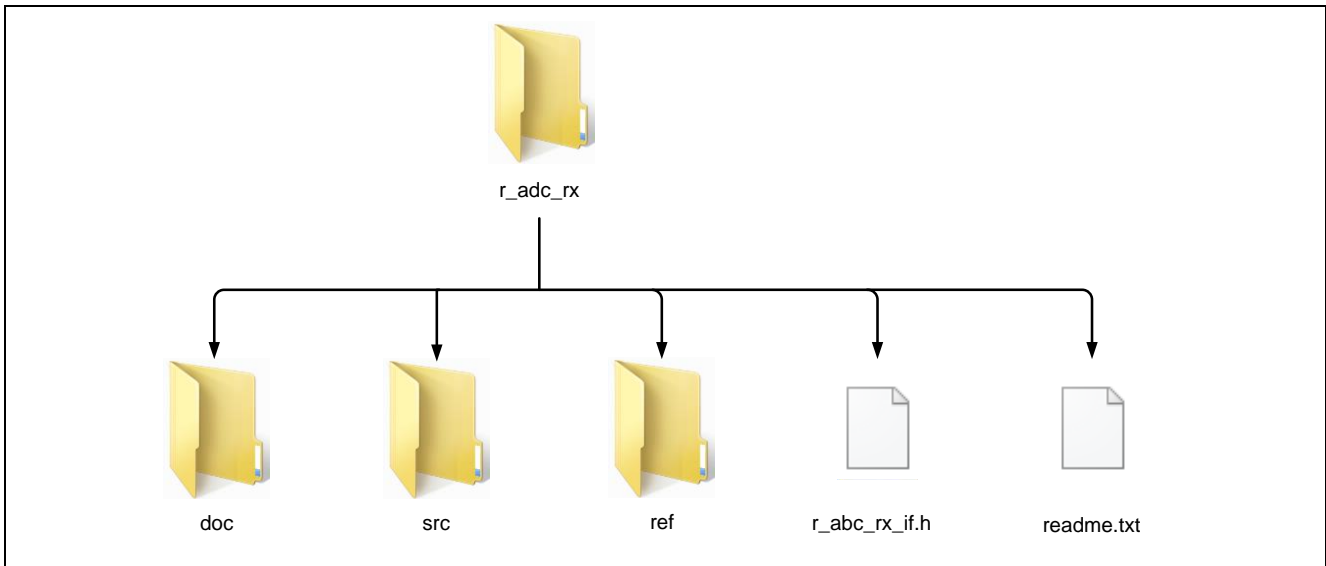


Figure 1.3 Contents of the Dummy FIT Module “r_abc_rx”

Figure 1.3 shows a typical file structure of a FIT module.

The names of any FIT modules begin with the prefix “r_” which represents the module is a Renesas module. The “r_” prefix is followed by the module function and “_rx” all in lower-case letters. “_rx” represents the RX Family.

The “doc” folder contains documents relating to the FIT module.

All of the source and header files required for the FIT module are stored in the “src” folder. The “src” folder may contain subfolders if necessary.

The “ref” folder contains configuration header files with the suffix “_config”. This is the FIT module configuration file. It includes several macros for figuring code to meet the user’s needs.

Two files are contained directly under the FIT module folder begin with the “r_”.

One is a header file with the suffix “_if”. This file includes an interface (e.g. prototype declarations for the API functions, type definitions, macros) required for using the FIT module.

The other is the readme.txt file. This file includes information such as the FIT module version and functions.

2. Manual Addition Procedure

This chapter explains how to create a new CS+ project and how to add the BSP and CMTW FIT modules.

This chapter assumes the Renesas StarterKit+ for RX64M (RSK+RX64M) platform is used. Take appropriate measures when using a different platform.

Figure 2.1 shows the steps for Configuring a New Project.

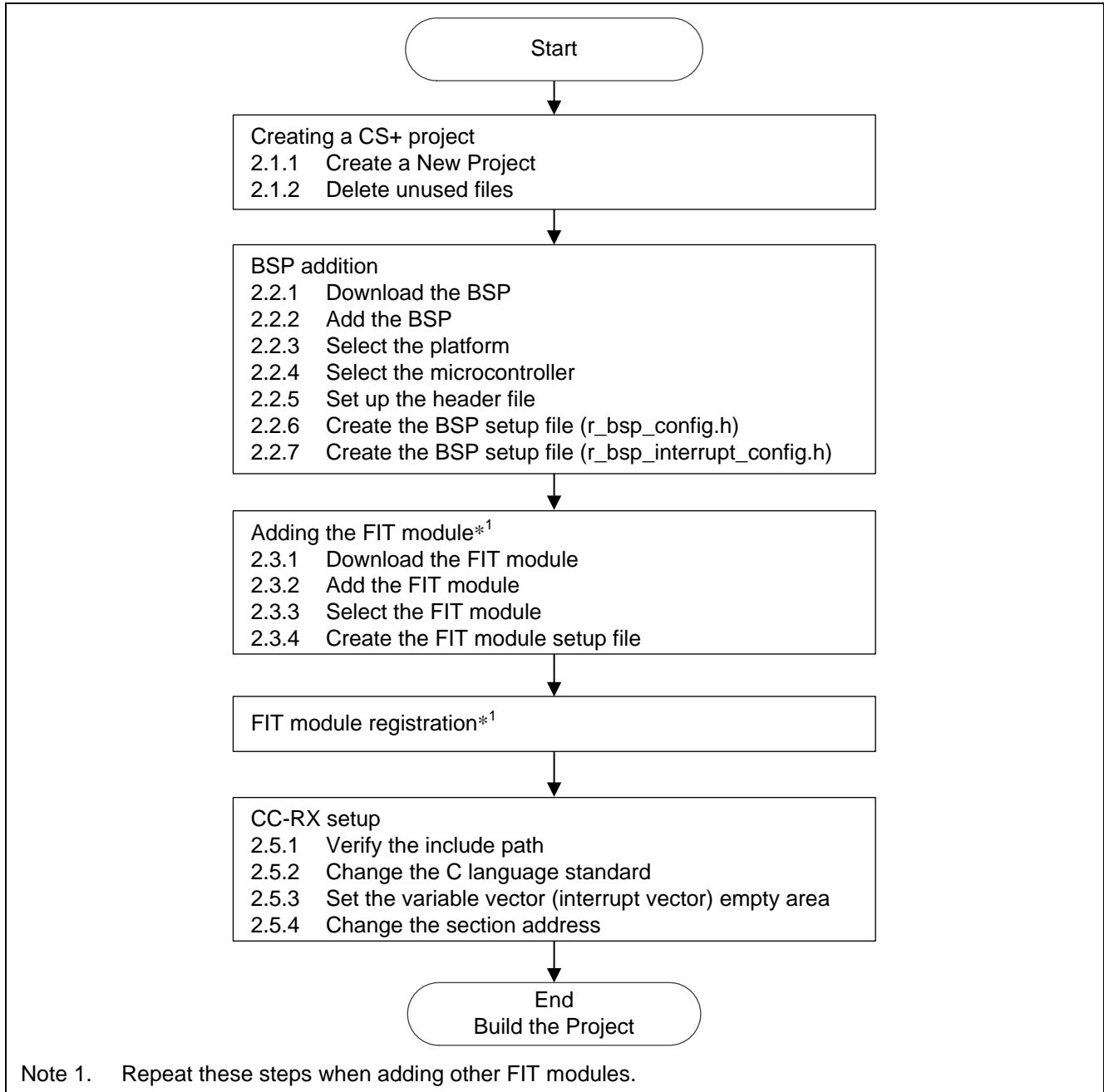
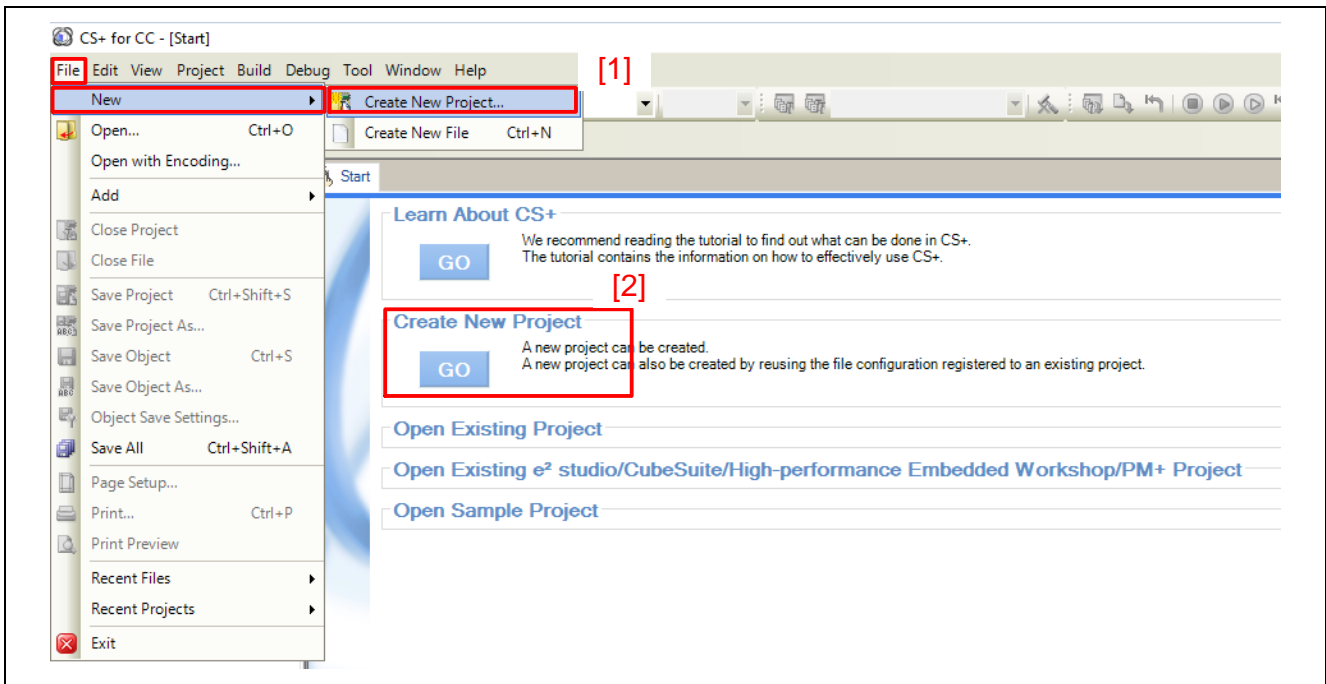


