

RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

Introduction

This application note describes how to port a software generated by RL78 Smart Configurator to another RL78. As an example, this application note explains the procedure to port the RL78/G23 sample code to the RL78/G15 sample code.

Target Device

RL78/G15

RL78/G22

When applying the sample program covered in this application note to another microcomputer, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

Contents

1. Overview	4
1.1 Target for Porting.....	4
1.2 Sample Code Structure	5
1.3 Porting Method	10
2. Porting Procedure.....	12
2.1 Advance preparation	12
2.2 Copy the project	13
2.3 Open the project.....	13
2.3.1 CS+ for CC	13
2.3.2 e ² studio.....	14
2.3.3 IAR EW for Renesas RL78.....	16
2.4 Change the compiler version.....	17
2.4.1 CS+ for CC	17
2.4.2 e ² studio.....	18
2.4.3 IAR EW for Renesas RL78.....	18
2.5 Check code changes	19
2.6 Change resources	20
2.6.1 SC startup procedure: CS+ for CC.....	21
2.6.2 SC startup procedure: e ² studio	22
2.6.3 SC startup procedure: IAR EW for Renesas RL78	23
2.6.4 Change resources	24
2.7 Generate report of SC configuration	26
2.8 Change the device.....	28
2.8.1 CS+ for CC	28
2.8.2 e ² studio.....	31
2.8.3 IAR EW for Renesas RL78.....	34
2.9 Check the SC migration report.....	37
2.10 Change settings and resources.....	40
2.10.1 Clock Settings.....	40
2.10.2 System Settings.....	42
2.10.3 Interrupt	43
2.10.4 Software Component.....	44
2.10.5 Pins.....	49
2.11 Save the SC settings	62
2.12 Code generation	62
2.13 Edit the code.....	63
2.14 Build project of destination RL78.....	68
3. Example of Porting	69

3.1	Advance preparation	69
3.2	Copy the project	72
3.3	Open the project	72
3.4	Change the compiler version.....	72
3.5	Check code changes	72
3.6	Change resources	72
3.7	Generate report of SC configuration	73
3.8	Change the device.....	73
3.9	Check the SC migration report	73
3.10	Change settings and resources.....	73
3.10.1	Clock settings	73
3.10.2	System settings	75
3.10.3	Interrupt	75
3.10.4	Software Component.....	77
3.10.5	Pins.....	78
3.11	Save the SC settings	80
3.12	Code Generation	80
3.13	Edit the code.....	80
3.14	Build project of destination RL78.....	81
4.	Sample code.....	83
5.	Reference Document.....	83
	Revision History	84

1. Overview

1.1 Target for Porting

This application note describes the procedure to port the RL78's program (device driver) generated by Smart Configurator (SC) to the program for another RL78 product.

However, if the RL78 product as the destination device (called destination RL78) does not have the same peripheral functions as the original RL78 product (called source device), the device driver cannot be ported. Also, even if the same peripheral functions are equipped, if the operating mode of the peripheral functions used in the sample code does not exist in the destination RL78, porting is not possible.

When the device driver cannot be ported, create a new project for the destination RL78 and develop the program newly.

The RL78 products as the destination device is shown in Table 1-1.

Table 1-1 Destination RL78

RL78/G1x	RL78/G15
RL78/G2x	RL78/G22

Table 1-2 Criteria for Determining Whether Porting is Possible

Destination RL78	Porting is Possible / Impossible
The destination RL78 does not have the same peripheral functions as the source RL78.	Impossible (develop a new program for destination RL78)
The destination RL78 has the same peripheral functions and the same channels as the source RL78.	Possible
The destination RL78 has the same peripheral functions as the source RL78 but there is not the same channels as the source RL78.	Possible if the channel can be changed
The destination RL78 has the same peripheral functions as the source RL78 but there is not the same operation mode as the source RL78.	Impossible (develop a new program for destination RL78)
The operating voltage of the destination RL78 differs from that of the source RL78.	Possible if the operating voltage can be set within the range of operation ^{Note}
The operating frequency of the destination RL78 product differs from that of the source RL78.	Possible if the operating frequency can be set within the range of operation ^{Note}
The destination RL78 does not have the same clock source used in the sample code as the source RL78.	Impossible (develop a new program for destination RL78)

Note. When designing and implementing an actual circuit, make sure to satisfy the hardware's electrical specifications of the destination RL78.

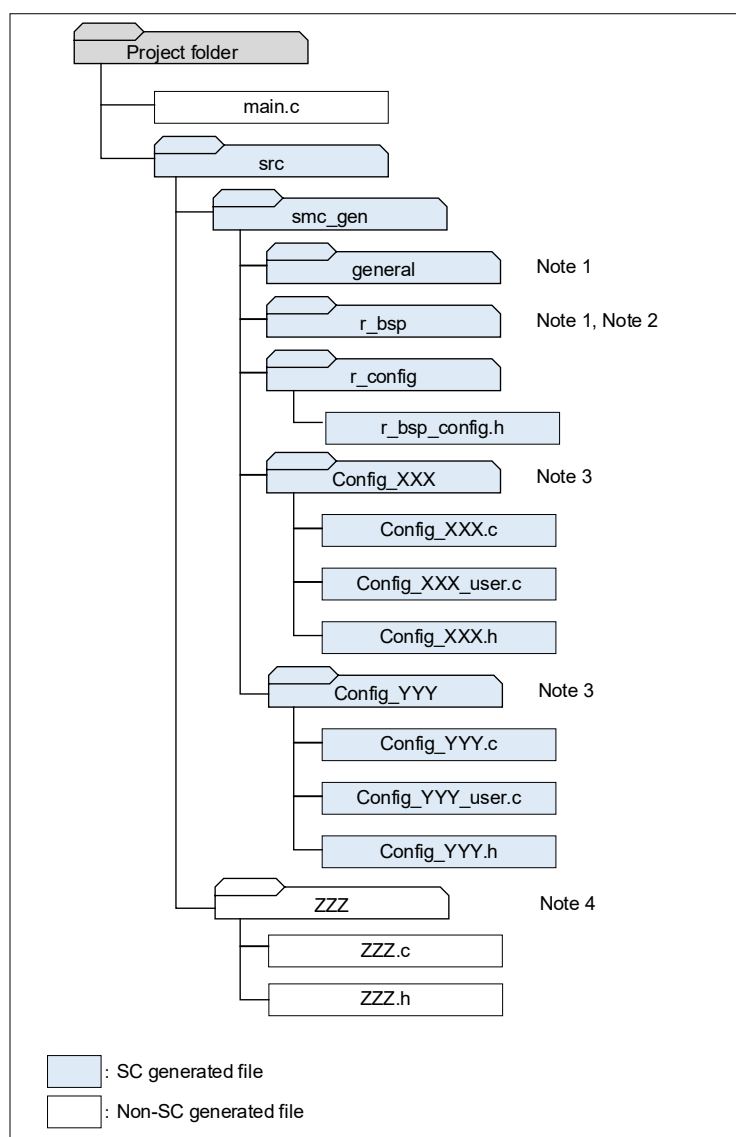
1.2 Sample Code Structure

The sample code of an RL78 product consists of files generated by Smart Configurator (SC) (called SC generated files) and other files (called non-SC generated files). SC generated files contain the Board Support Package (BSP), and codes that configure the peripheral functions for a target RL78 product of the project.

Figure 1-1 shows an example project containing sample code. Table 1-3 shows an overview of SC generated files and functions.

From Figure 1-3 to Figure 1-8 show an overview of processing from occurrence of CPU reset to calling of the main function.

Figure 1-1 Example of Sample Code Project



Note 1. The files generated under each folders are omitted.

Note 2. The files are provided by Board Support Package (BSP).

Note 3. XXX and YYY are replaced by abbreviations of peripheral feature names. The number of folders that exist differs depending on the sample code.

Note 4. ZZZ is any character. Folder/files indicated by ZZZ may not exist or may exist multiple times depending on the sample code.

RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

Table 1-3 Outline of SC generated files and folders

Folder name	File name	Description
general	(omitted)	Macro and Typedef definitions used in common by SC generated files
r_bsp	(omitted)	Board Support Package (BSP) ^{Note 1}
r_config	r_bsp_config.h	Platform setting ^{Note 1} (Startup file provided by BSP, files referred by BSP API function)
Config_XXX ^{Note 2, 3}	Config_XXX.c Config_XXX_user.c Config_XXX.h	Peripheral function initial setting processing, Peripheral function start processing, Peripheral function stop processing Interrupt function of peripheral function Macro and Typedef definitions for peripheral function
ZZZ ^{Note 4}	(omitted)	Non-SC generated file (Processing functions required for the sample code)

Note 1. For details of Board Support Package (BSP) and platform setting, refer to RL78 Family Board Support Package Module Using Software Integration System Application Note (R01AN5522).

Note 2. For details of files and functions generated by Smart Configurator (SC), refer to Smart Configurator User's Manual: RL78 API Reference (R20UT4852).

Note 3. XXX is replaced by abbreviations of peripheral feature names.

Note 4. ZZZ is any character.

Remark. The code in the file generated by SC is generated according to the settings in SC, therefore, the generated code is basically used without changing it. User code can be added in the comment for user code description in each file. The code written in this comment will be protected from being overwritten even if SC generates code again.

Figure 1-2 Example of comment for user code

```

/*=====
Includes
=====
#include "r_cg_macrodriver.h"
#include "r_cg_userdef.h"
#include "Config_TAU0_0.h"
/* Start user code for include. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/*=====
Pragma directive
=====
#pragma interrupt r_Config_TAU0_0_channel0_interrupt (vect=INTTMO0)
/* Start user code for pragma. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/*=====
Global variables and functions
=====
/* Start user code for global. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/*=====
Function Name: R_Config_TAU0_0_Create_UserInit
Description : This function adds user code after initializing the TAU0 channel0.
Arguments : None
Return Value : None
=====
void R_Config_TAU0_0_Create_UserInit(void)
/* Start user code for user init. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
*/

```

Figure 1-3 Processing image from CPU reset generation to main function: Startup function

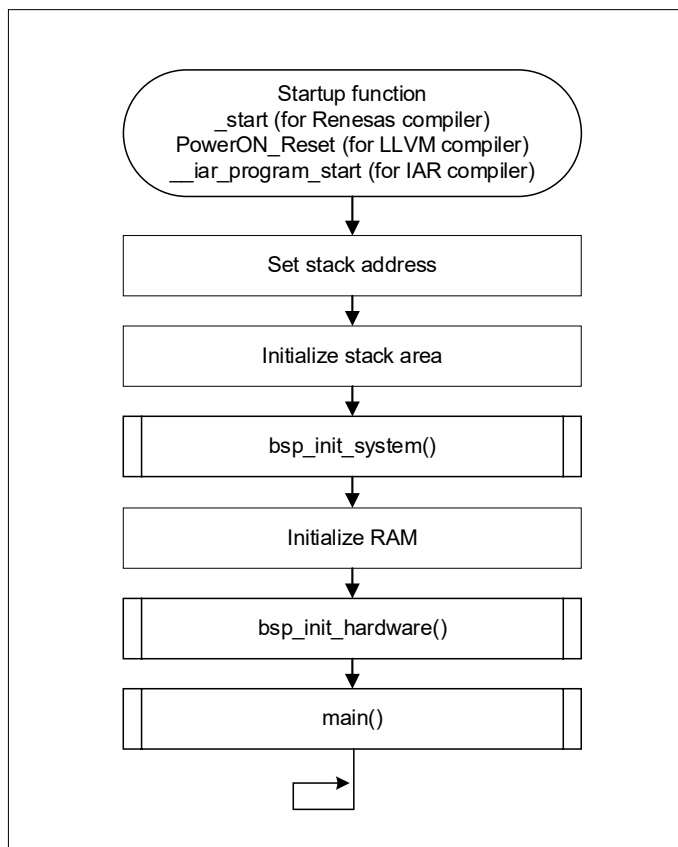
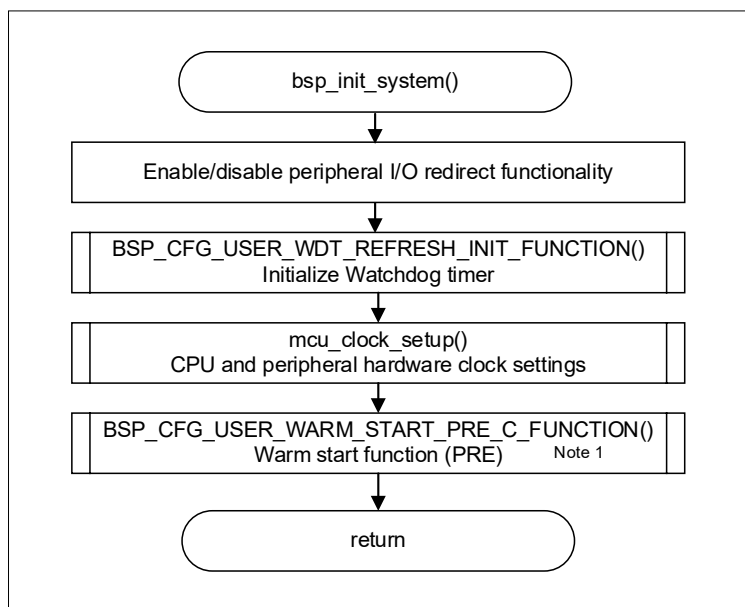
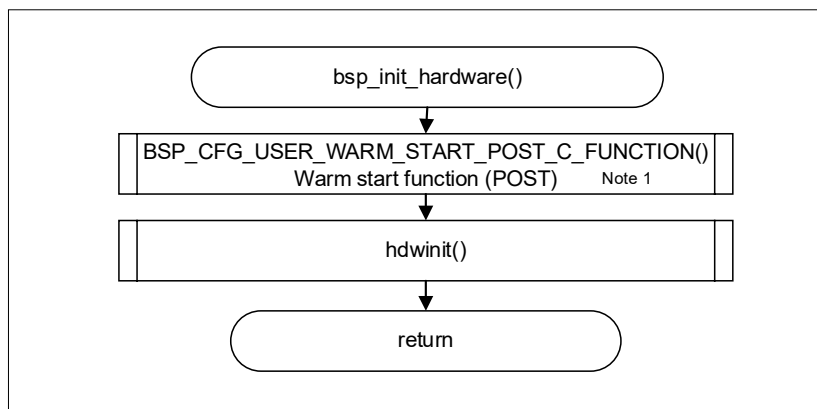


Figure 1-4 Processing image from CPU reset generation to main function: bsp_init_system()



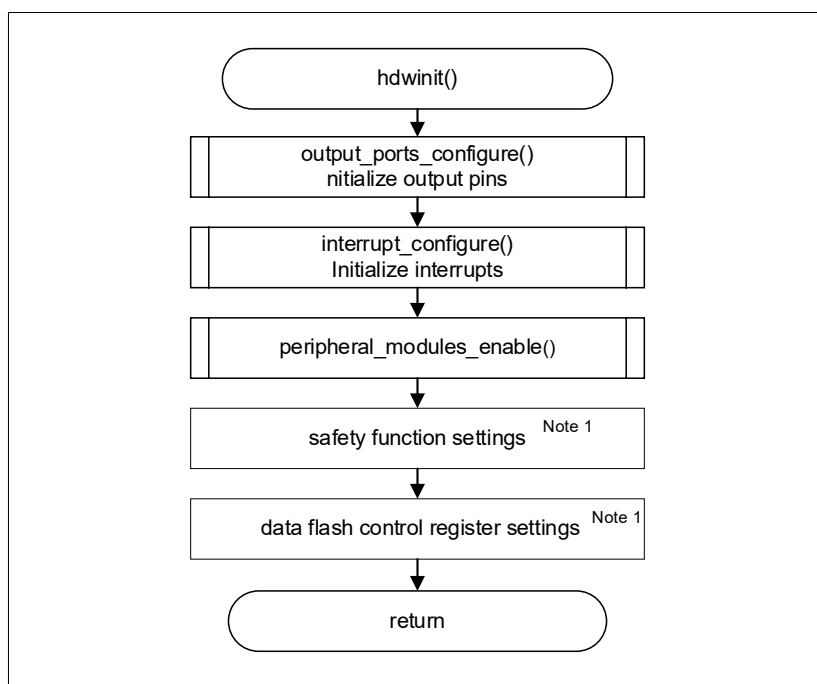
Note 1. The processing depends on the setting of configuration options in `r_bsp_config.h`. When using Smart Configurator (SC), it is possible to set configuration options on the software component settings screen. Configuration options that are set in SC are automatically reflected in `r_bsp_config.h` during code generation.