Figure 2.1 Configuring a New Project

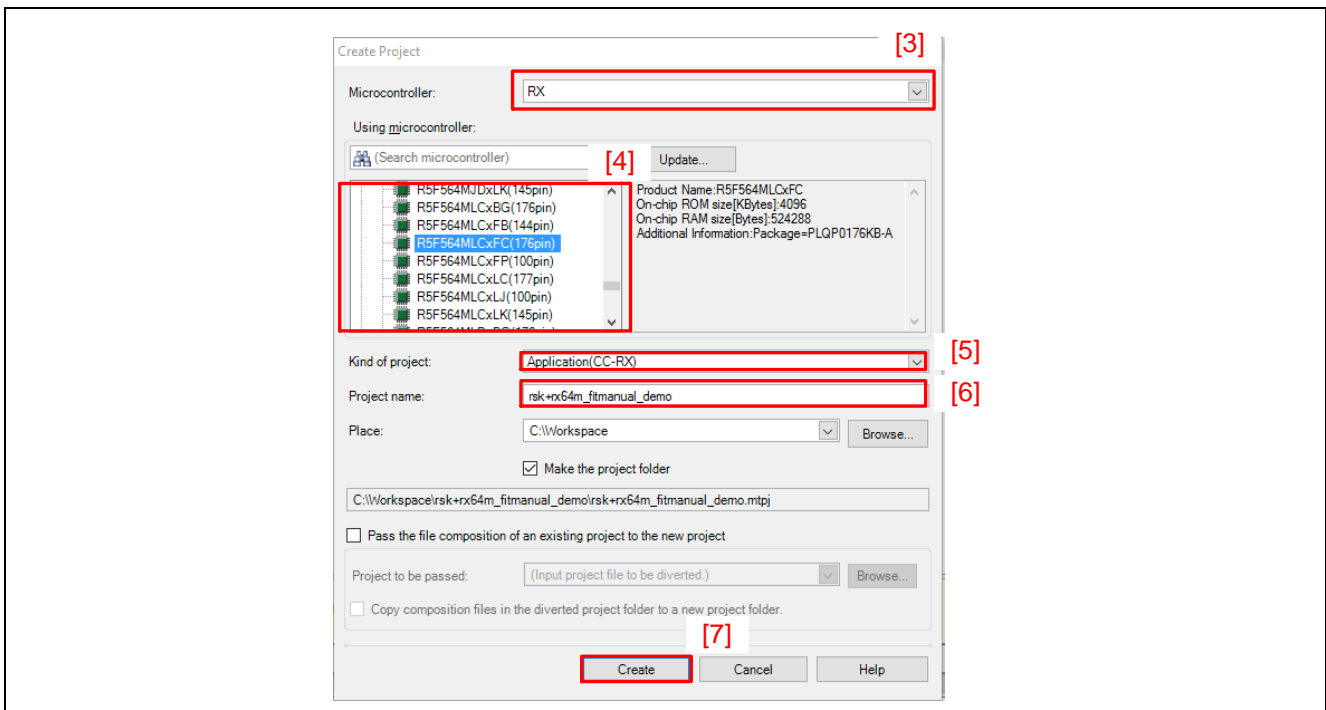
2.1 Creating a CS+ Project

2.1.1 Create a New Project

1. Start CS+, and open the project creation dialog by either selecting [1], from the menu bar, File(F) → New(N) → Create New Project(N), or clicking [2] the GO button in the New Project Creation Area in the Start Panel.



2. Select RX as the Microcontroller(T). [3]
3. Select the Microcontroller used(M) to match the platform you are using. [4]
In the example, the R5F564MLCxFC (176 pins) used in the RSK+RX64M is selected.
4. Select Application (CC-RX) as the Project type(K). [5]
5. Enter the project name to be created in the Project name(N). [6]
In the example, rsk+rx64m_fitmanual_demo is used. In the following descriptions, note that <rsk+rx64m_fitmanual_demo> will be replaced by the name of the users project when actually using this system.
6. Click the Create(C) button to create the project. [7]



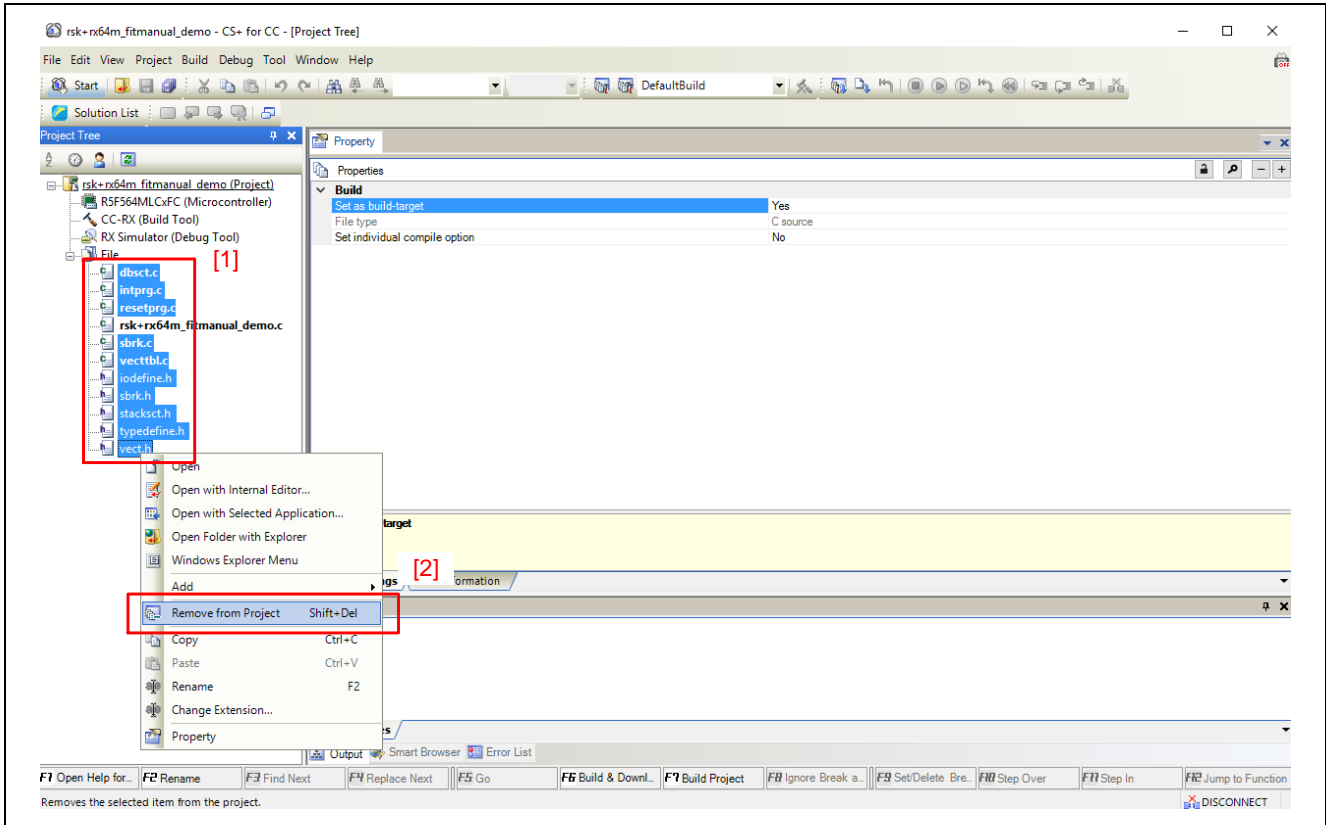
2.1.2 Delete Unused Files

Since the files generated when a new project are created overlap with those in the board support package (BSP) used, those files are not required. Therefore, the following should be run to delete the unnecessary files. Note that this operation does not need to be performed in this order.

Remove the files that do not include a main() function from the CS+ project

1. Select the files other than <rsk+rx64m_fitmanual_demo>.c in the project tree. [1]
2. With the files still selected, right click with the mouse and select Remove from project(R). [2]

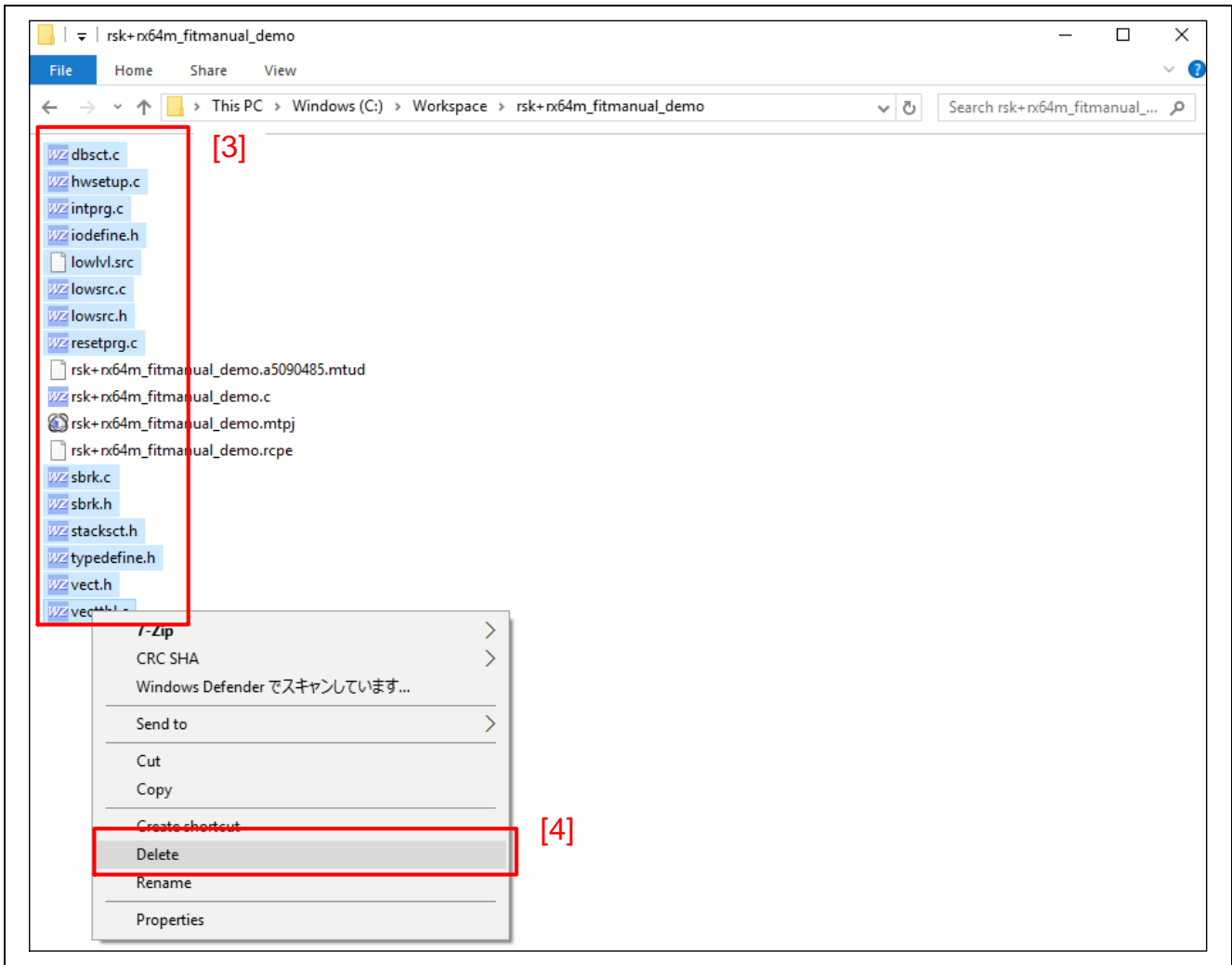
In the example, the files dbstc.c, intrpg.c, resetprg.c, sbrkc.c, vecttbl.c, iodefine.h, sbrk.h, stacksct.h, typedefine.h, and vect.h are removed from the project.



Delete the files that do not include a main() function

1. In Windows Explorer, select the files that do not start with <rsk+rx64m_fitmanual_demo>. [3]
2. With the files still selected, right click with the mouse and select Delete(D). [4]

In the example, the files dbstct.c, hwsetup.c, intprg.c, iodefne.h, lowlvl.src, lowsrc.c, lowsrc.h, resetprg.c, sbrk.c, sbrk.h, stacksct.h, typedefne.h, vect.h, and vecttbl.c, are deleted.



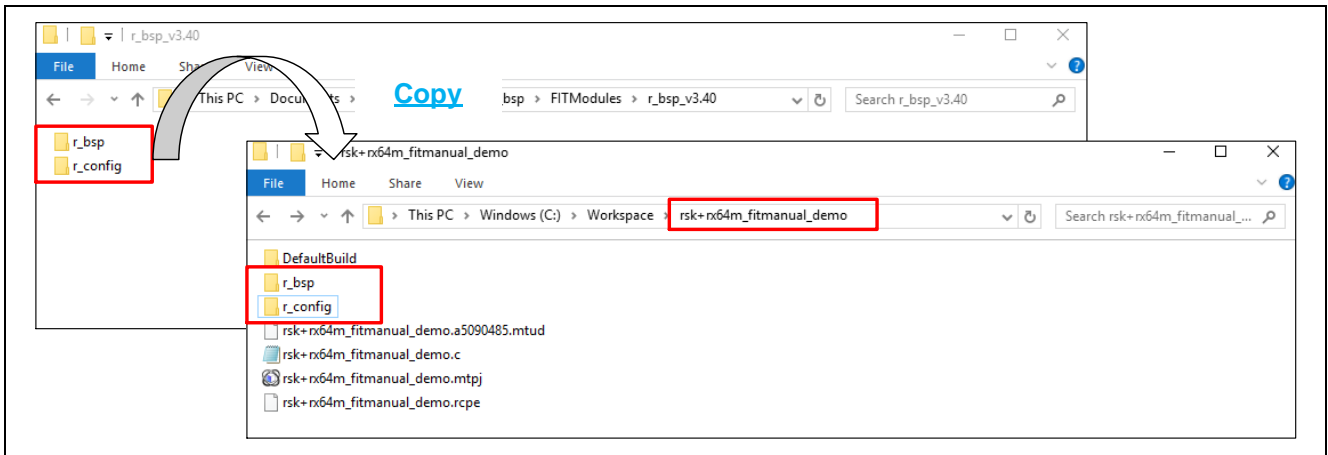
2.2 BSP Addition

2.2.1 Download the BSP

Download the RX Family Board Support Package Module Using Firmware Integration Technology (r01an1685EJxxxx) package from the Renesas Electronics Corporation web site. This package may be downloaded to any folder on the PC you are using.

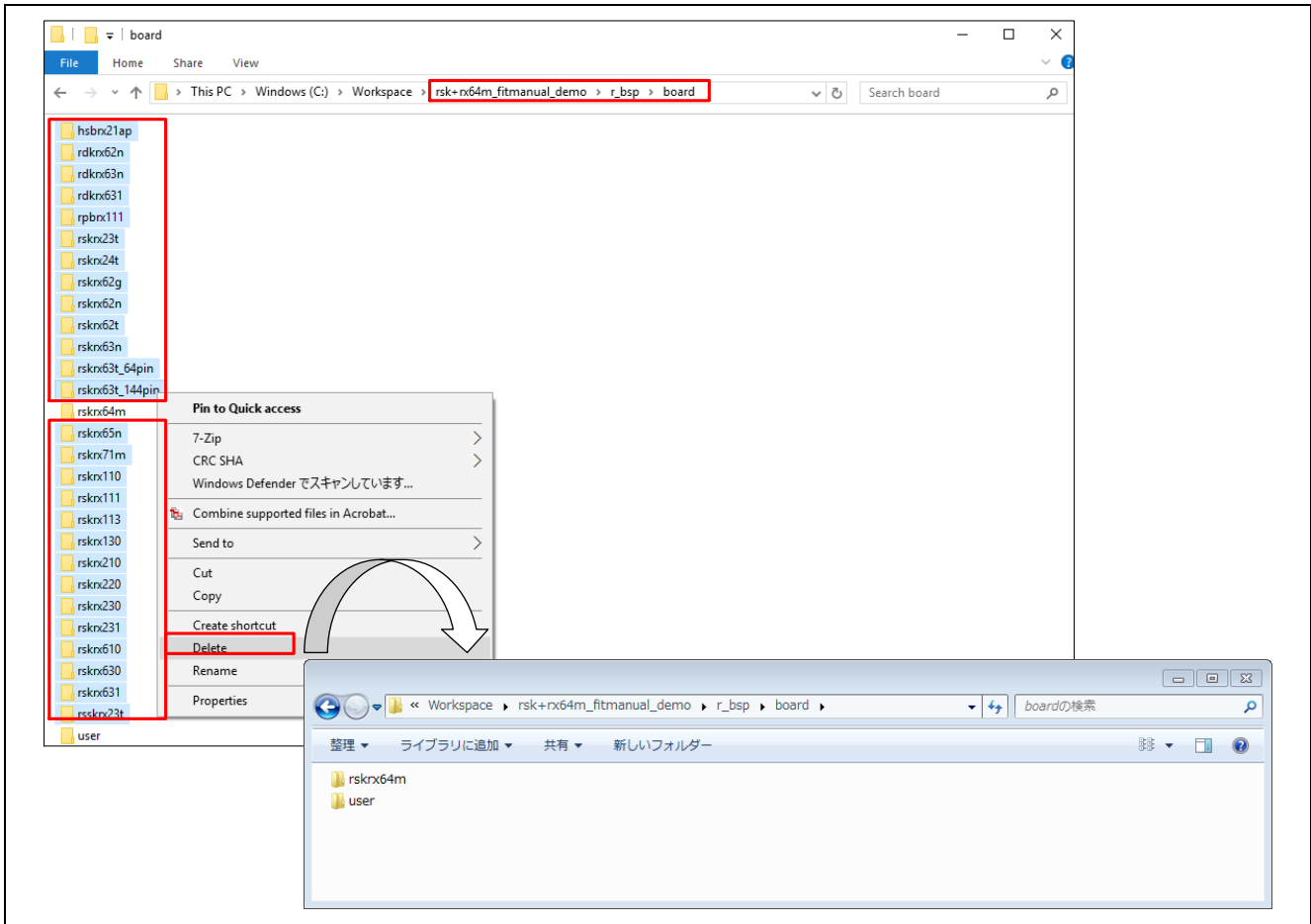
2.2.2 Add the BSP

Using Windows Explorer, copy the `r_bsp` and `r_config` folders from the file you just downloaded to the `<rsk+rx64m_fitmanual_demo>` folder.



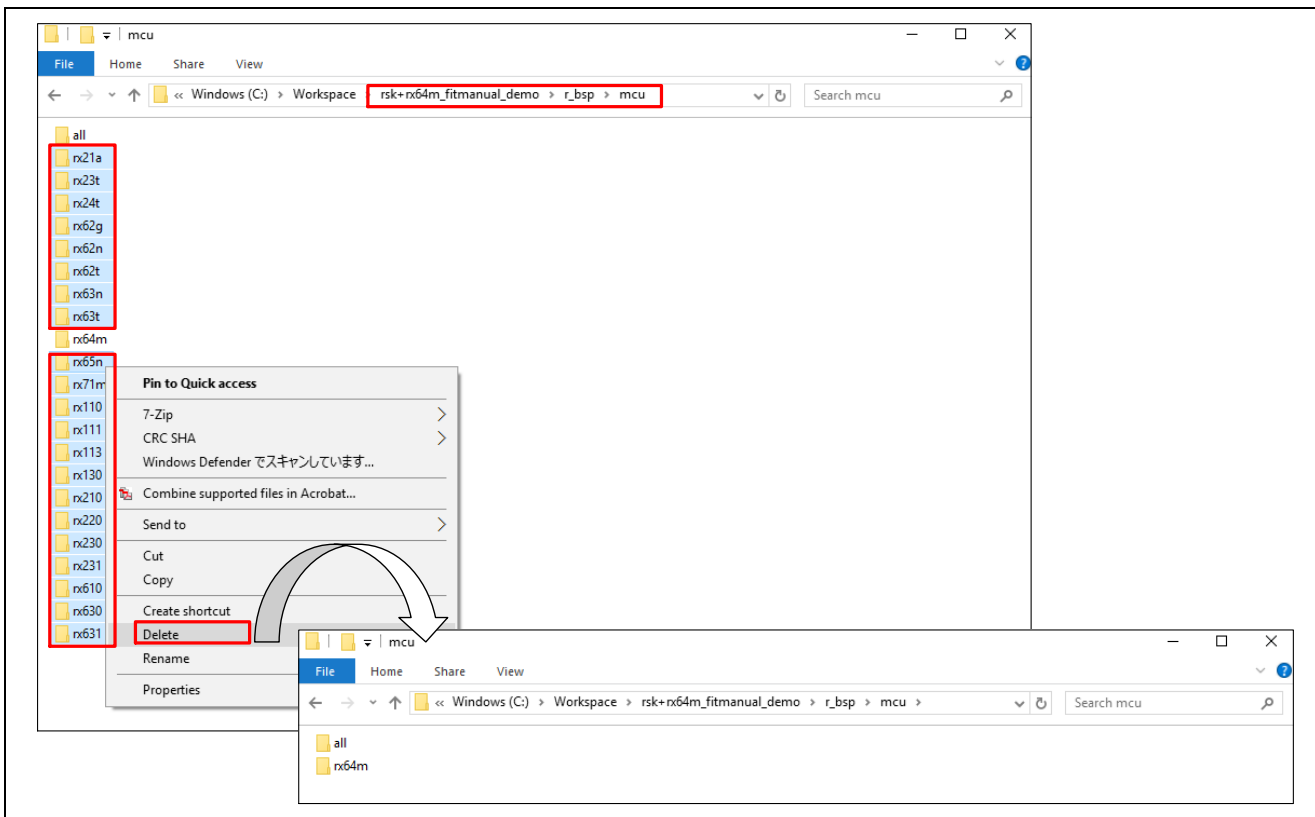
2.2.3 Select the Platform

In Windows Explorer, move to the `r_bsp/board` folder, and delete all the folders other than the user folder and the `rskrx64m` folder. If you are using a platform other than RSK+RX64M, instead of the `rskrx64m` folder, leave the platform folder that corresponds to your platform remaining.