Figure 1-5 Processing image from CPU reset generation to main function: bsp_init_hardware()



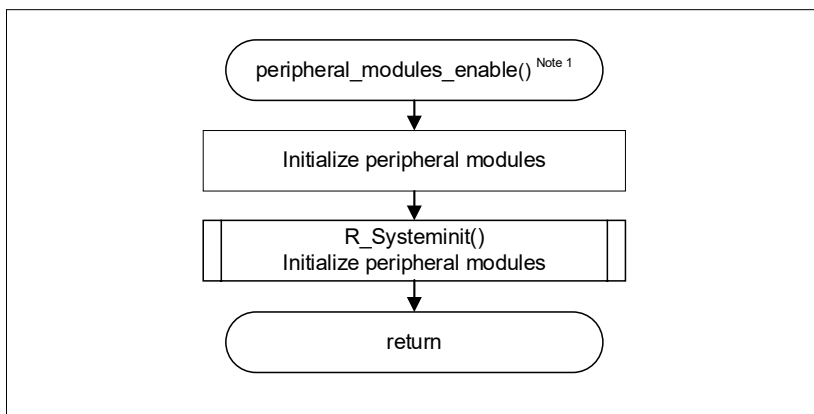
Note 1. The processing depends on the setting of configuration options in `r_bsp_config.h`. When using Smart Configurator (SC), it is possible to set configuration options on the software component settings screen. Configuration options that are set in SC are automatically reflected in `r_bsp_config.h` during code generation.

Figure 1-6 Processing image from CPU reset generation to main function: hdwinit()



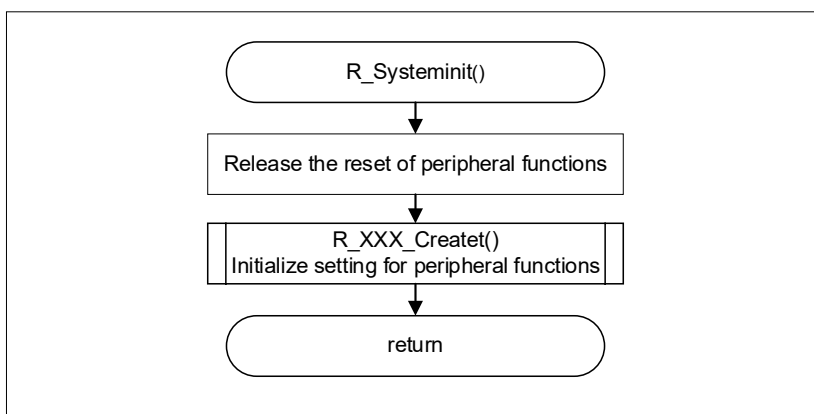
Note 1. The processing depends on the setting of configuration options in `r_bsp_config.h`. When using Smart Configurator (SC), it is possible to set configuration options on the software component settings screen. Configuration options that are set in SC are automatically reflected in `r_bsp_config.h` during code generation.

Figure 1-7 Processing image from CPU reset generation to main function: peripheral_modules_enable()



Note 1. The processing depends on the setting of configuration options in `r_bsp_config.h`. When using Smart Configurator (SC), it is possible to set configuration options on the software component settings screen. Configuration options that are set in SC are automatically reflected in `r_bsp_config.h` during code generation.

Figure 1-8 Processing image from CPU reset generation to main function: R_Systemiinit()



Remark. For details, refer to RL78 Family Board Support Package Module Using Software Integration System Application Note (R01AN5522).

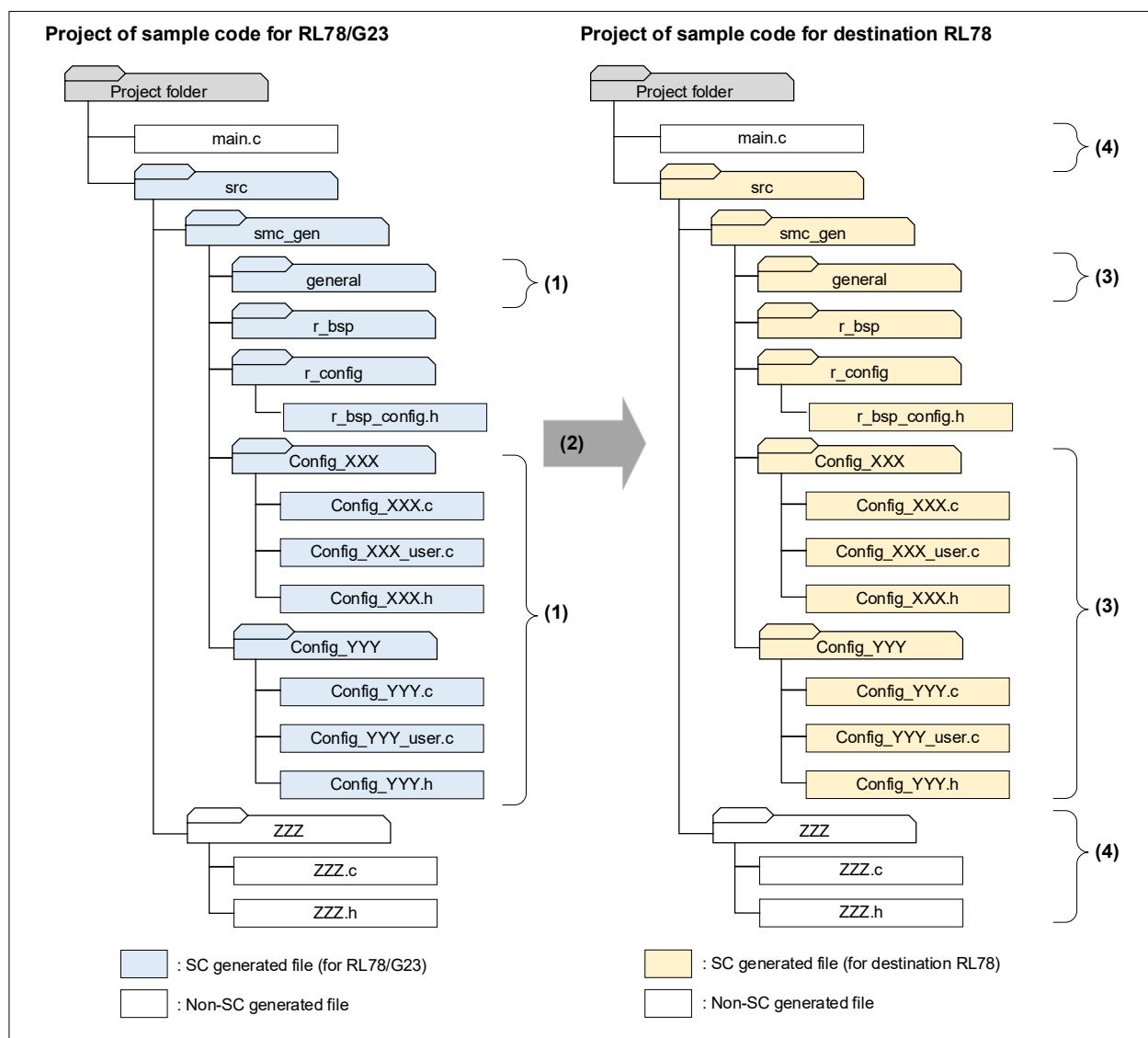
1.3 Porting Method

When porting the sample code, change the device of the project to the destination RL78 and change resources to fit the destination RL78.

In some sample codes, the code generated by Smart Configurator (SC) partially might be commented out or modified in other ways. Check if the SC generated code for the source RL78 is modified or not because the same modifications have to be reflected in the destination RL78's code before building the code. Note that the destination RL78 might not be able to use the same resources as the source RL78. In that case, the resource needs to be changed to another resource.

Figure 1-9 shows the overview of porting sample code and Figure 1-10 shows the flow of the porting procedure.

Figure 1-9 Overview of porting



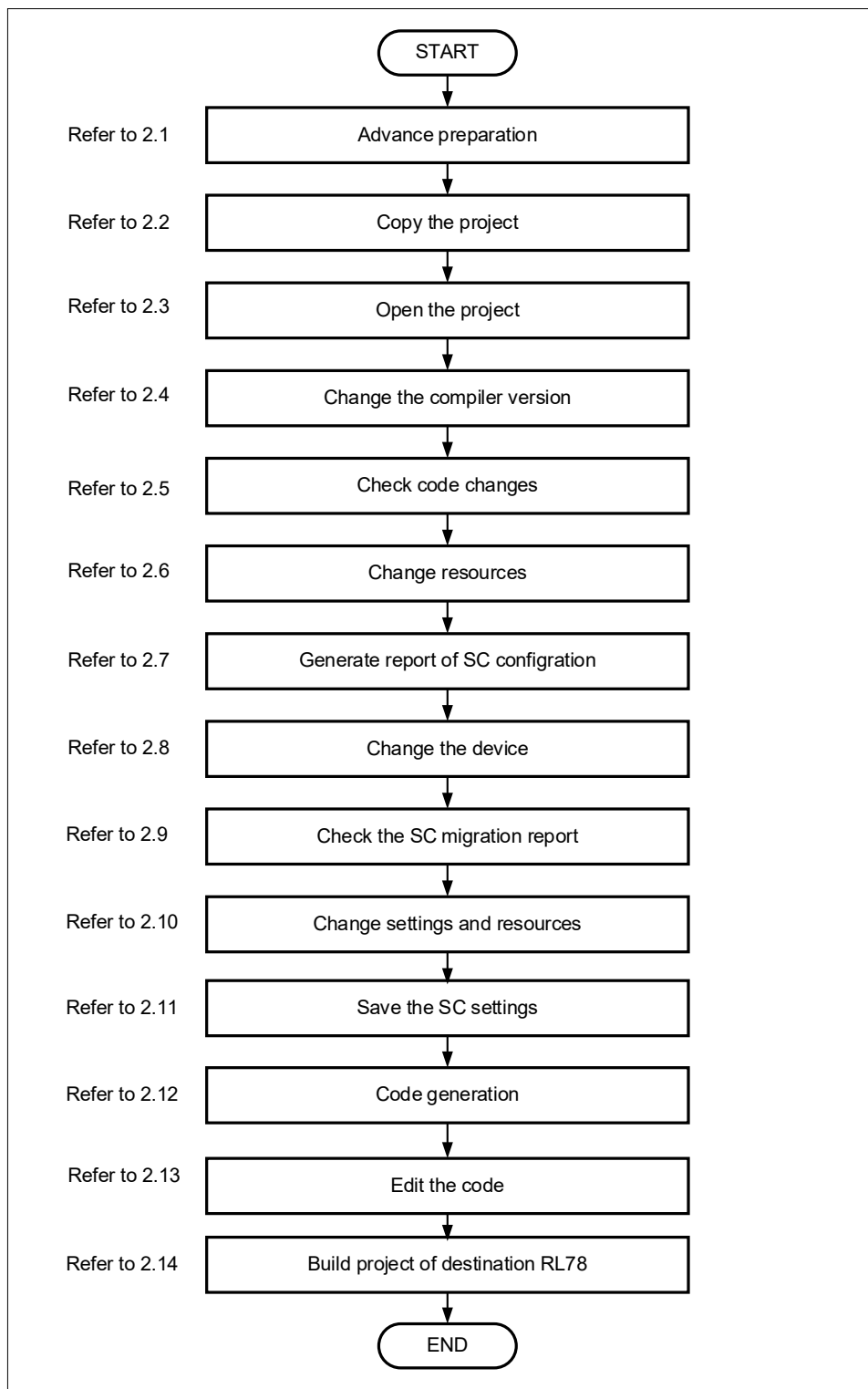
(1) Check whether SC generated code is modified (added or deleted any code) or not.

(2) Change the target device of the source project to the destination RL78 and generate code for the destination RL78.

(3) Make the same modification in the destination RL78's code indicated by (3) if the modification is found in step (1).

(4) If the resource is changed, modify the code accordingly.

Figure 1-10 Flow of Porting Procedure



2. Porting Procedure

2.1 Advance preparation

Check the port pins and peripheral functions used in the sample code for the source RL78 by referring to the document of the sample code. Then, consider how to port by referring the user's manual of the destination RL78. If it corresponds to "impossible" in Table 1-2, it is necessary to develop a new program for the destination RL78.

Table 2-1, Table 2-2, Table 2-3 and Table 2-4 show check items for operation voltage, operation frequency, peripheral function and used pin.

Table 2-1 Check items for operation voltage

Destination RL78	Porting method
It is enabled to operate at the same voltage as the source RL78.	Use the same operation voltage
It is disabled to operate at the same voltage as the source RL78.	Change to another voltage ^{Note}

Note. When designing and implementing an actual circuit, make sure to satisfy the hardware's electrical specifications of the destination RL78.

Table 2-2 Check items for operation frequency

Destination RL78	Porting method
It is enabled to operate at the same frequency as the source RL78.	Use the same frequency
It is disabled to operate at the same frequency as the source RL78.	Change to another frequency ^{Note}

Note. When designing and implementing an actual circuit, make sure to satisfy the hardware's electrical specifications of the destination RL78.

Table 2-3 Check items for peripheral function

Destination RL78	Porting method
There is the same channel as the source RL78.	Use the same channel
There is not the same channel as the source RL78.	Change to another channel

Table 2-4 Check items for used pin

Destination RL78	Porting method
There is the same port pin as the source RL78, and that pin have the same alternate function.	Use the same pin
There is the same port pin as the source RL78, but that pin does not have the same alternate function.	When using the port function without using the alternative function: Use the same port pin When using the alternative function: Change to the pin that have the same alternative function
There is not the same port pin as the source RL78.	When using the port function without using the alternative function: Change to another port pin When using the alternative function: Change to the pin that have the same alternative function

2.2 Copy the project

Copy the source sample code project to any folder as the destination project.

2.3 Open the project

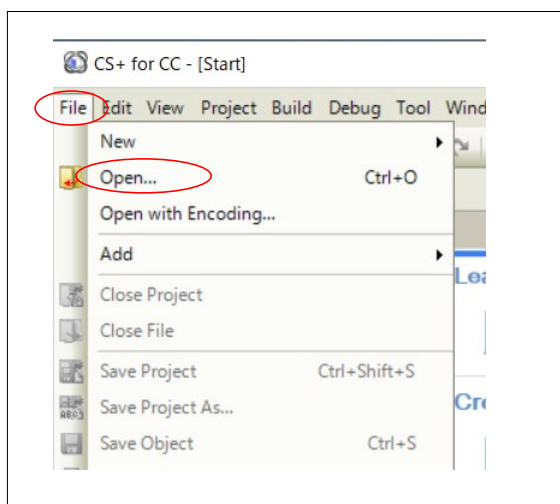
This section describes how to open a project for each development tool (IDE).