2.2.4 Select the Microcontroller

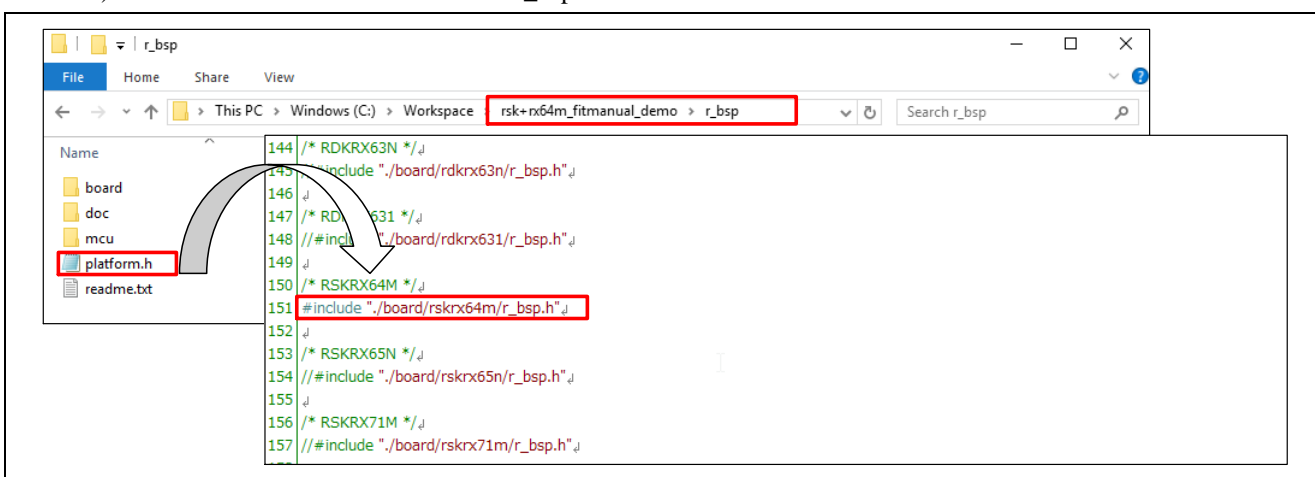
In Windows Explorer, move to the `r_bsp/mcu` folder, and delete all the folders other than the `all` folder and the `rx64m` folder. If you are using a microcontroller other than the RX64M, instead of the `rx64m` folder, leave the microcontroller folder that corresponds to your microcontroller remaining.



2.2.5 Set Up the Header File

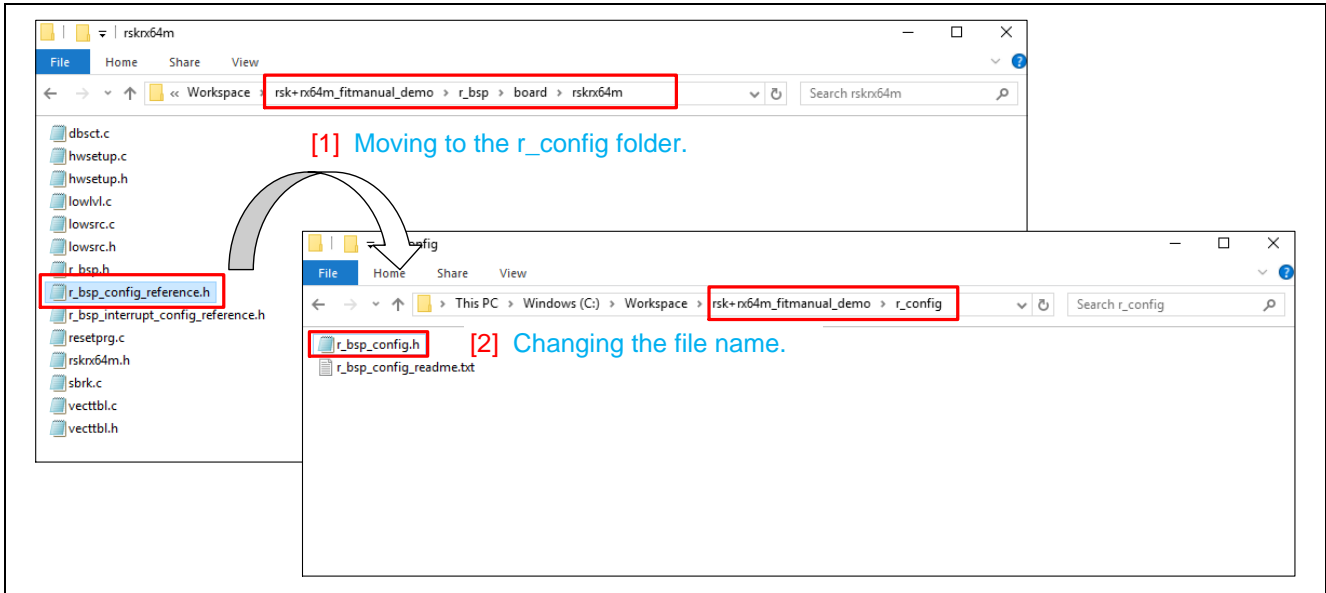
1. Edit the `platform.h` file in the `r_bsp` folder and select the platform you are using.
2. Open the `platform.h` file and remove the comment on the `#include` directive that specifies the platform that you are using.

For example, to select RSK+RX64M, remove the comment (that is, remove the `/**` characters at the start of the line) from the `#include` `"/board/rskrx64m/r_bsp.h"` directive.



2.2.6 Create the BSP Setup File (r_bsp_config.h)

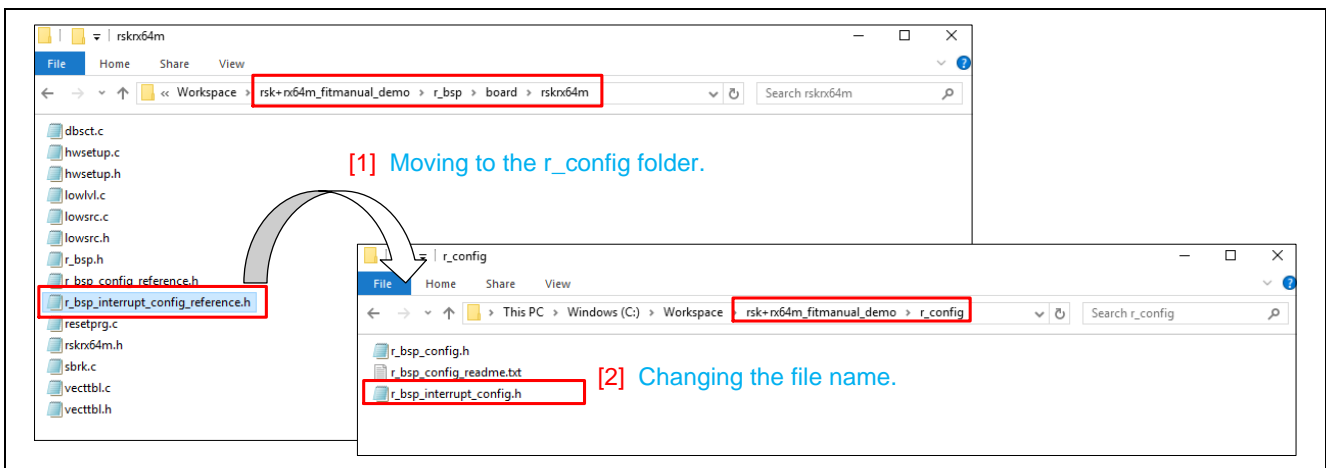
1. Using Windows Explorer, move the r_bsp_config_reference.h file in the r_bsp/board/rskrx64m folder to the r_config folder. [1]
2. Change the name of the moved r_bsp_config_reference.h file to be r_bsp_config.h. [2]
3. If you are using a platform other than RSK+RX64M, create an r_bsp_config.h that matches the platform used. See the BSP module document in the r_bsp/doc folder for the contents of the r_bsp_config.h file.



2.2.7 Create the BSP Setup File (r_bsp_interrupt_config.h)

The contents of this chapter applies when the microcontroller used has a selectable interrupt function.

1. Using Windows Explorer, move the r_bsp_interrupt_config_reference.h file in the r_bsp/board/rskrx64m folder to the r_config folder. [1]
2. Using Windows Explorer, change the name of the moved r_bsp_interrupt_config_reference.h file to be r_bsp_interrupt_config.h. [2]
3. If you are using a platform other than RSK+RX64M, create an r_bsp_interrupt_config.h that matches the platform used. See the BSP module document in the r_bsp/doc folder for the contents of the r_bsp_interrupt_config.h file.



2.3 Adding the FIT Module

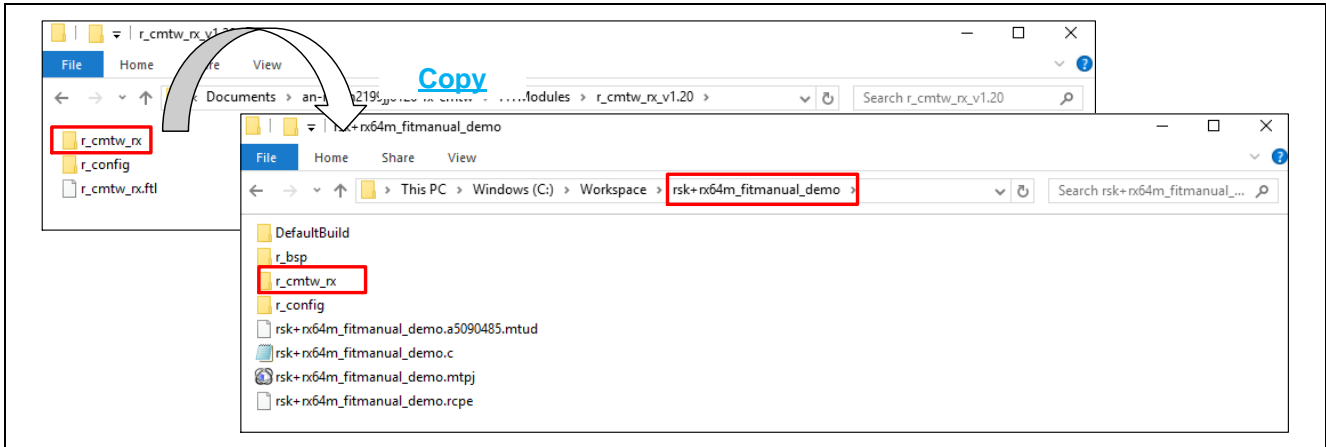
This chapter describes adding the FIT module using the CMTW FIT module as an example.

2.3.1 Download the FIT Module

Download the RX Family CMTW Module Using Firmware Integration Technology (R01AN2199EJxxxx) package from the Renesas Electronics Corporation web site and This package may be downloaded to any folder on the PC you are using.