2.3.1 CS+ for CC

Start CS+ for CC and open the destination project.

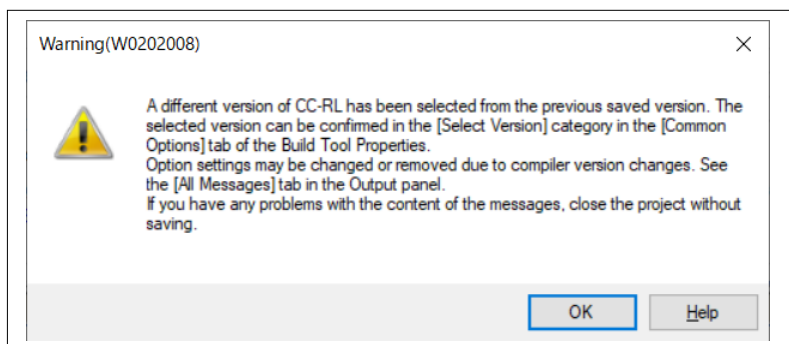
After starting CS+ for CC, click the [File] – [Open] menu and load the destination project: .mtpj file.

Figure 2-1 Open the destination project (.mtpj)



Note. If the version of the tool used when creating the sample code is different from the current version of the tool, a warning message will be displayed to inform that the versions are different. Check the contents and click the [OK] button.

Figure 2-2 [Warning] dialog

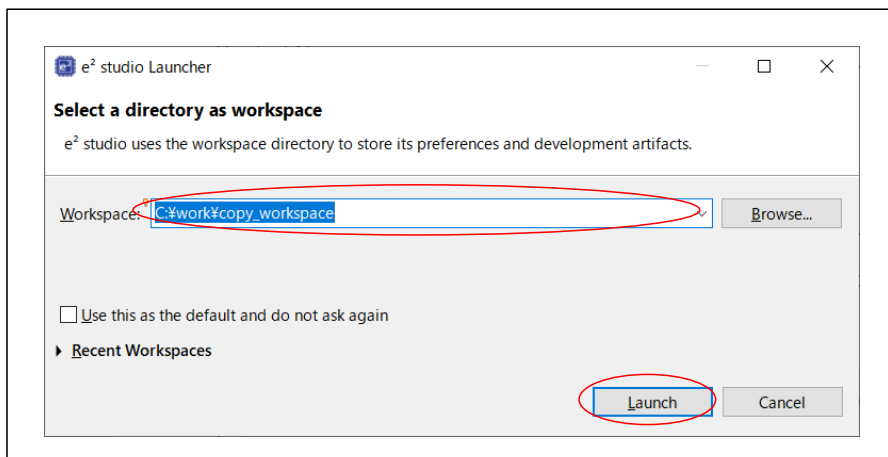


2.3.2 e² studio

Start e² studio and open the destination project.

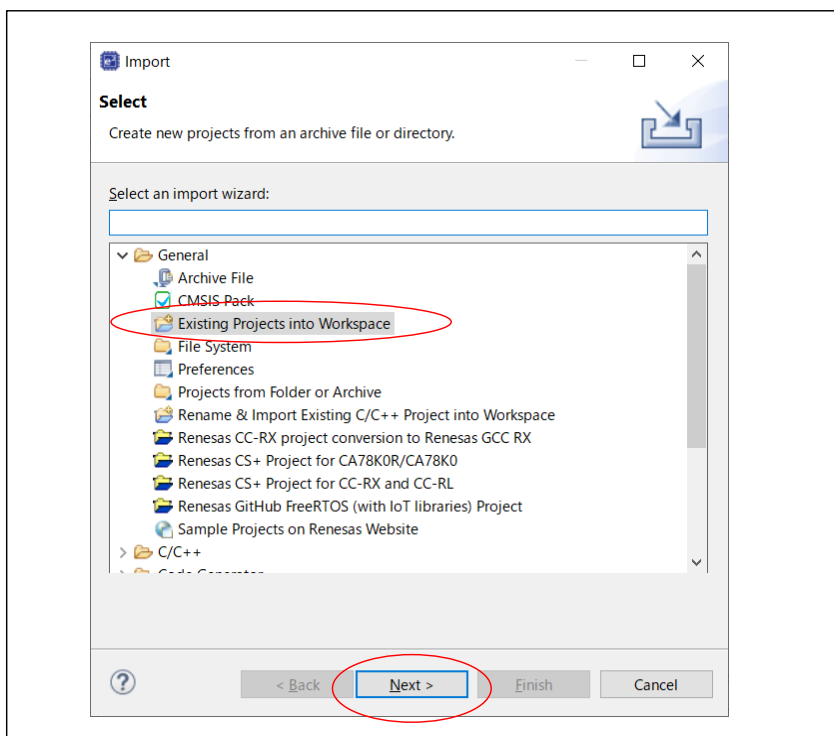
After starting e² studio, specify any folder at the [Workspace:] on the [e² studio Launcher] and click the [Launch] button.

Figure 2-3 [e² studio Launcher]



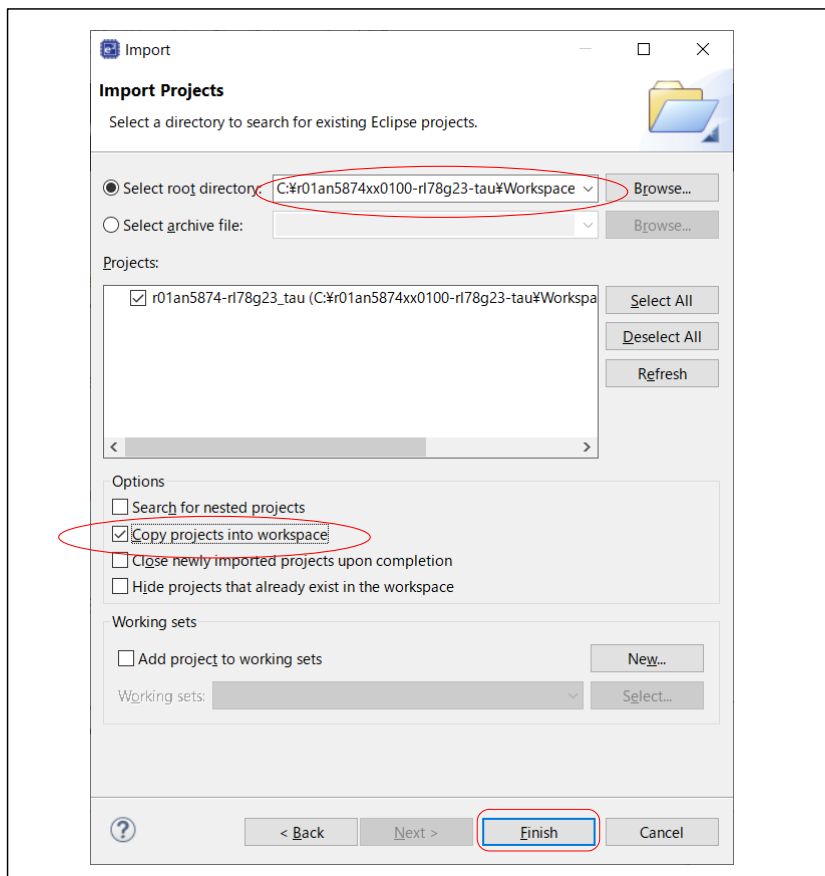
Click the [File] – [Import] menu of e² studio. Then, click the [General] - [Existing Projects into Workspace] and click the [Next] button on the [Import] dialog.

Figure 2-4 [Import] dialog (1/2)



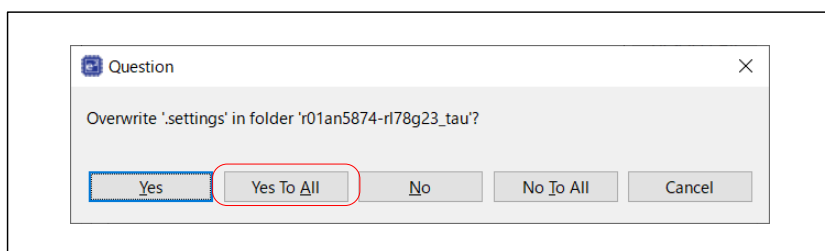
Specify the folder where the destination project exists at the [Select root directory:] on the [Import] dialog. Place a check mark in the [Copy projects into workspace] and click the [Finish] button.

Figure 2-5 [Import] dialog (2/2)



Click the [Yes To All] button on the [Question] dialog.

Figure 2-6 [Question] dialog

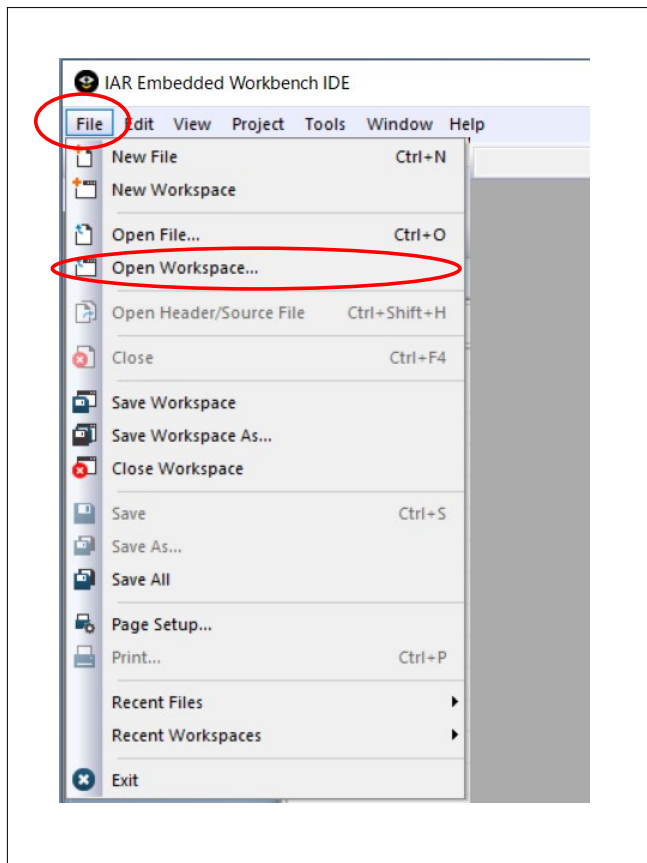


2.3.3 IAR EW for Renesas RL78

Start IAR EW for Renesas RL78 and open the destination workspace (project).

After starting IAR EW for Renesas RL78, click the [File] - [Open Workspace] menu and load the destination workspace: .eww file.

Figure 2-7 Open the destination workspace (.eww)



2.4 Change the compiler version

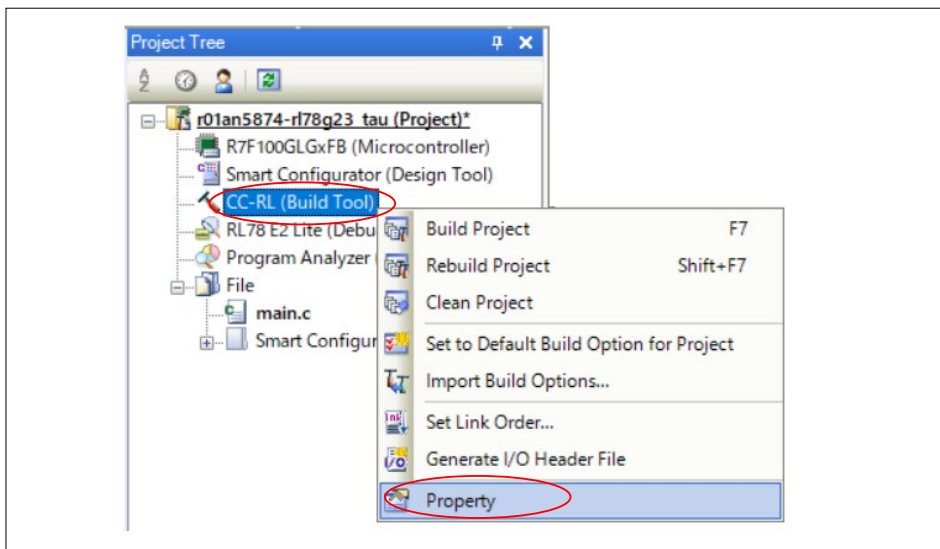
This section describes how to change the compiler version for each development tool (IDE).

2.4.1 CS+ for CC

Check the compiler version that the project uses and change the version if you want to change.

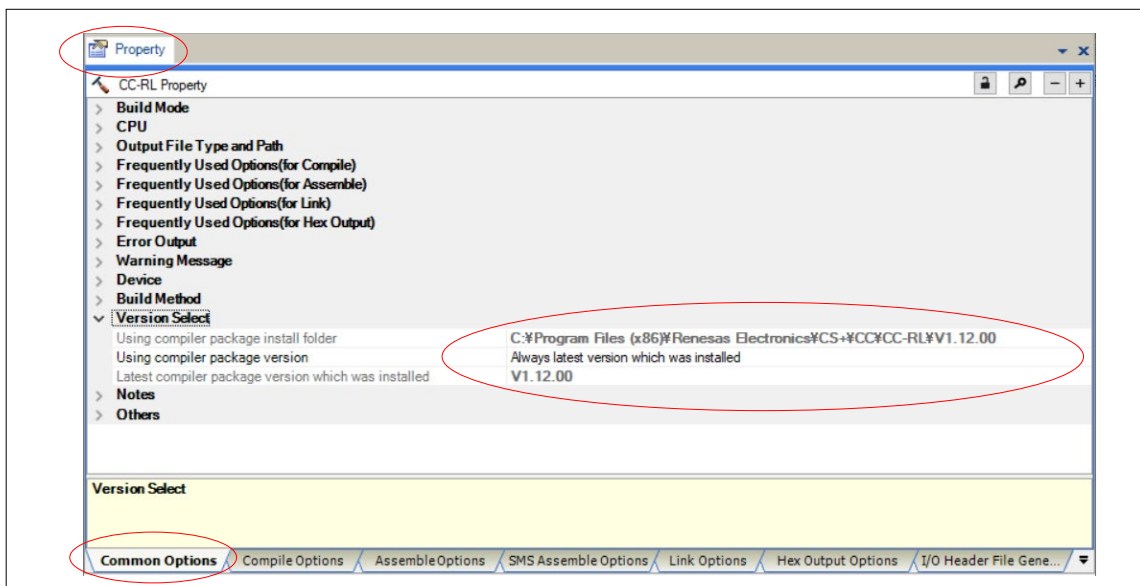
On the Project Tree of CS+ for CC, right-click the [CC-RL (Build Tool)] and click the [Property] menu.

Figure 2-8 CC-RL property



On the [Property] – [Common] tab, specify the compiler version to use.

Figure 2-9 CC-RL property: Version Select

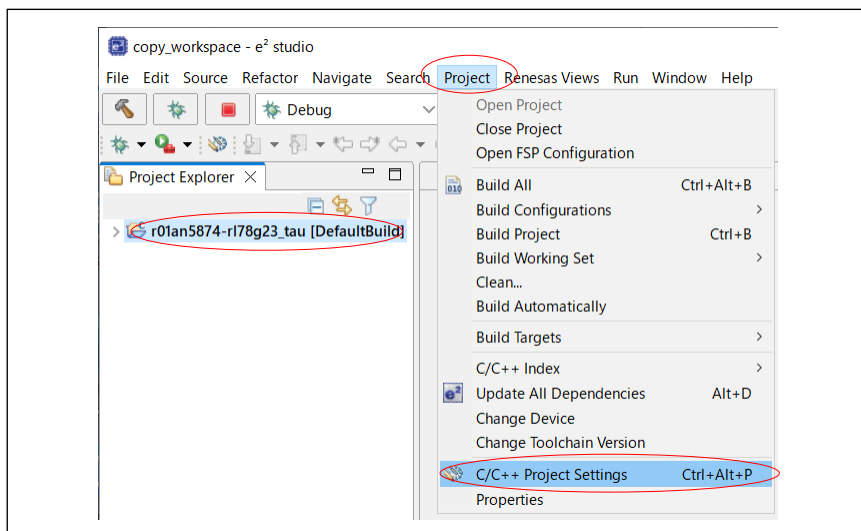


2.4.2 e² studio

Check the compiler version that the project uses and change the version if you want to change.

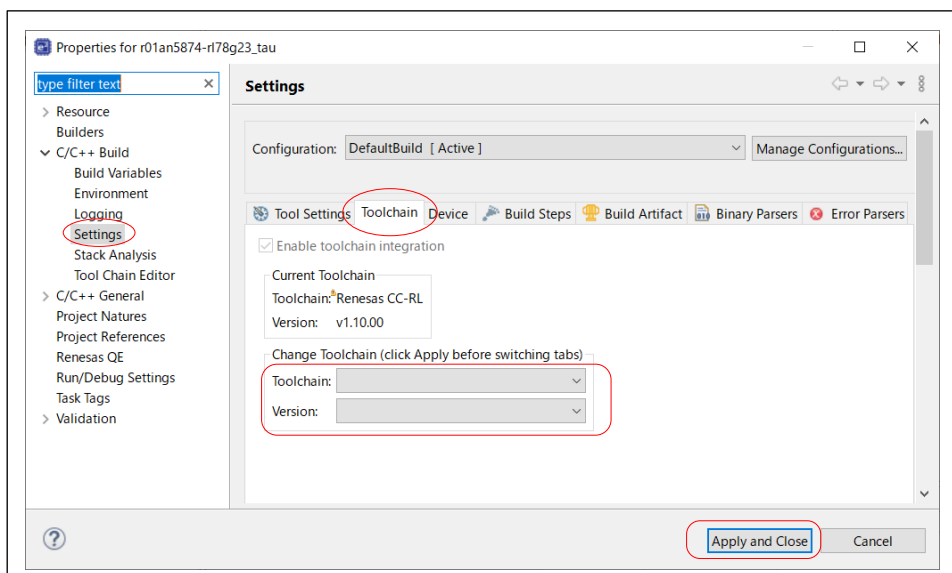
On the Project Explorer of e² studio, click the project name and click the [Project] – [C/C++ Project Settings] menu.

Figure 2-10 Toolchain settings (1/2)



On the [C/C++ Build] – [Settings] – [Toolchain] tab of the [Property] dialog, specify the version to use and click the [Apply and Close] button.

Figure 2-11 Toolchain settings (2/2)



2.4.3 IAR EW for Renesas RL78

Change the version of IAR EW for Renesas RL78 if you want to change the compiler version. The compiler included in the IAR EW Renesas RL78 is used.

2.5 Check code changes

Check if SC generated codes in the source project have been modified.

The target are files indicated by (1) in Figure 1-9. Open the target file in an editor and check if there are comments indicating modifications in the SC generated code. If there are any modifications, make the same modifications to the file indicated by (3) in Figure 1-9.

Figure 2-12, Figure 2-13 and Figure 2-14 show the project tree image of each IDE.

In the example of each figure, the files under the Config_XXX and general folders indicated by the red frame are the target to check.

Figure 2-12 CS+ for CC

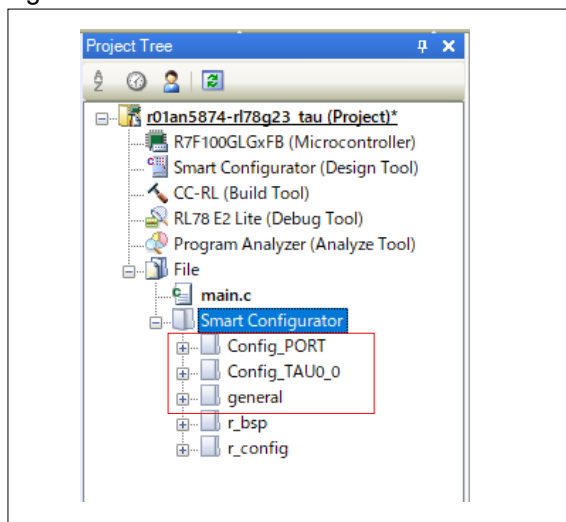


Figure 2-13 e² studio

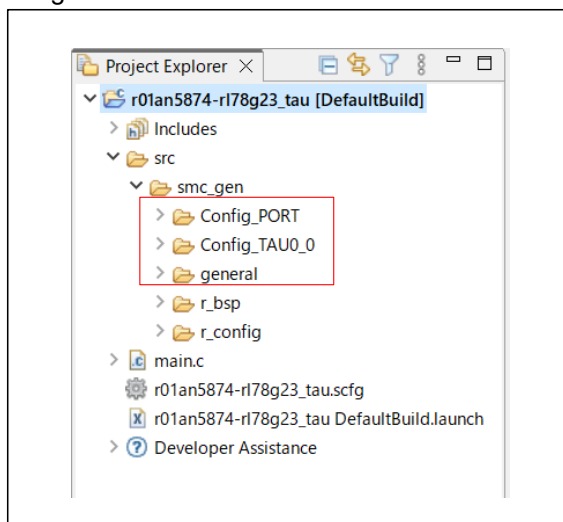
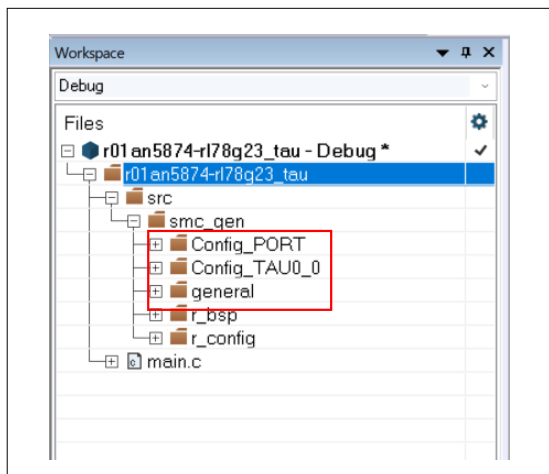


Figure 2-14 IAR EW for Renesas RL78



2.6 Change resources

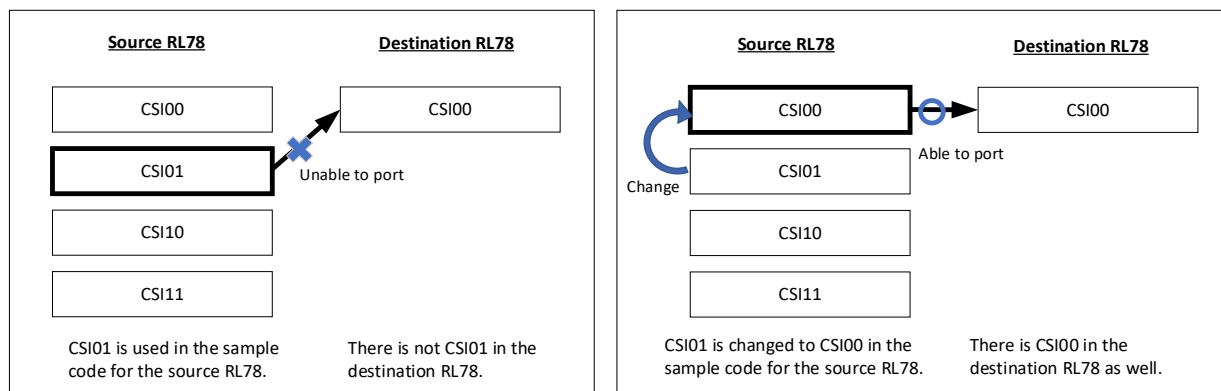
If you select "Change to another channel" as the porting method for peripheral functions in 2.1 Advance preparation, change the channel (resource) on Smart Configurator (SC) before changing the device to the destination RL78. If you select "Use the same channel", this step is not necessary. In addition, the used pin is checked in "2.10.5 Pins".

The following is an example for the case of "Change to another channel".

- The source RL78 has CSI00 and CSI01. CSI01 is used in the sample code for the source RL78.
- The destination RL78 has CSI00.

In this case, change the resource from CSI01 to CSI00 on the source RL78, and then change the target device of the sample code to the destination RL78.

Figure 2-15 Image of resource change



Remark. Resources can be changed even after "2.8 Change the device". However, in this case, the configuration setting of CSI01 is deleted when changing the device, therefore, it is necessary to manually set configuration for CSI00 after changing the device.

The procedure for starting SC differs for each IDE. However, the basic operation of SC is the same in all IDEs. First, the procedure for starting SC in each IDE is explained. After that, how to change resource is explained.

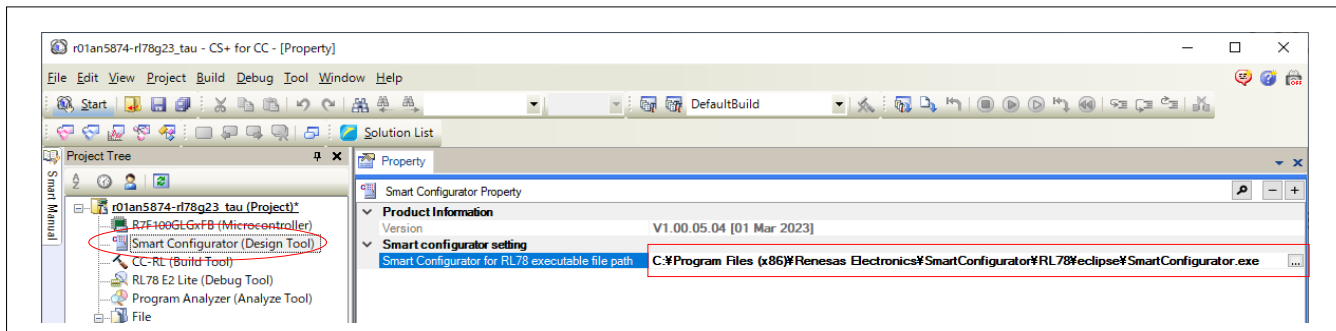
Table 2-5 Procedure of starting SC (Smart Configurator)

IDE	Startup procedure
CS+ for CC	Start from CS+ for CC
e ² studio	Start in e ² studio (plug-in)
IAR EW for Renesas RL78	Start individually without IAR EW

2.6.1 SC startup procedure: CS+ for CC

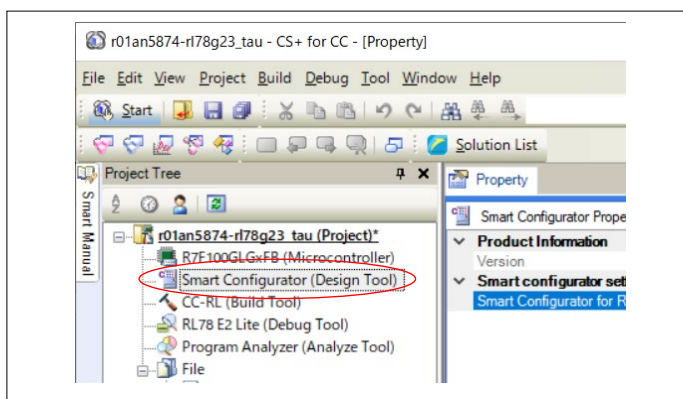
On the Project Tree of CS+ for CC, click the [Smart Configurator (Design Tool)] to open the [Property] tab for SC and specify the path where exe file of SC is installed.

Figure 2-16 Specifying SC executable file path



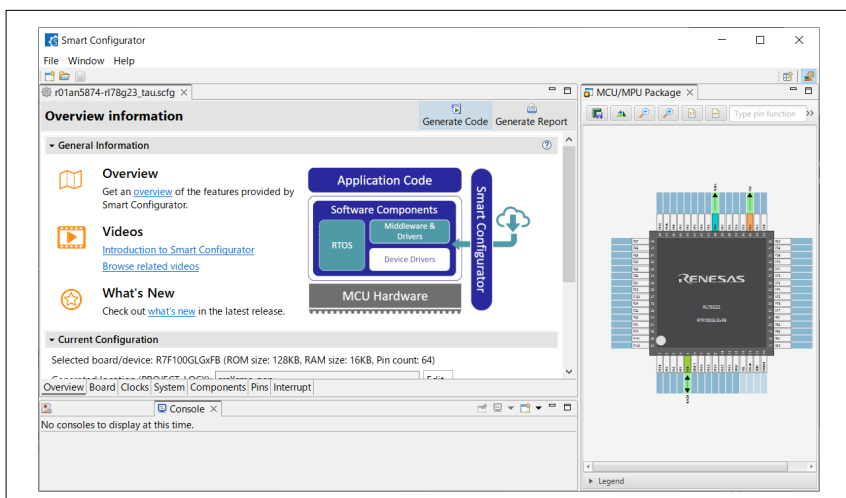
On the Project Tree of CS+ for CC, double-click the [Smart Configurator (Design Tool)] to start SC.

Figure 2-17 Starting SC



SC is started.

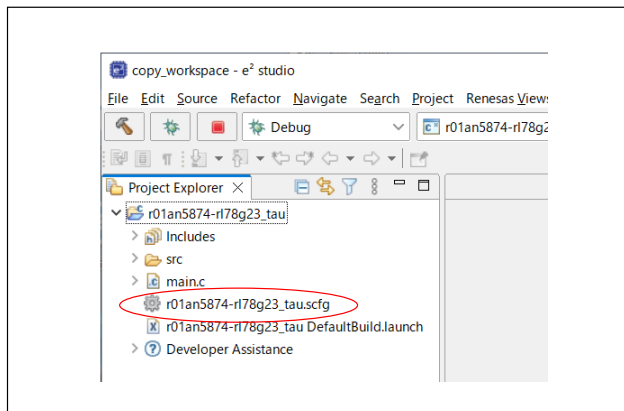
Figure 2-18 Smart Configurator (SC)



2.6.2 SC startup procedure: e² studio

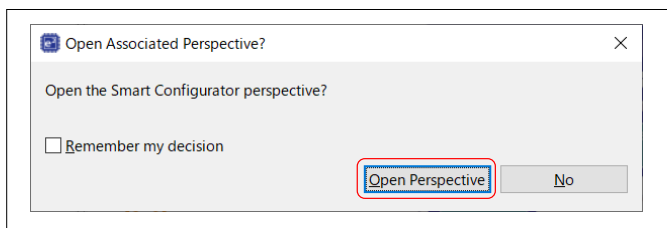
On the Project Explorer of e² studio, double-click .scfg file to start SC.