2.3.2 Add the FIT Module

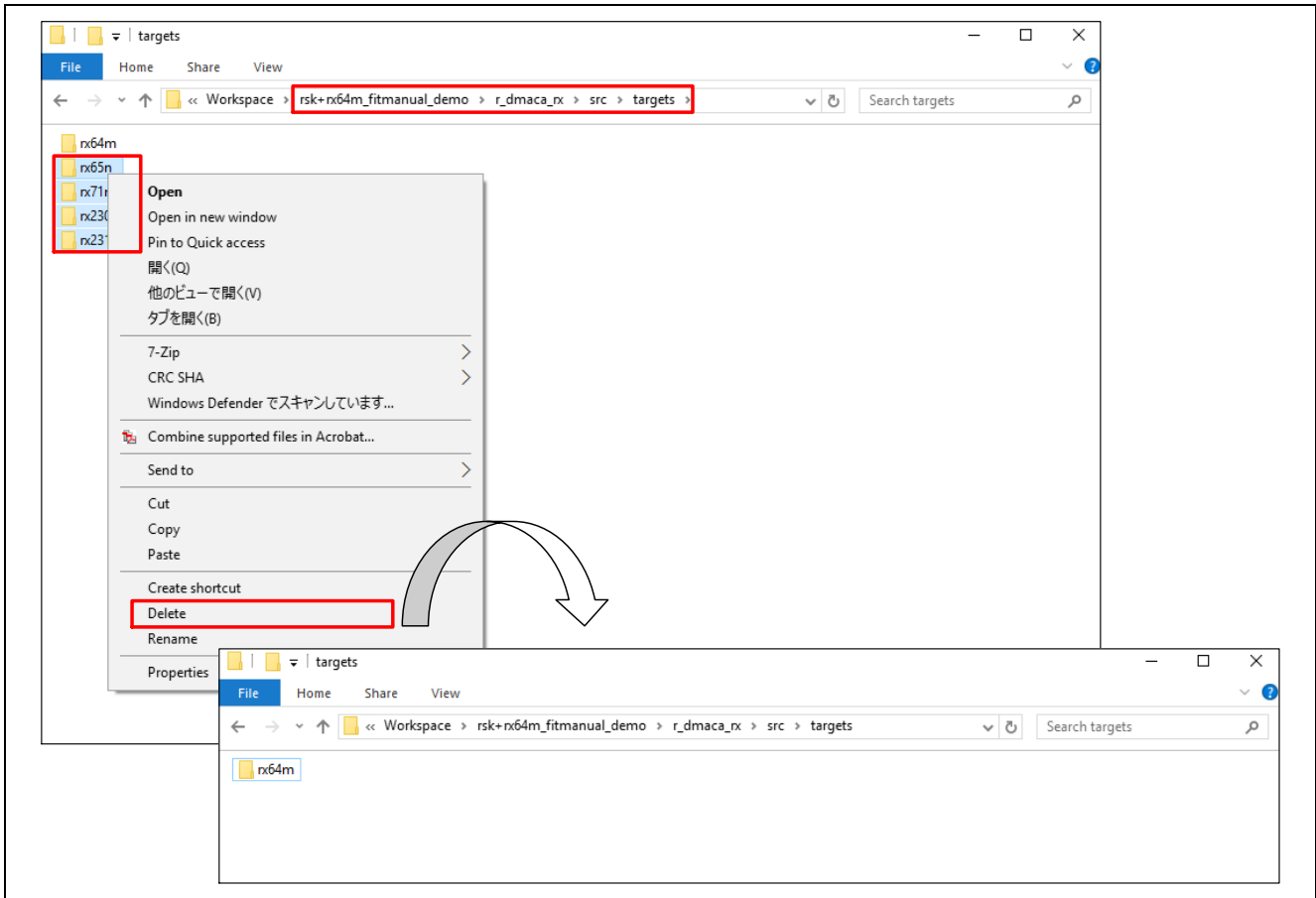
Using Windows Explorer, copy the FIT module `r_cmtw_rx` folder to the `<rsk+rx64m_fitmanual_demo>` folder.



2.3.3 Select the FIT Module

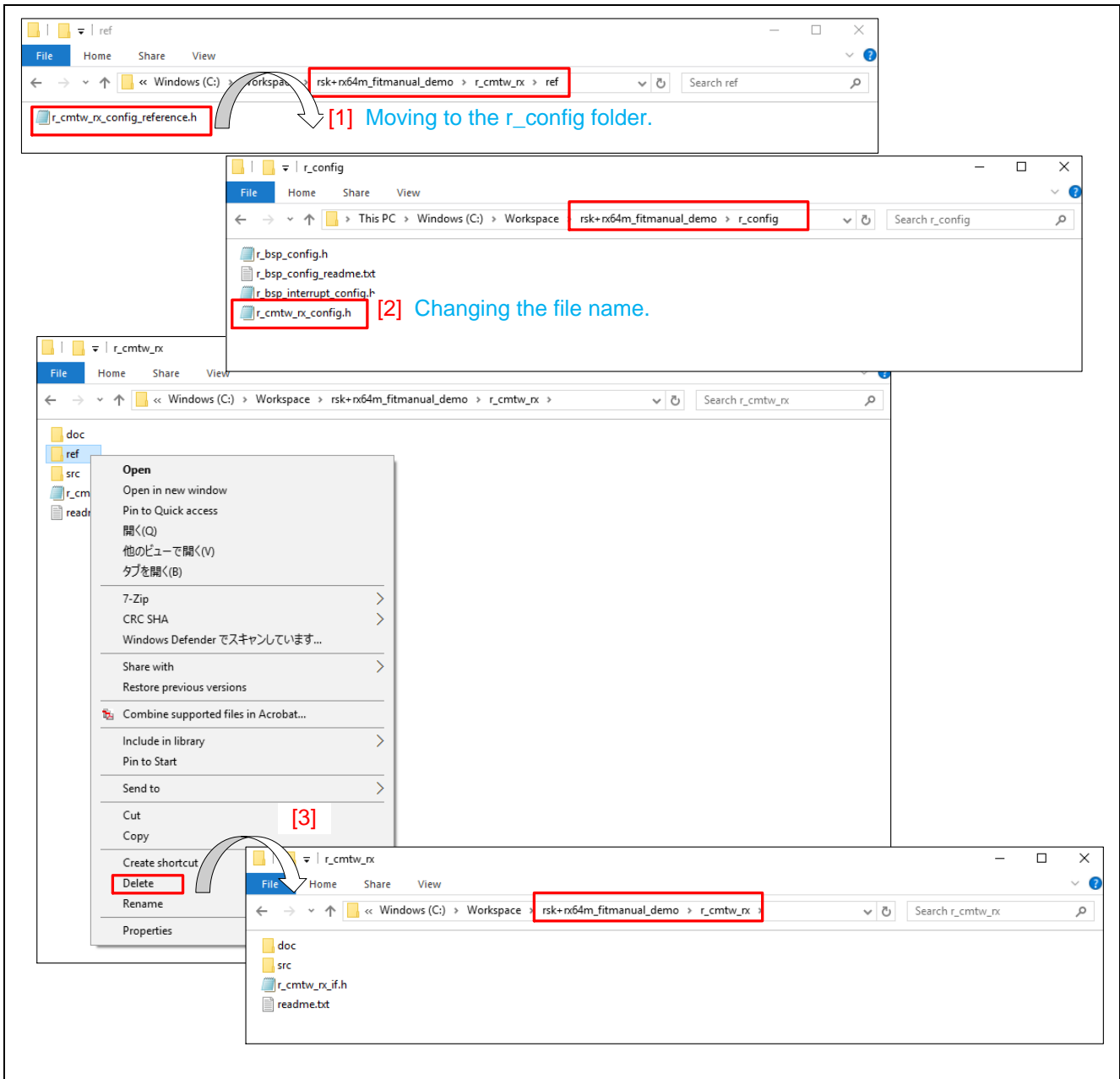
This chapter applies when there is a targets folder in the in the FIT module used. This chapter uses DMACA as an example.

Using Windows Explorer, switch to the `r_dmaca_rx\src\targets` folder and delete all folders except for the `rx64m` folder. If you are using a microcontroller other than the RX64M, instead of the `rx64m` folder, leave the microcontroller folder that corresponds to your microcontroller remaining.



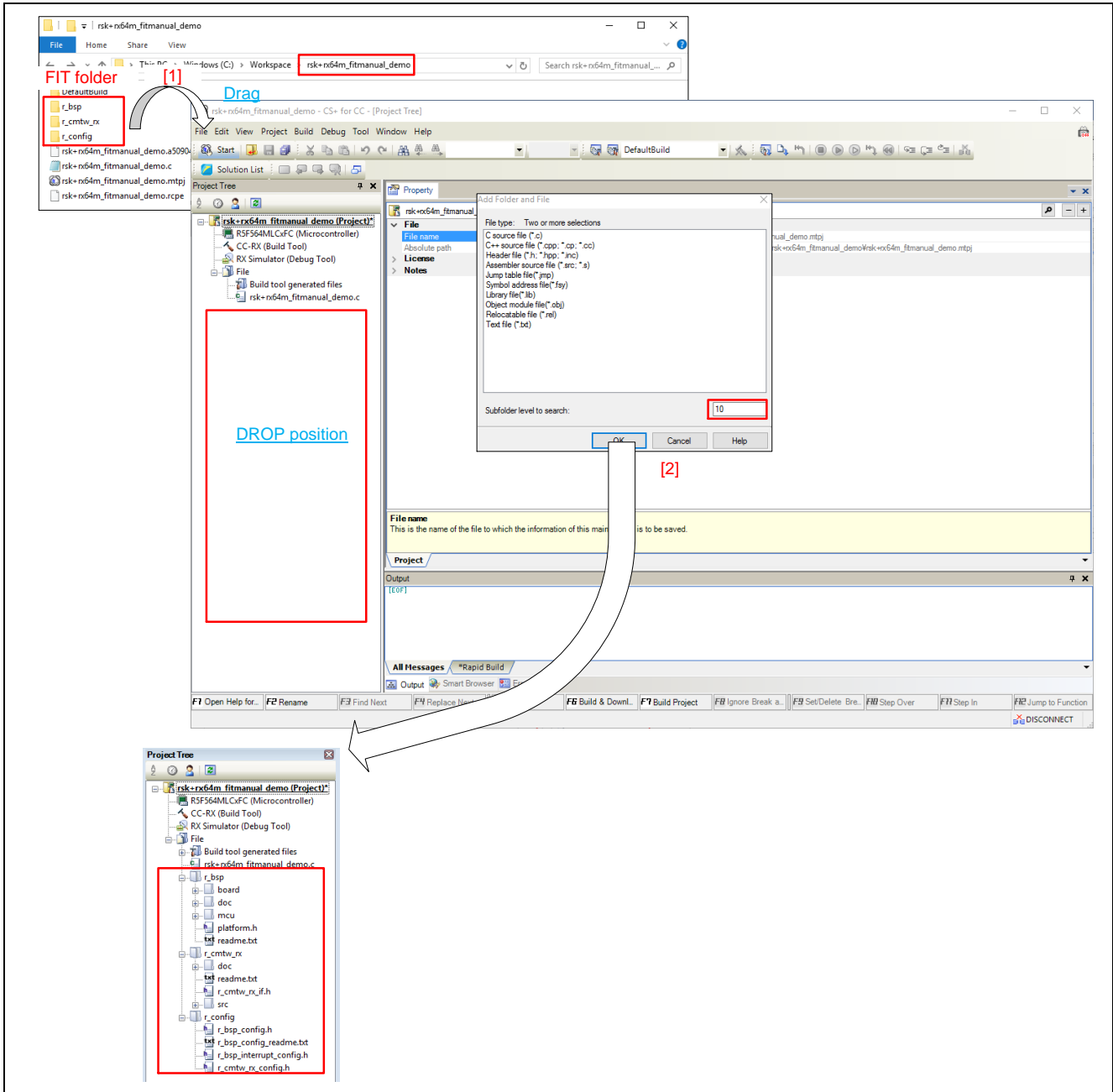
2.3.4 Create the FIT Module Setup File

1. Using Windows Explorer, move the `r_cmtw_rx_config_reference.h` file in the `r_cmtw_rx/ref` folder to the `r_config` folder. [1]
2. Using Windows Explorer, change the name of the moved `r_cmtw_rx_config_reference.h` file to be `r_cmtw_rx_config.h`. [2]
3. Using Windows Explorer, delete the `r_cmtw_rx/ref` folder. [3]
4. For details on the content of the `r_cmtw_rx_config.h` file, see the CMTW module document in the `r_cmtw_rx/doc` folder.