Figure 2-19 Starting SC



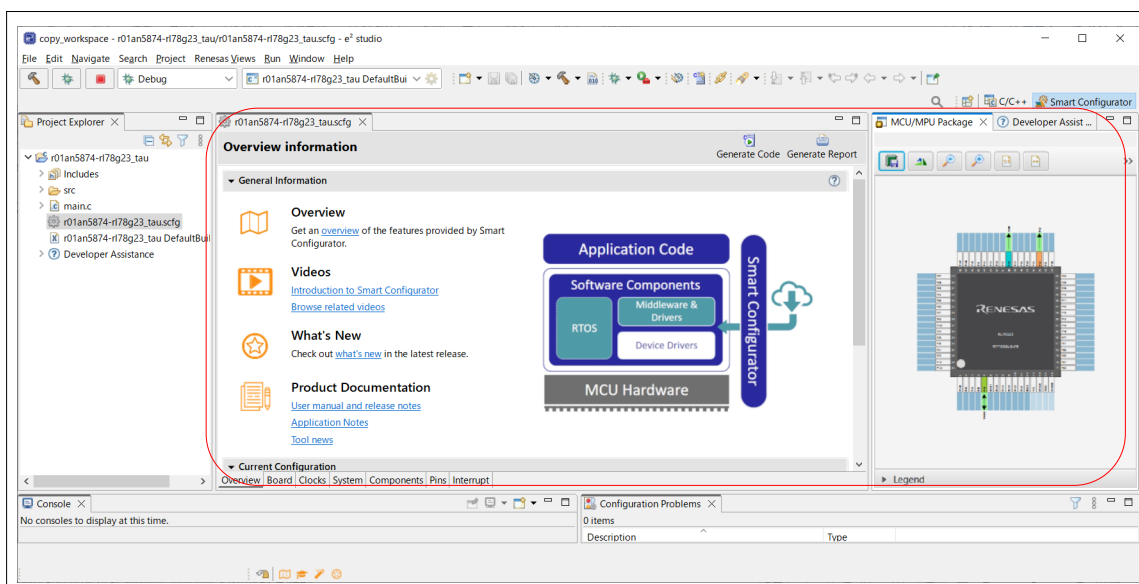
Click the [Open Perspective] button on the [Open Associated Perspective?] dialog.

Figure 2-20 [Open Associated Perspective?] dialog



SC is started.

Figure 2-21 Smart Configurator (Plug-in)

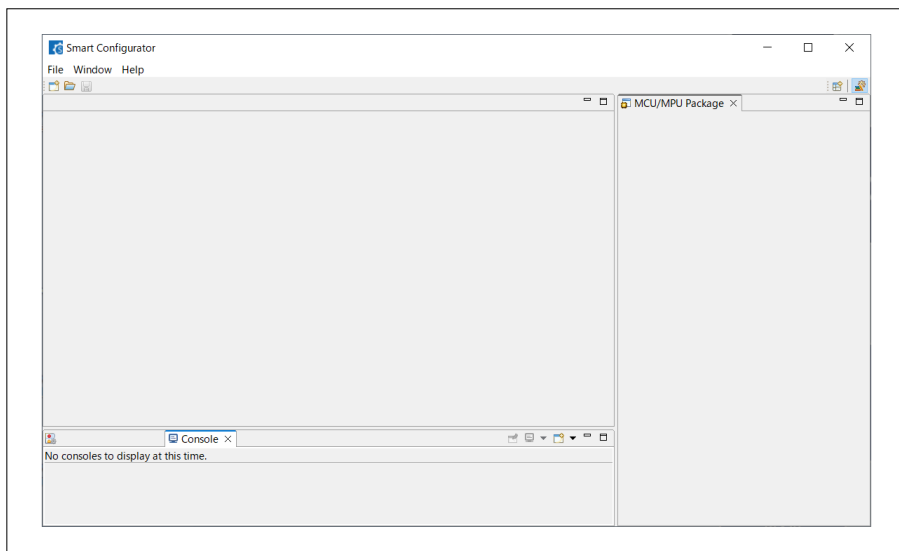


2.6.3 SC startup procedure: IAR EW for Renesas RL78

From the [Start] menu of Windows, click the [Renesas Electronics Smart Configurator] – [Smart Configurator for RL78 Vx.x.x] to start SC. (x.x.x: version number)

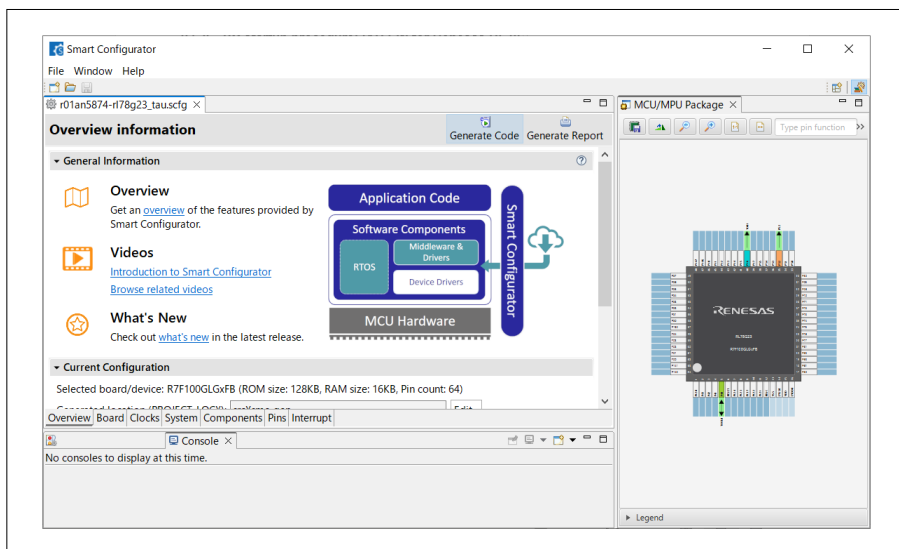
SC is started.

Figure 2-22 Smart Configurator (SC)



Click the [File] – [Open] menu and open .scfg file for the destination RL78.

Figure 2-23 Smart Configurator (SC) after opening .scfg



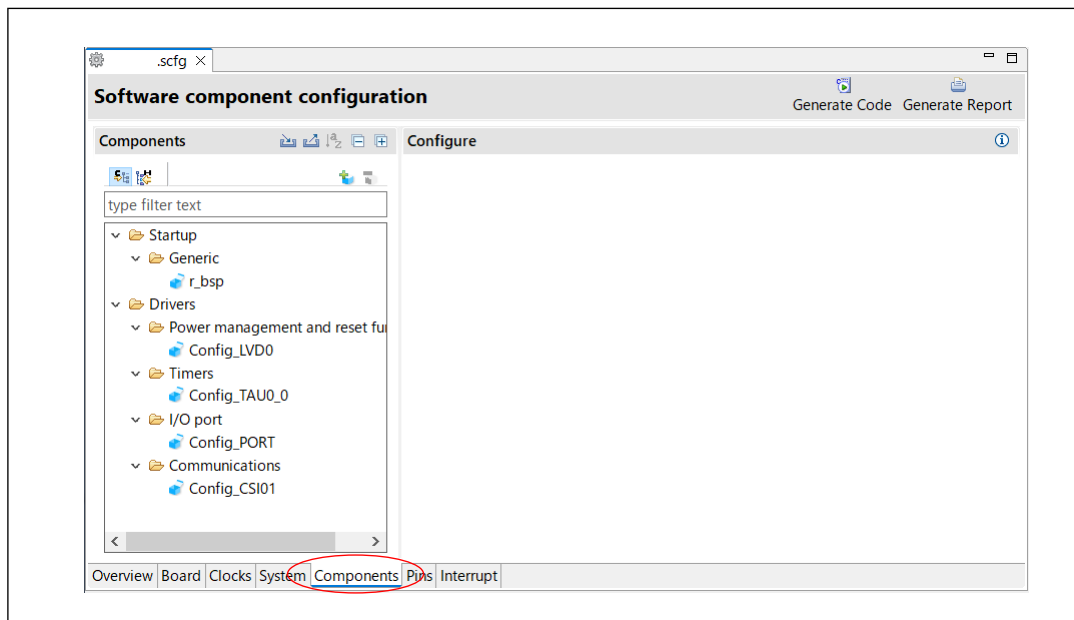
Remark. By registering the SC executable file in the [Tools] – [Configure Tools] menu of IAR EW for Renesas RL78, the menu to start SC will be added to the [Tools] menu of IAR EW, and SC can be started from IAR EW.

2.6.4 Change resources

The procedure for changing from CSI01 to CSI00 is explained as an example.

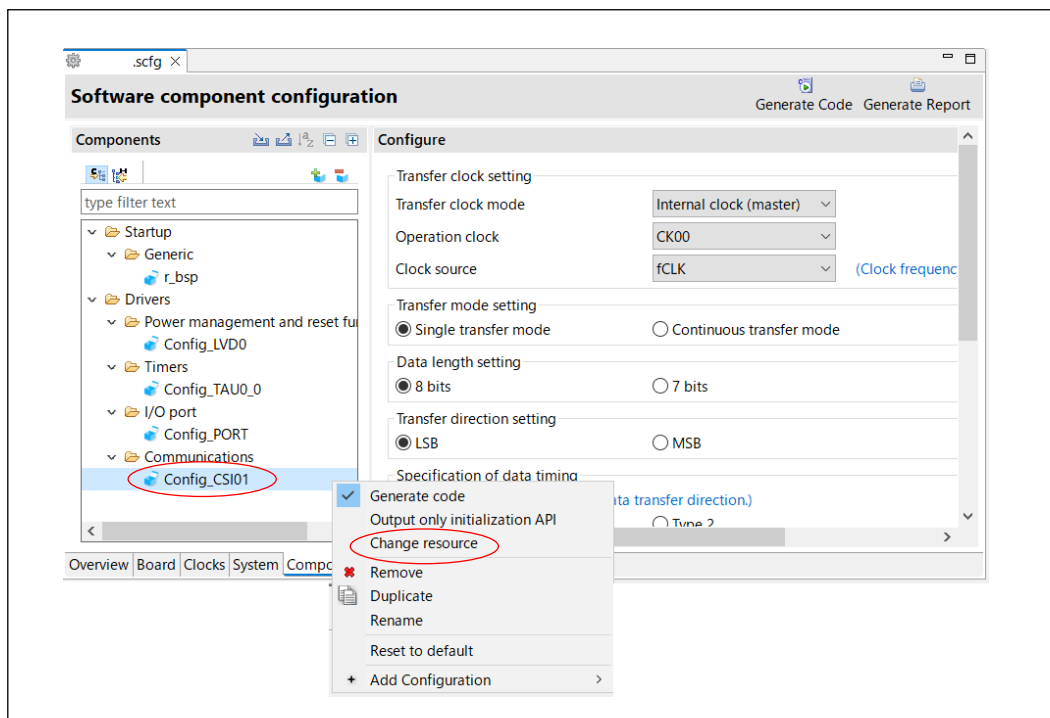
Click the [Components] tab on SC.

Figure 2-24 [Components] tab



Change the resource. Right-click the [Config_CSI01] and click the [Change resource] menu.

Figure 2-25 [Change resource] menu



Change the resource on the [Resource Selection] dialog. After changing the resource, click the [Finish] button.

Figure 2-26 [Resource selection] dialog (1/4)

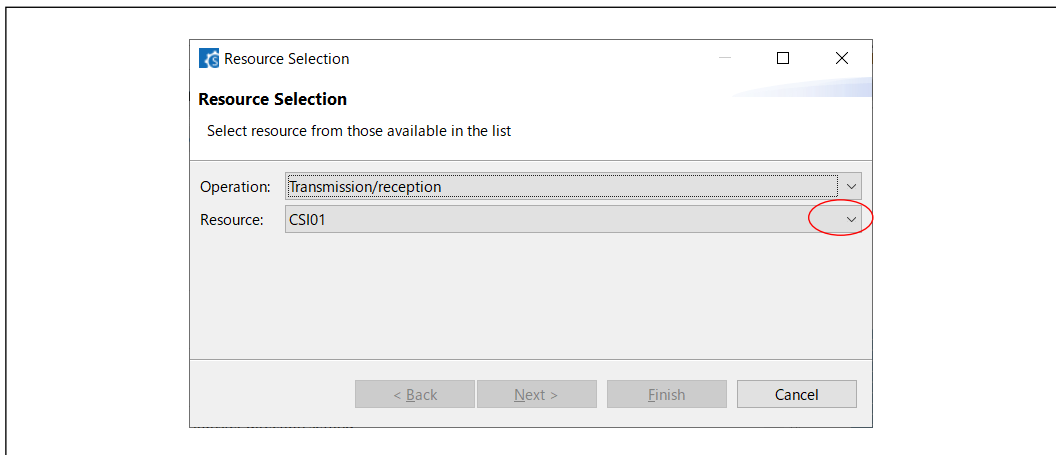


Figure 2-27 [Resource selection] dialog (2/4)

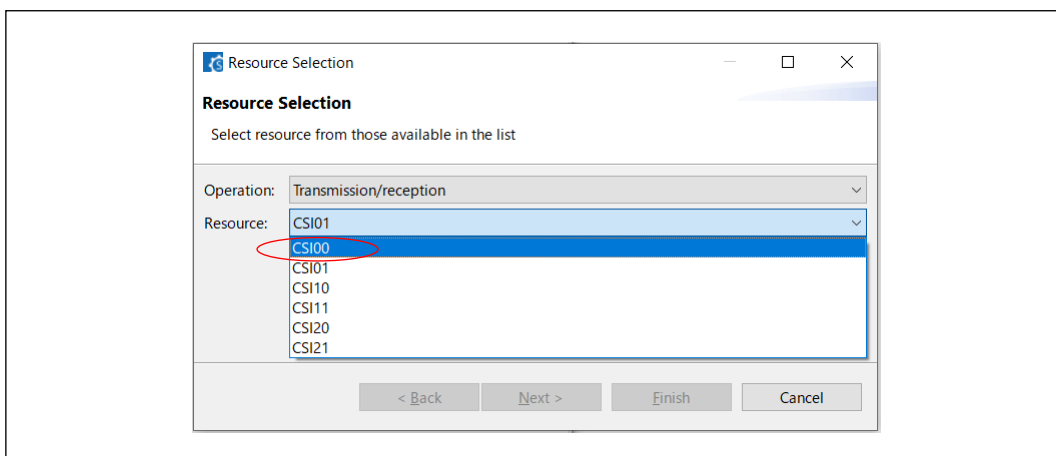


Figure 2-28 [Resource selection] dialog (3/4)

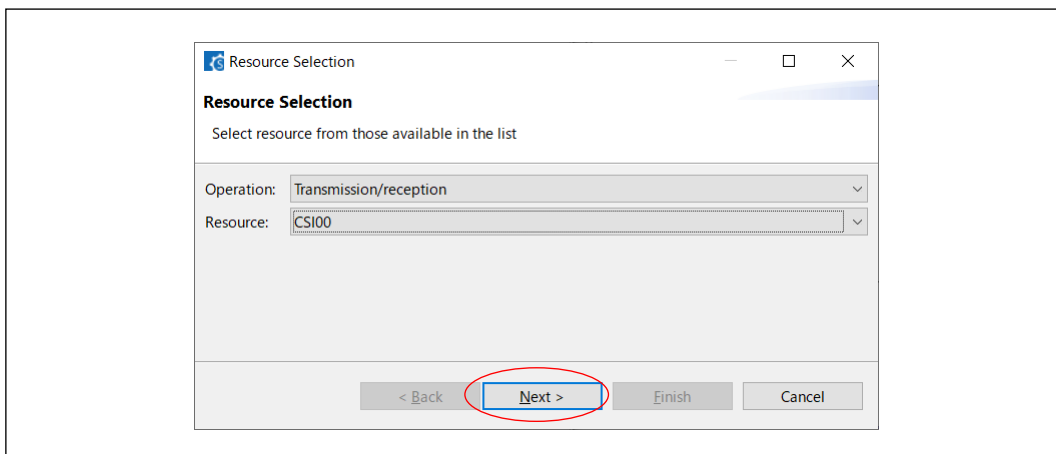
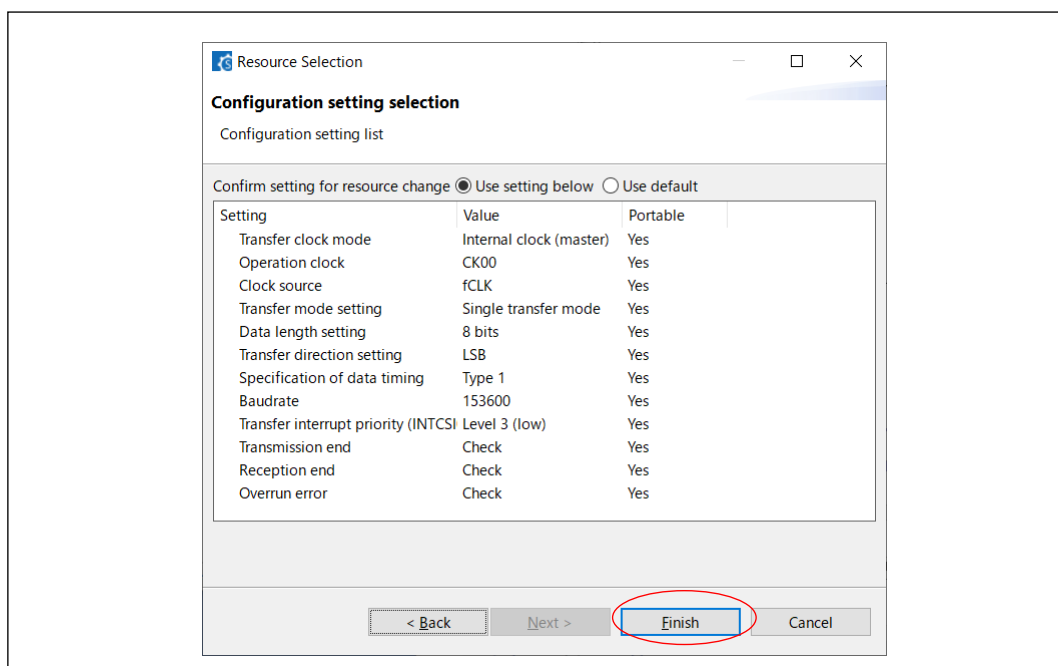


Figure 2-29 [Resource selection] dialog (4/4)

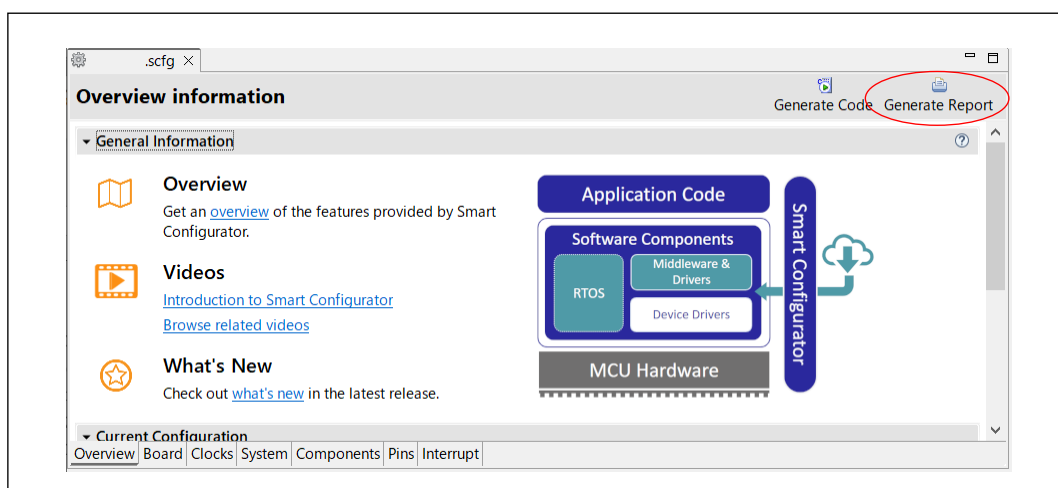


2.7 Generate report of SC configuration

Generate the report of SC configuration as the configuration report file before changing the device to the destination RL78 on SC. This configuration report will be referred to confirm the configuration of the source RL78 in "2.10 Change settings and resources".

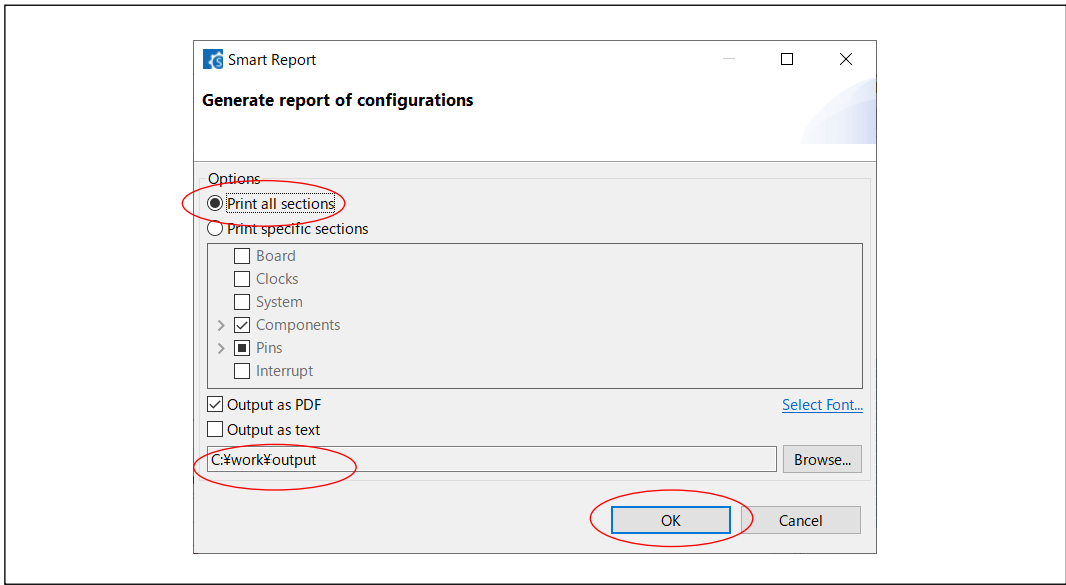
Click the [Generate report] button.

Figure 2-30 [Generate Report] button



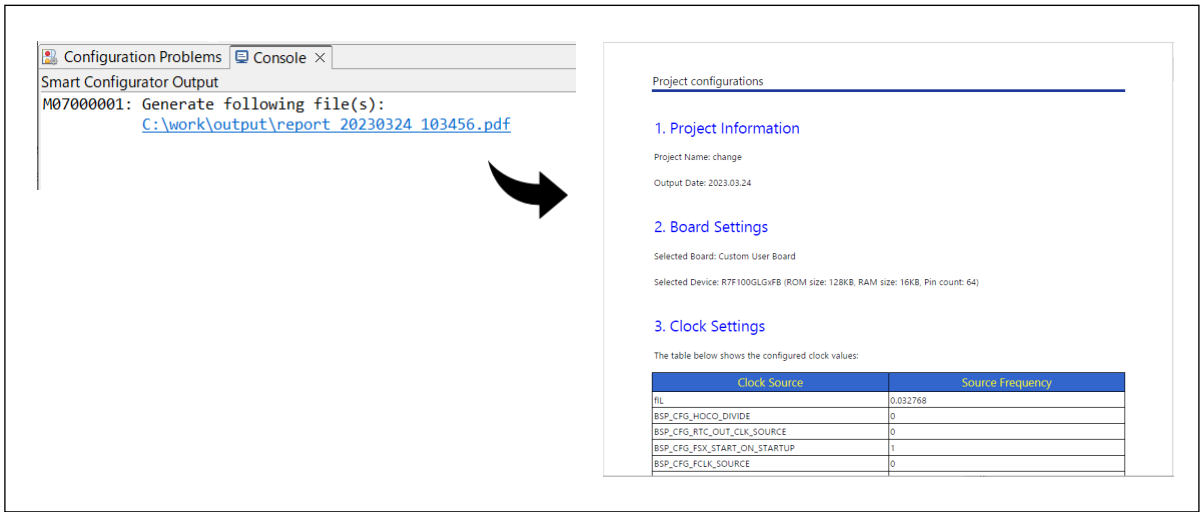
On the [Smart Report] dialog, select the [Print all sections] and specify the output folder, then click the [OK] button.

Figure 2-31 [Smart Report] dialog



The configuration report file is generated in the specified folder. The configuration report file can be opened form the link on the [Console] tab of SC.

Figure 2-32 Configuration report file



2.8 Change the device

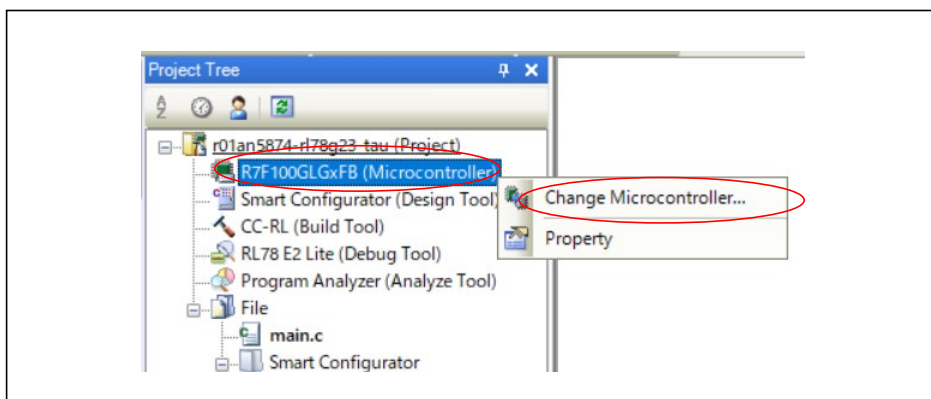
This section describes how to change the device from the source RL78 to the destination RL78 for each development tool (IDE).

2.8.1 CS+ for CC

Change the device on both CS+ for CC and Smart Configurator (SC).

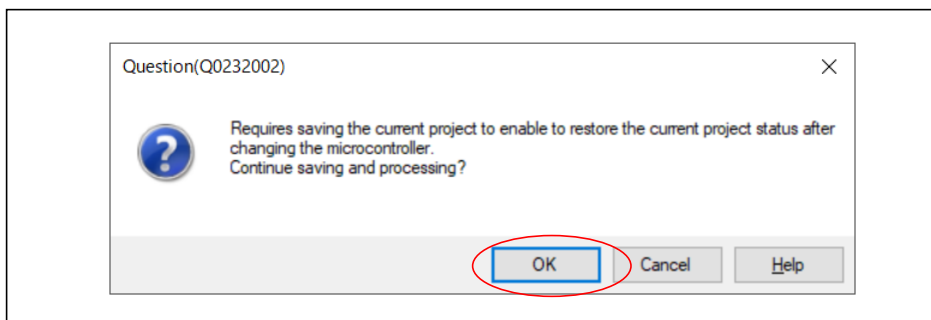
On the [Project Tree] of CS+ for CC, right-click the microcontroller name and click the [Change Microcontroller] menu.

Figure 2-33 Change Microcontroller



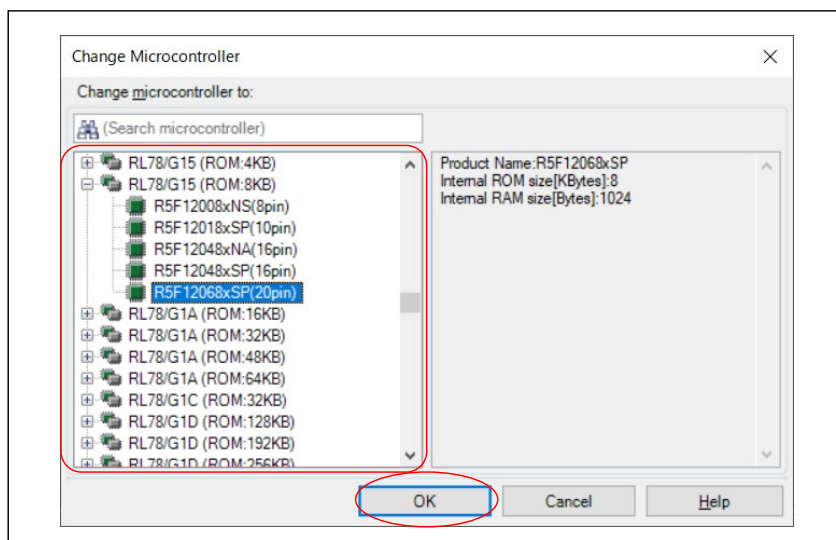
On the [Question] dialog, click the [OK] button.

Figure 2-34 [Question] dialog



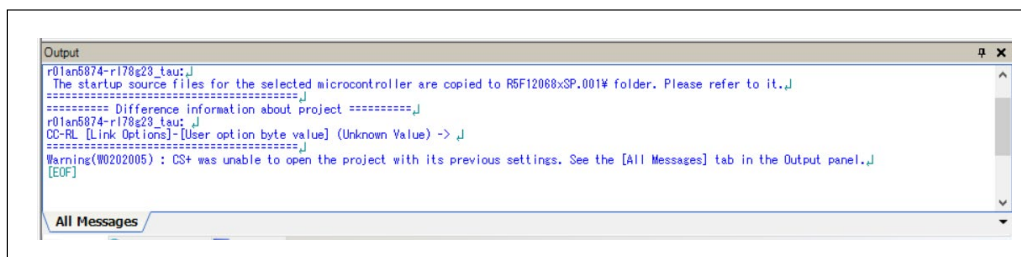
On the [Change Microcontroller] dialog, specify the destination RL78 and click the [OK] button.

Figure 2-35 [Change Microcontroller] dialog



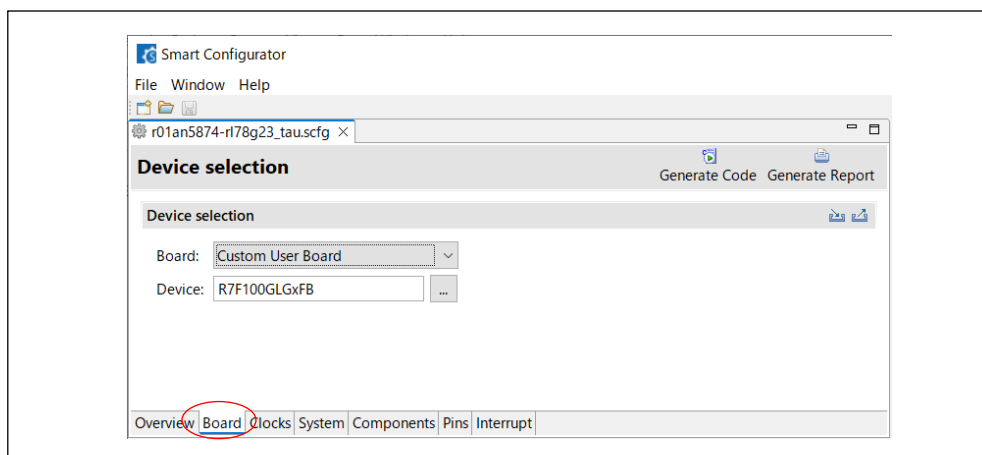
Remark. CC-RL option settings may not be ported due to differences in device specifications between the source RL78 and the destination RL78. Anything that could not be ported will be displayed on the [Output] tab of CS+ for CC. When the code for the destination RL78 is generated in “2.12 Code generation”, settings such as the user option byte are reconfigured according to the destination RL78, therefore, only check the contents here.

Figure 2-36 Message example: when changing the device on CS+ for CC



Next, change the device on SC. Click the [Board] tab on SC.