2.4 FIT Module Registration

1. Using Windows Explorer, drag and drop the FIT folder onto the project tree panel for the created CS+ project. [1]
2. When the folder is dropped, the “Add Folder and File” dialog will open. Enter the FIT folder level number in the Subfolder level to search(S) item and click OK. [2] (Values less than zero or greater than 10 may not be entered for the level number. Here we take the future maximum value of 10.)
3. The FIT module has been added to the CS+ project.



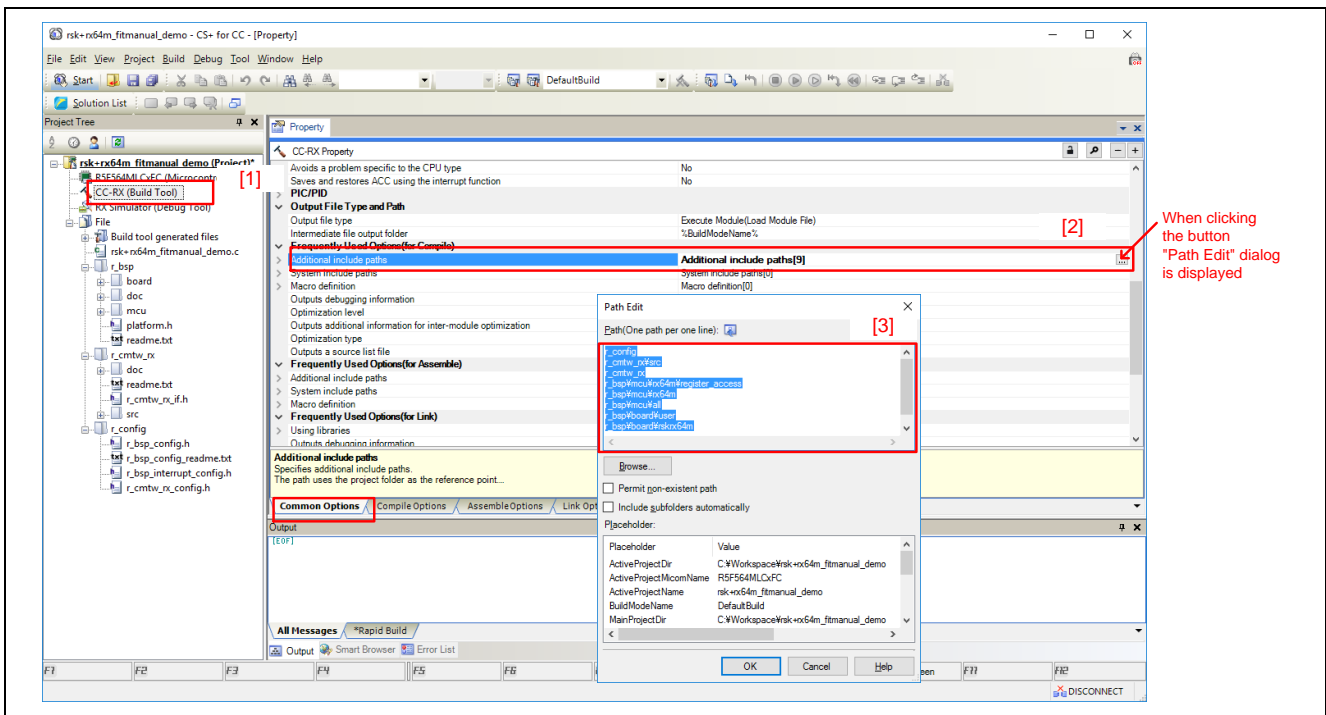
2.5 CC-RX Setup

This chapter describes the settings required to build the FIT module. The items not described here may be set to arbitrary values. Set those values as required.

2.5.1 Verify the Include Path

1. Select CC-RX (build tool) in the CS+ project tree panel. [1]
2. Select Add include path in the CS+ property panel Common options tab and verify the path in the Edit path dialog. [2]
3. The include path should be as follows. [3]

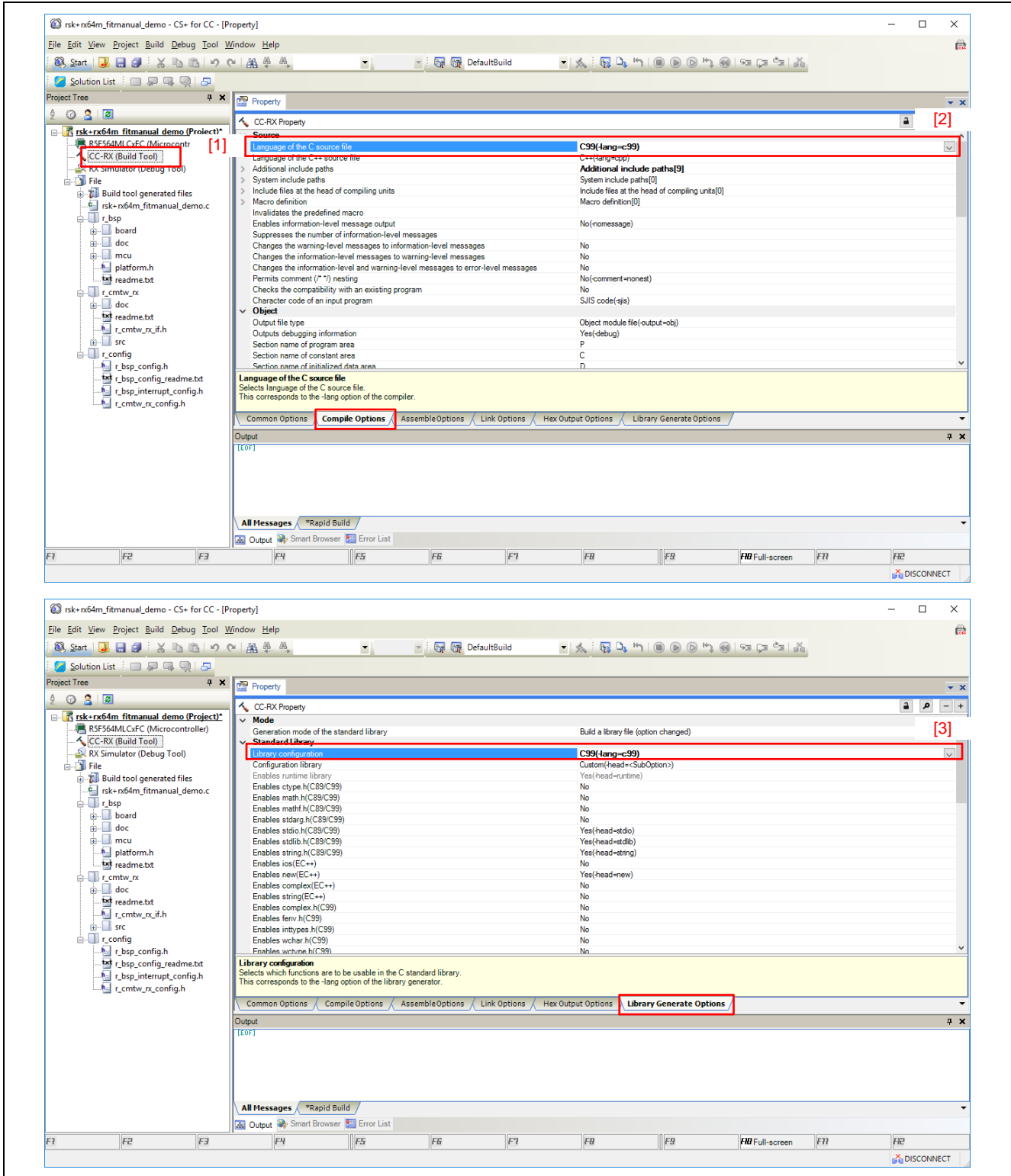
```
r_cmtw_rx\src
r_cmtw_rx
r_config
r_bsp\mcu\rx64m\register_access
r_bsp\mcu\rx64m
r_bsp\mcu\all
r_bsp\board\user
r_bsp\board\rskrx64m
r_bsp
```



2.5.2 Change the C Language Standard

1. Select CC-RX (build tool) in the CS+ project tree panel. [1]
2. Select the Compiler options tab in the CS+ property panel and set the C source file language source item to “C99(-lang=c99)”. [2]
3. Similarly, select Library generation options tab in the CS+ property panel and set the Library structure under Standard library to “C99(-lang=c99)”. [3]