Figure 2-37 [Board] tab on SC

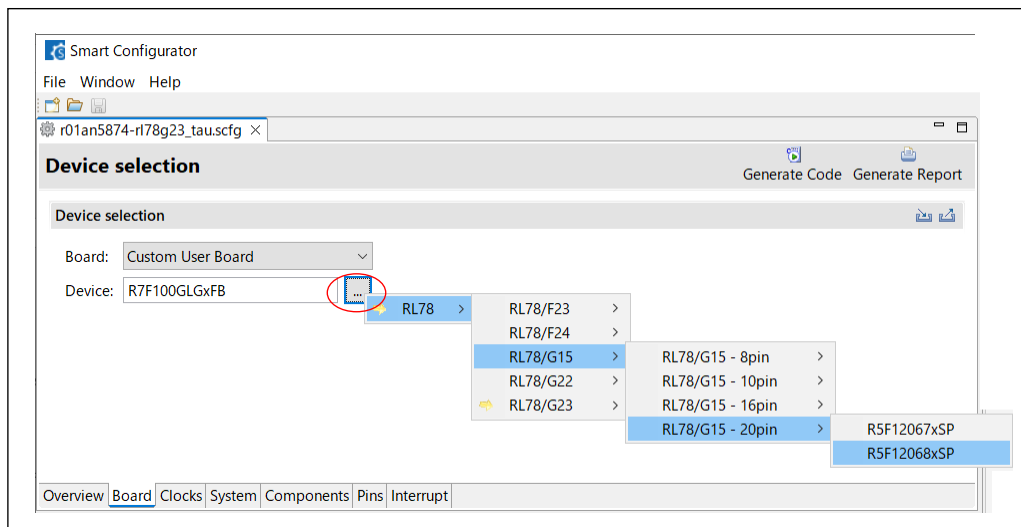


RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

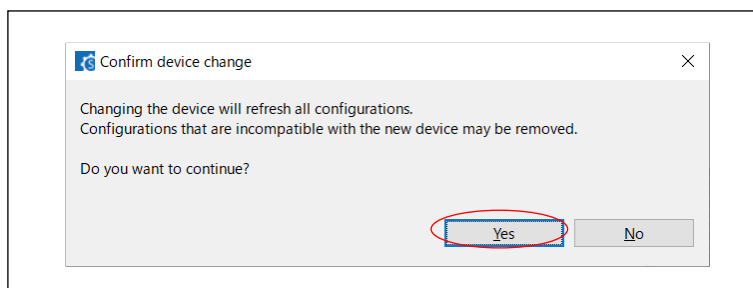
Click the [...] and specify the destination RL78.

Figure 2-38 Device selection on SC



The [Confirm device change] dialog appears. Click the [Yes] button.

Figure 2-39 [Confirm device change] dialog

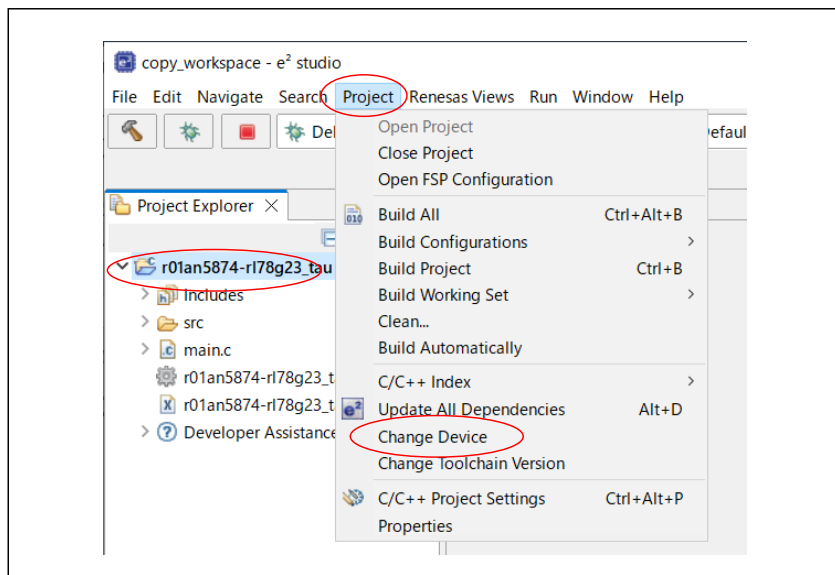


2.8.2 e² studio

Change the device on e² studio.

On the [Project Explorer] of e² studio, click the project and click the [Project] – [Change Device] menu.

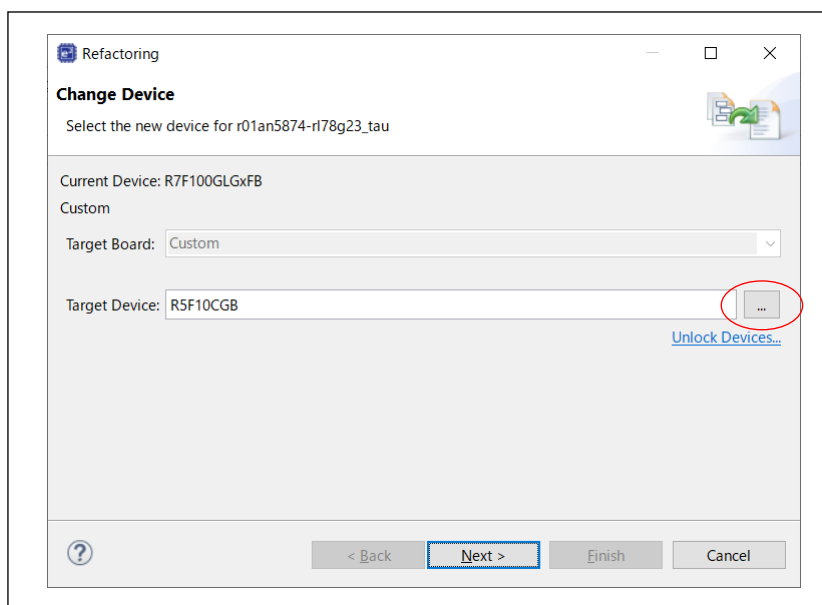
Figure 2-40 [Project] – [Change Device] menu



On the [Refactoring] dialog, click the [...] button.

(It is possible to input the name as the destination RL78 in the [Target Device] textbox.)

Figure 2-41 [Refactoring] dialog (1/4)

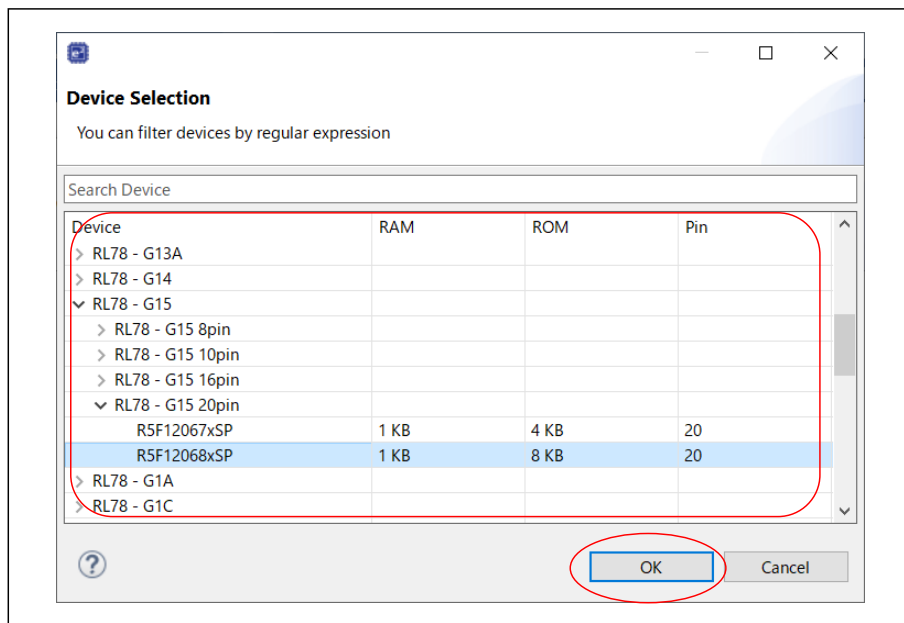


RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

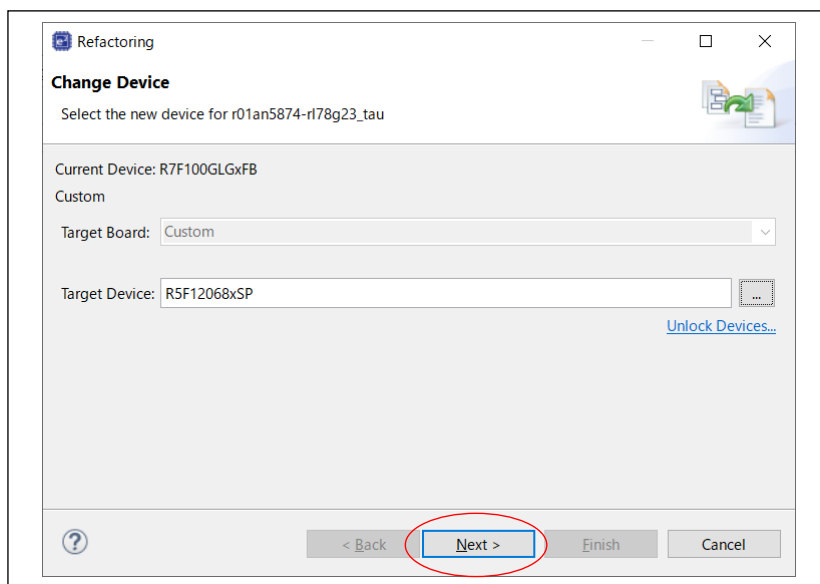
On the [Device Selection] dialog, specify the destination RL78 and click the [OK] button.

Figure 2-42 [Device Selection] dialog



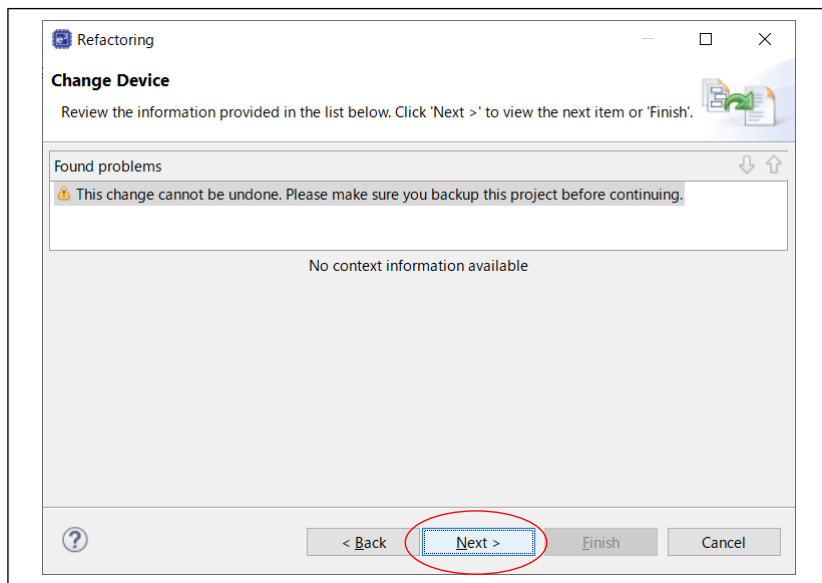
On the [Refactoring] dialog, click the [Next] button.

Figure 2-43 [Refactoring] dialog (2/4)



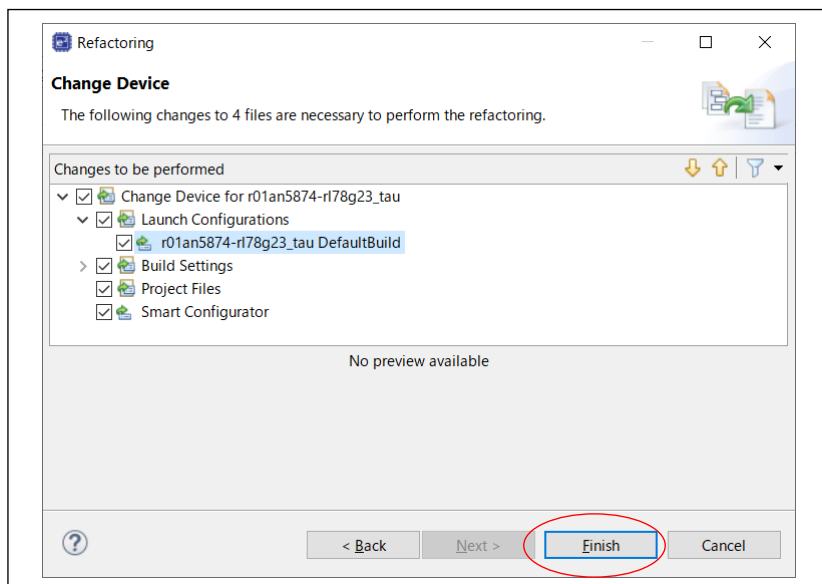
Check the information and click the [Next] button.

Figure 2-44 [Refactoring] dialog (3/4)



Check the information and click the [Finish] button. It is not necessary to change the checkbox.

Figure 2-45 [Refactoring] dialog (4/4)

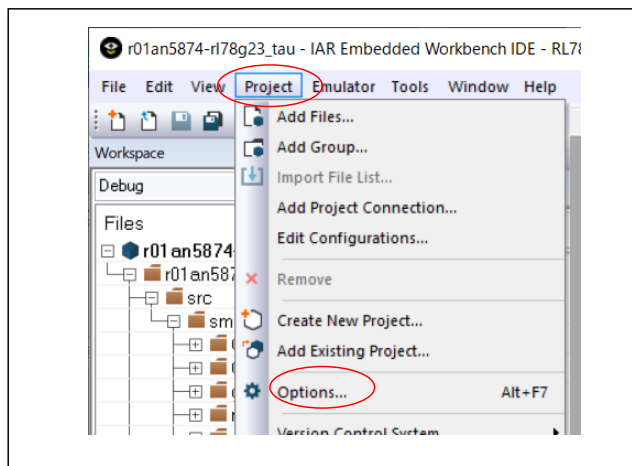


2.8.3 IAR EW for Renesas RL78

Change the device on both IAR EW for Renesas RL78 and Smart Configurator (SC).

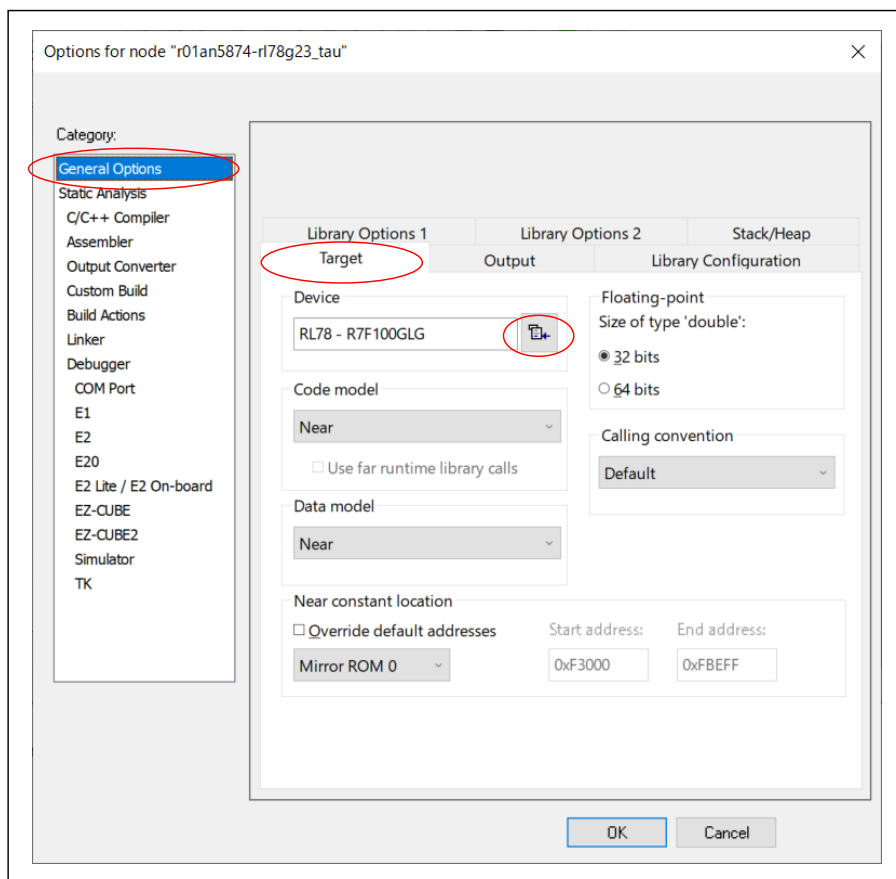
On IAR EW for Renesas RL78, click the [Project] - [Option] menu.