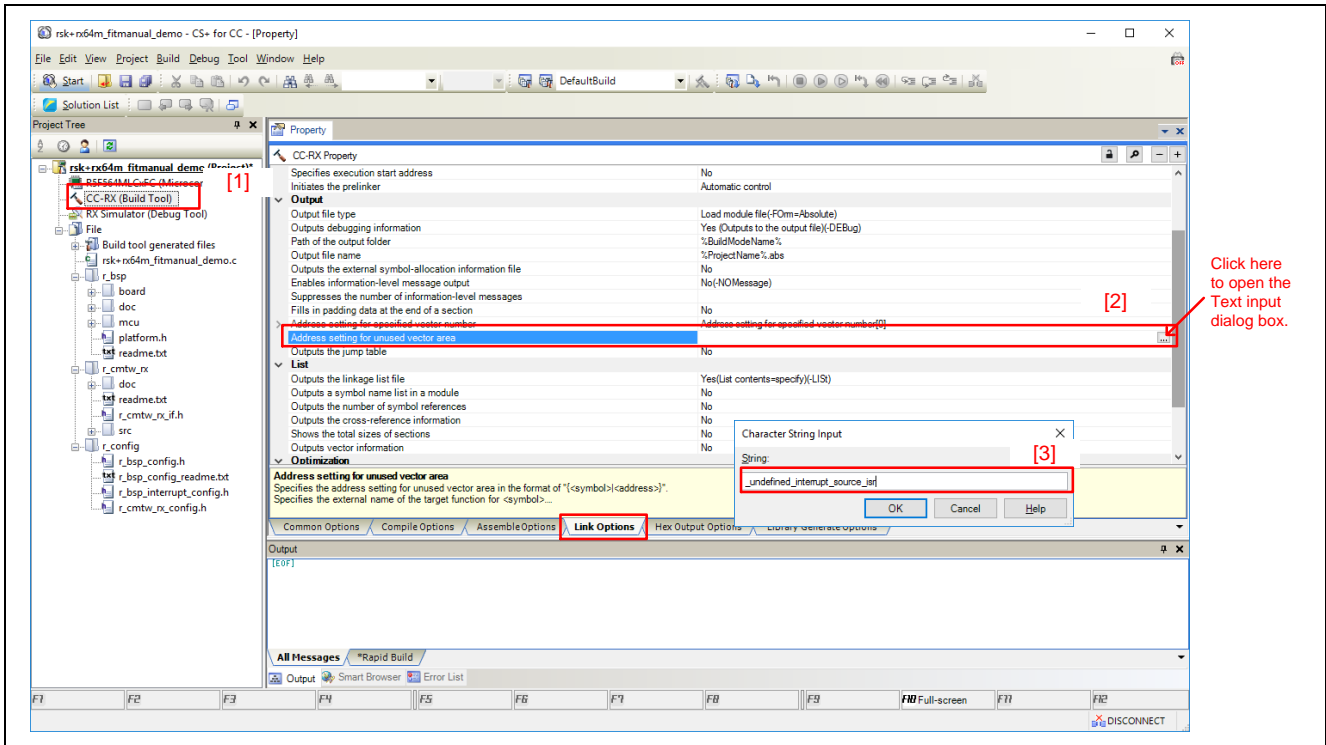
Note: To improve portability even more reliably, the FIT module source code is implemented using the ANSI C99 exact-width integer types.



2.5.3 Set the Variable Vector (Interrupt Vector) Empty Area

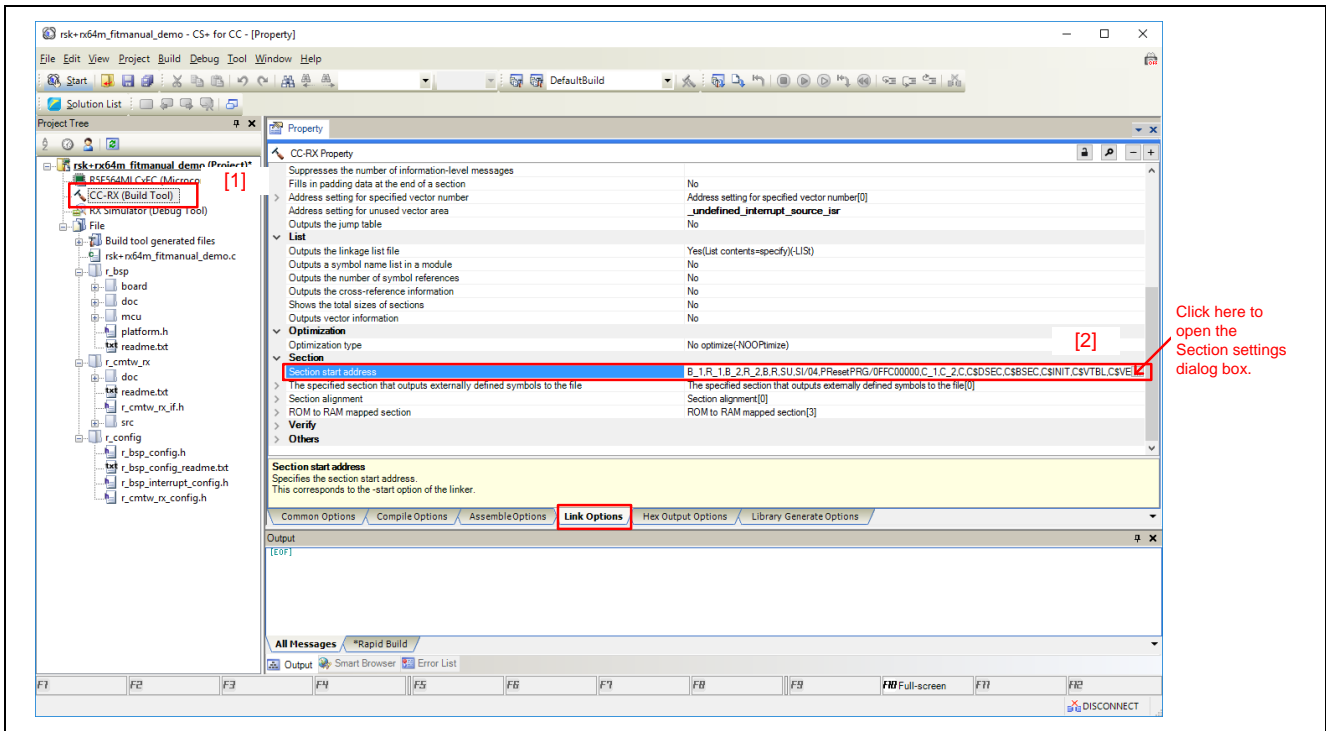
The unused interrupt vector area is filled with interrupt vectors to the undefined_interrupt_source_isr() function, and this is set up in the linker.

1. Select CC-RX (build tool) in the CS+ project tree panel. [1]
2. Select the Link options tab in the CS+ property panel and select the Variable vector empty area address under Output to open the Text input dialog box. [2]
3. Enter “_undefined_interrupt_source_isr” in the Character string(S) item in the Text input dialog box. [3]



2.5.4 Change the Section Address

1. Select CC-RX (build tool) in the CS+ project tree panel. [1]
2. Select the Link options tab in the CS+ property panel and select Section start address under section to open the Section settings dialog box. [2]

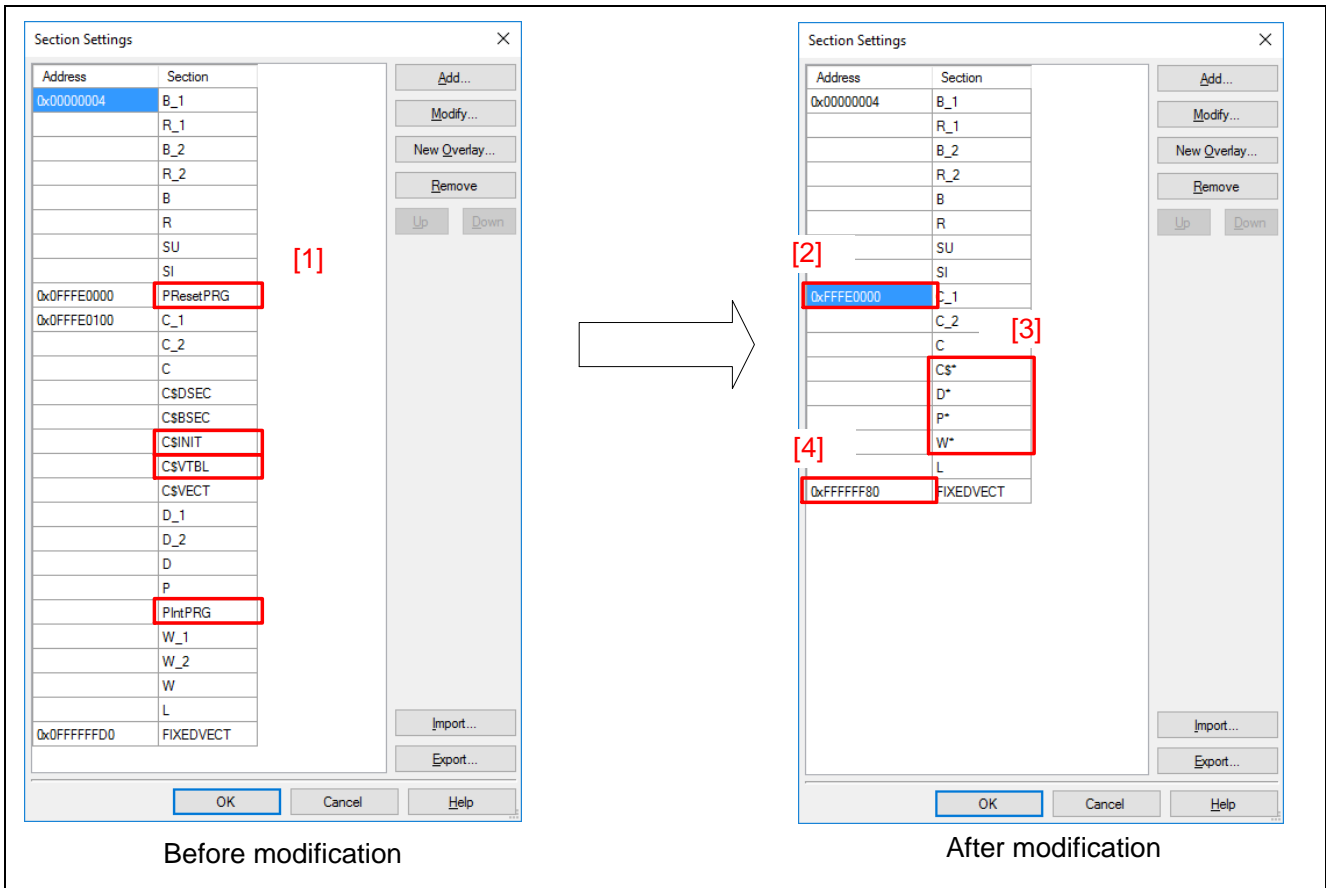


3. Refer to the following and modify the sections settings. (This applies when the CPU core is the RXv2)
 - Remove the unused sections (PResetPRG, C\$INIT, C\$VTBL, and PIntPRG). [1]
 - Set the C_1 section address to the address corresponding to the user area capacity of the microcontroller used. [2] (For example, set the ROM start address for the R5F564MLCxFC (4 MB) to 0xFFC0 0000.)
 - Change the section name as follows.[3]
 - C\$DSEC,C\$BSEC,C\$VECT → C\$*
 - D_1,D_2,D → D*
 - W_1,W_2,W → W*
 - P → P*

Before modification

After modification

4. Refer to the following and modify the sections settings. (This applies when the CPU core is the RXv1)
 - Remove the unused sections (PResetPRG, C\$INIT, C\$VTBL, and PIntPRG). [1]
 - Set the C_1 section address to the address corresponding to the user area capacity of the microcontroller used. [2] (For example, set the ROM start address for the R5F1115AxFM (128 KB) to 0xFFFFE 0000.)
 - Change the section name as follows.[3]
 - C\$DSEC,C\$BSEC,C\$VECT → C\$*
 - D_1,D_2,D → D*
 - W_1,W_2,W → W*
 - P → P*
 - Change the FIXEDVECT section address to 0xFFFF FF80. [4]



2.6 Build the Project

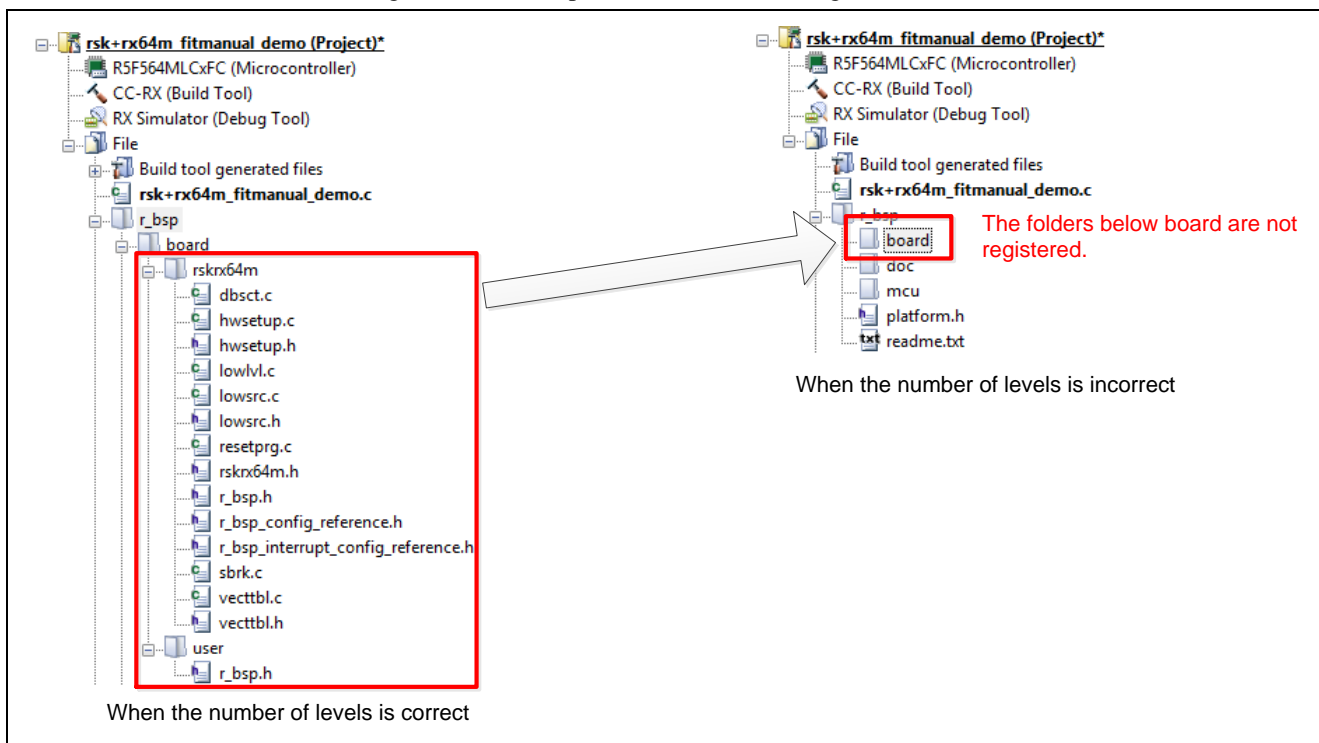
Create the user program and build the project.

3. Troubleshooting

1. Q: The following warnings are issued.
 W0561100: Cannot find "PResetPRG" specified in option "start"(resetprg.c)
 W0561100: Cannot find "C\$INIT" specified in option "start"
 W0561100: Cannot find "C\$VTBL" specified in option "start"
 W0561100: Cannot find "PIntPRG" specified in option "start"
 A: This is because the section names specified in the section settings could not be found. When a new CS+ project is created, PResetPRG, C\$INIT, C\$VTBL, and PIntPRG are set up in the defaults. Since these sections are not used when a FIT module is used, they must be deleted. See chapter 2.5.4, Change the Section Address, for details.

2. Q: We are using the RXv1. When we build the project, we get the following error. F0563100: Section address overflow out of range: FIXEDVECT.
 A: This error occurs because the address of the FIXEDVECT section exceeds the usable limit. When a new CS+ project is created, the FIXEDVECT address is set to H'FFFF FFD0. When a FIT module is used, this must be changed to H'FFFF FF80. See chapter 2.5.4, Change the Section Address, for details.

3. Q: We registered the FIT module, but none of the folders or files were added to CS+.
 A: When registering a FIT module, it is necessary to enter the number of folder levels. If the number of folder levels in the FIT module being registered exceeds the set number of folder levels, the folders deeper than the set number of levels are not registered. See chapter 2.4, FIT Module Registration, for details.



4. Q: When we build the project, we get the following error. F0520005: Could not open source file "r_xxxx_rx_config.h".
 A: When this error is displayed, it is possible that the source file r_xxxx_rx_config.h does not exist. There is a "ref" folder in the r_xxxx_rx FIT module folder in the project file you are using. Move the r_xxxx_rx_config_reference.h file in that folder to the r_config folder and change its name to r_xxxx_rx_config.h. See chapter 2.3.4, Create the FIT Module Setup File for details.

5. Q: When we build the project, we get the following error multiple times. E0562310: Undefined external symbol “symbol” referenced in “folder”.
- A: When this error is displayed, it is possible that folders for microcontrollers other than the one actually used are remaining in the `r_XXXX_RX/src/targets` folder in the project file you are using. Remove the folders for microcontrollers other than the one you are using. See chapter 2.3.3, Select the FIT Module, for details.
6. Q: When we build the project, we get the following error. F0520005: Could not open source file “`r_bsp_interrupt_config.h`”.
- A: When this error is displayed, it is possible that `r_bsp_interrupt_config.h` does not exist. Move the `r_bsp_interrupt_config_reference.h` file in the `r_bsp/board/<platform_used>` folder in the project file you are using to the `r_config` folder and change its name to `r_bsp_interrupt_config.h`. See chapter 2.2.7, Create the BSP Settings File (`r_bsp_interrupt_config.h`), for details.
7. Q: When we build the project, we get the following error. F0520005: Could not open source file “`r_bsp_config.h`”.
- A: When this error is displayed, it is possible that `r_bsp_config.h` does not exist. Move the `r_bsp_config_reference.h` file in the `r_bsp/board/<platform_used>` folder in the project file you are using to the `r_config` folder and change its name to `r_bsp_config.h`. See chapter 2.2.6, Create the BSP Setup File (`r_bsp_config.h`), for details.
8. Q: When we build the project, we get the following errors.
- F0520035: #error directive: “Error - No platform defined in platform.h!”
- F0520005: Could not open source file “`./board/platformname /r_bsp.h`”
- A: When these errors are displayed, it is possible that the platform settings in the `platform.h` file have not been made correctly. Remove the commenting from `#include` statement to match the platform you are using. See chapter 2.2.5, Set Up the Header File, for details.
9. Q: When we build the project, we get the following warning. W0561010: Duplicate file specified in option “input”.
- A: When this error is displayed, it is possible that folders for microcontrollers other than the one actually used are remaining in the `r_bsp/mcu` folder in the project file you are using. Remove the folders for unneeded microcontrollers. See chapter 2.2.4, Select the Microcontroller, for details.
10. Q: When we build the project, enormous numbers of errors are generated.
- A: When large numbers of errors are generated, it is possible that folders for platforms other than the platform you are using remain in `r_bsp/board` folder in the project file used. Remove the unnecessary platform folders. See chapter 2.2.3, Select the Platform, for details.
11. Q: When we build the project, we get the following error. E0562142: Interrupt number “xx” of “vector” has multiple definitions.
- A: This is because multiple vector number definitions have been input. When CS+ is used to create a new project, as the default, the files `intprog.c` and `vect.h` are generated. When BSP and FIT modules are used, it is necessary to either remove the `intprog.c` and `vect.h` files or to remove the multiple vector number definitions within these files. See chapter 2.1.2, Delete Unused Files, for details.

4. Notes

- The FIT module applies ANSI C99 “Exact-width integer types” for a more reliable source code, and increased portability. Source code created by the user should comply with ANSI C99.
- The necessary memory size differs for each FIT module. The memory size also differs according to the FIT module settings.
- The BSP is required for using the FIT module.

5. FIT Module

FIT modules can be downloaded from the Renesas Electronics website.

6. Reference Documents

Technical Update/Technical News

The latest information can be downloaded from the Renesas Electronics website.

User’s Manual: Development Tools

CC-RX Compiler User’s Manual (R20UT3248)

The latest version can be downloaded from the Renesas Electronics website.

CS+ V5.00.00 Integrated Development Environment User’s Manual: Project Operation (R20UT3928)

The latest version can be downloaded from the Renesas Electronics website.

Website and Support

Renesas Electronics Website

<http://www.renesas.com/>

Inquiries

<http://www.renesas.com/contact/>

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

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jan. 15, 2014	—	First edition issued
1.01	Dec. 1, 2014	1	Added the description regarding CS+.
1.02	Dec. 10, 2014	—	Changed the document title according to the renaming of the product. (Previous title: RX Family Adding Firmware Integration Technology Modules to CubeSuite+ Projects)
		3	Figure 1.1: Revised.
1.10	Mar. 23, 2017	1	Deleted text describing the change to the product name of the Renesas integrated development environment.
		3	Changed CubeSuite+ to CS+ in the terminology definitions. Deleted CGC from, and added CMTW and DMACA to the terminology definitions
		5	Changed Figure 1.2
		6	Changed Figure 1.3 Corrected descriptions in Figure 1.3
		7	Changed chapter titles and corrected text Changed Figure 2.1
		8 to 10	Corrected Figure 2.1
		11 to 14	Corrected chapter 2.2 Added chapter 2.2.1, Download BSP Added chapter 2.2.7, Create the BSP Settings File (r_bsp_interrupt_config.h)
		15 to 17	Corrected chapter 2.3 Added chapter 2.3.1, Download the FIT Module
		—	Deleted (1), Folder structure
		19 to 24	Corrected chapter 2.5 Deleted (2), Change build method
		25, 26	Added chapter 3., Troubleshooting
		27	Modified the reference documents.

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. 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products

Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

- The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the 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 disputes 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, drawing, chart, program, algorithm, 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 not alter, modify, copy, or otherwise misappropriate 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, copy or otherwise misappropriation of Renesas Electronics products.
5. 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; and 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.
Renesas Electronics products are neither intended nor 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 damages (space and 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 third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the 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 radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. 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. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of 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 by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use 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.
9. 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 not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.3.0-1 November 2016)



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "<http://www.renesas.com/>" for the latest and detailed information.

Renesas Electronics America Inc.

2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited

9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH

Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.

Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.

Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited

Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.

13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.

80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.

Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jin Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.

No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.

12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141