Figure 2-46 [Project] – [Option] menu



On the [Option] dialog, click the [General Options] and the [Target] tab. Then, click the button at the [Device].

Figure 2-47 [Option] dialog (1/2)

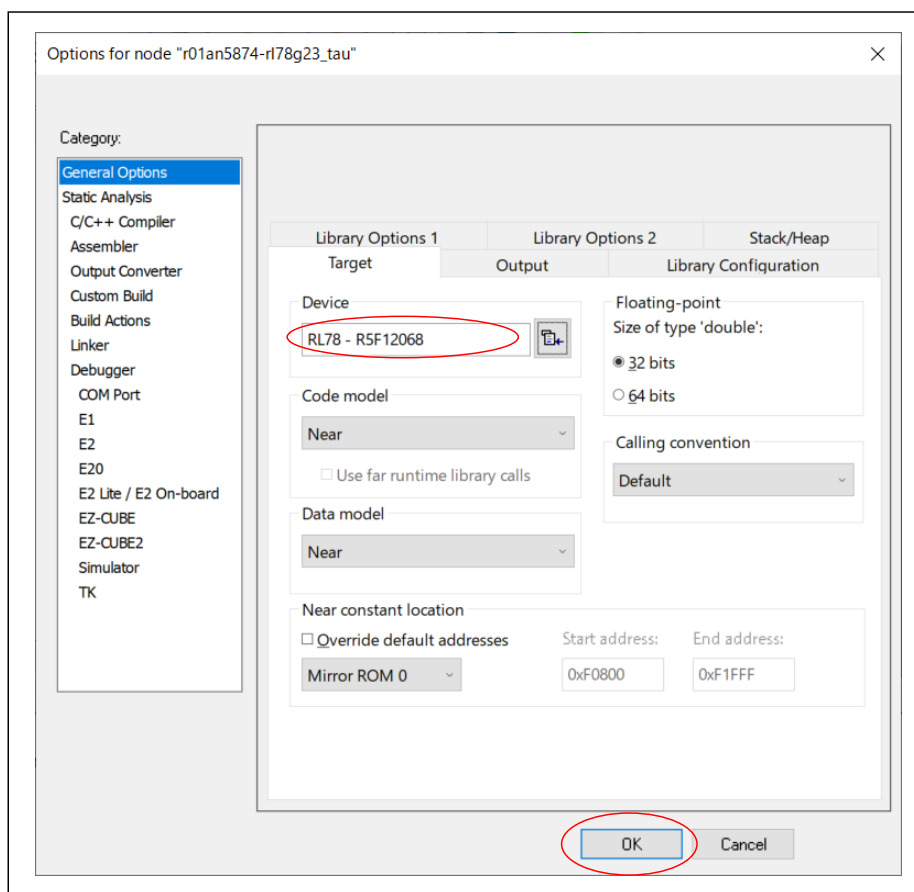


RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

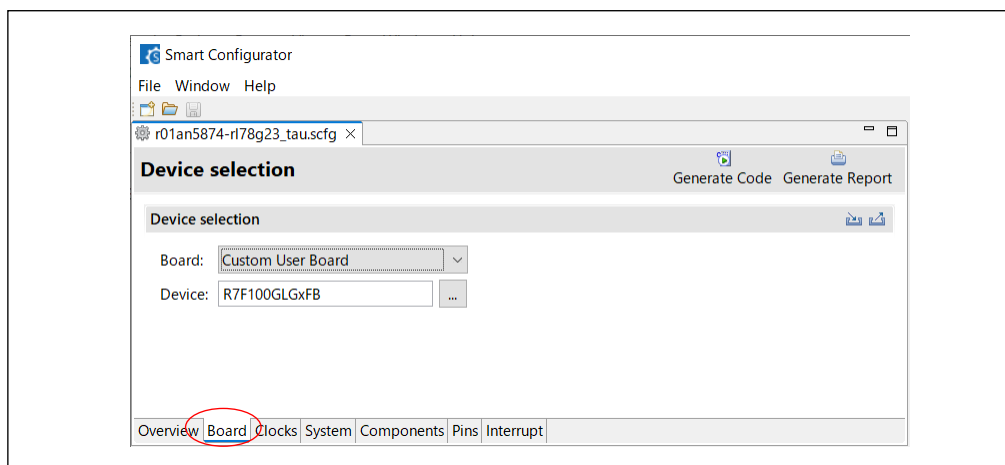
Specify the destination RL78 from the device name list. Then, click the [OK] button.

Figure 2-48 [Option] dialog (2/2)



Next, change the device on SC. Click the [Board] tab on SC.

Figure 2-49 [Board] tab on SC

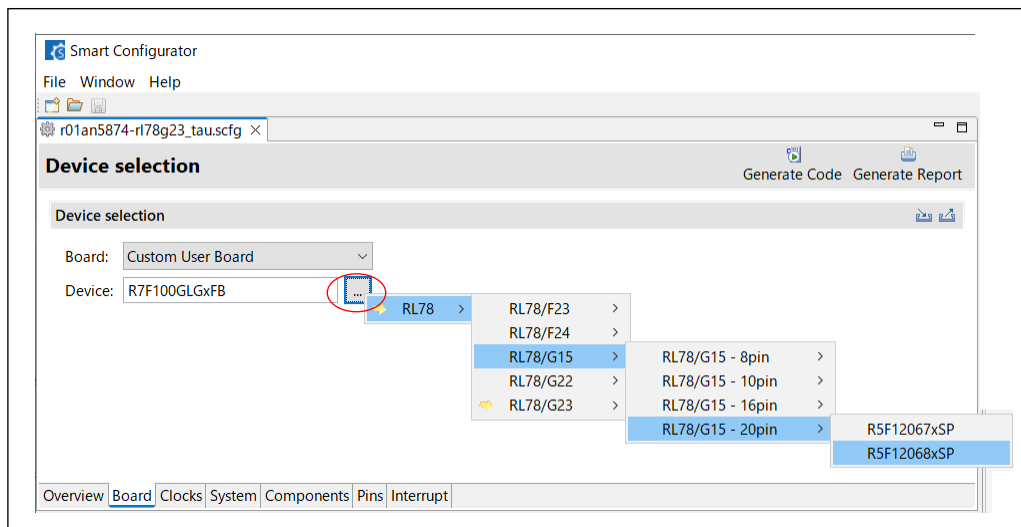


RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

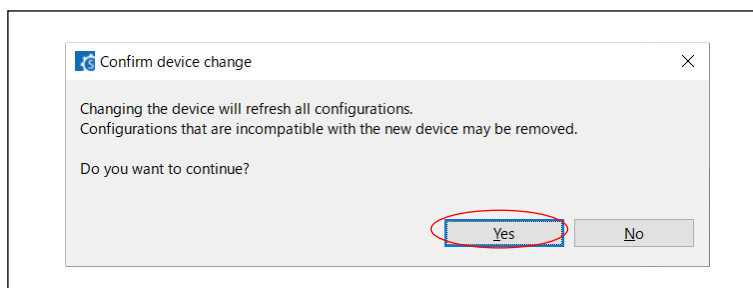
Click the [...] and specify the destination RL78.

Figure 2-50 Device selection on SC



The [Confirm device change] dialog appears. Click the [Yes] button.

Figure 2-51 [Confirm device change] dialog

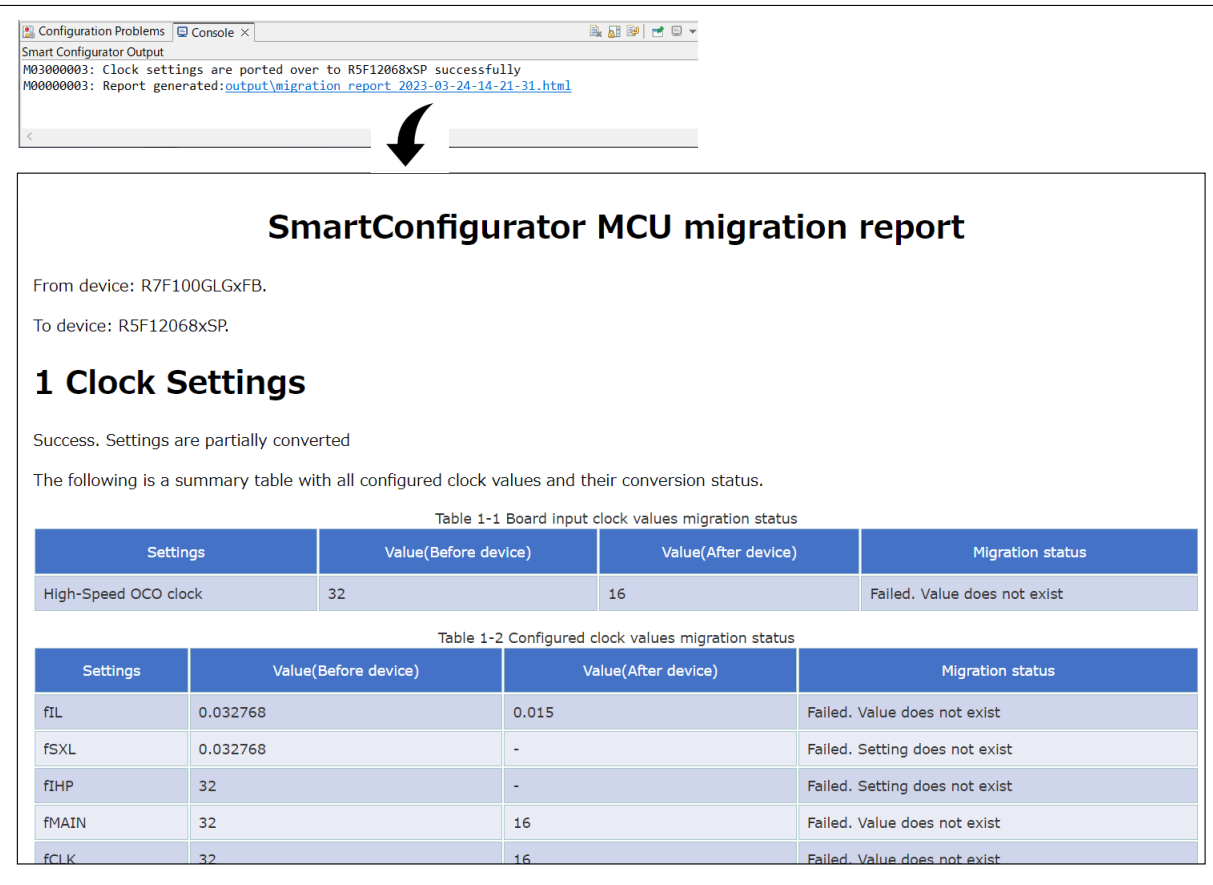


2.9 Check the SC migration report

Smart Configurator (SC) generates the migration report when changing the device in “2.8 Change the device”. Check the result by referring to the migration report. Take necessary action according to the migration result.

On the [Console] tab of SC, click the link to show the migration report.

Figure 2-52 Migration report



The screenshot shows the Smart Configurator interface. The top console window displays the following output:

```
Smart Configurator Output
M03000003: Clock settings are ported over to R5F12068xSP successfully
M00000003: Report generated: output\migration\_report\_2023-03-24-14-21-31.html
```

A large black arrow points from the link in the console to the migration report preview below.

SmartConfigurator MCU migration report

From device: R7F100GLGxFB.
To device: R5F12068xSP.

1 Clock Settings

Success. Settings are partially converted

The following is a summary table with all configured clock values and their conversion status.

Table 1-1 Board input clock values migration status

Settings	Value(Before device)	Value(After device)	Migration status
High-Speed OCO clock	32	16	Failed. Value does not exist

Table 1-2 Configured clock values migration status

Settings	Value(Before device)	Value(After device)	Migration status
fIL	0.032768	0.015	Failed. Value does not exist
fSXL	0.032768	-	Failed. Setting does not exist
fIHP	32	-	Failed. Setting does not exist
fMAIN	32	16	Failed. Value does not exist
fCLK	32	16	Failed. Value does not exist

Table 2-6 and Table 2-7 show the contents in the migration report.

Table 2-6 Migration status in migration report (1/2)

Item	Status	Cause / Action	Reference chapter
Clock Settings	Success.	[Cause] The clock configuration of the source RL78 was migrated to the destination RL78. [Action] No action required.	-
	Failed. Value does not exist	[Cause] The frequency used on the source RL78 could not be migrated to the destination RL78. [Action] Check the frequency of the applicable item and change the frequency accordingly on the destination RL78.	2.10.1
	Failed. Setting does not exist	[Cause] The setting item that exist for the source RL78 does not exist for the destination RL78. [Action] If the applicable item is used in the sample code, it is impossible to port the sample code.	-
Interrupt	Success.	[Cause] The configuration of the source RL78 was migrated to the destination RL78. [Action] No action required.	-
	Failed. Setting does not exist	[Cause] The interrupt source used on the source RL78 could not be migrated to the destination RL78. [Action] If the applicable item is not used in the sample code, no action required. If the applicable item is used in the sample code, it is impossible to port the sample code.	2.10.3
Pins	Success.	[Cause] The port or alternative function configuration of the source RL78 was migrated to the destination RL78. [Action] After changing the device, be sure to check the pin setting status even if the result is "Success."	2.10.5
	Discarded. Function is not available	[Cause] The port or alternative function that exist for the source RL78 does not exist on the destination RL78. [Action] Use another pin instead. If it cannot be changed to another pin that has the same function, it is impossible to port the sample code.	2.10.5

Table 2-7 Migration status in migration report (2/2)

Item	Status	Cause / Action	Reference chapter
Software Component Note 1	Success. Settings are fully converted	[Cause] The configuration of the source RL78 was migrated to the destination RL78. [Action] No action required.	-
	Success. Settings are partially converted	[Cause] The partial configuration of the source RL78 was migrated to the destination RL78. [Action] Check the configuration of applicable component and the impact on the sample code. If the sample code is affected by the unconverted settings, it is impossible to port the sample code.	2.10.4(2)
	Fails. Channel does not exist in this device	[Cause] The channel used on the source RL78 does not exist on the destination RL78. [Action] If there is another channel available, manually add another channel with the same settings as the source RL78. If there is not another channel available, it is impossible to port the sample code. [Remark] In case the source RL78 and the destination RL78 have the same number channel, it may be possible to port the sample code by changing the resource (channel) before changing the device. (Reference: 2.6 Change resources)	-
	Fails. Component is not supported in this device	[Cause] The component (Peripheral) supported on the source RL78 does not exist on the destination RL78. [Action] If the applicable component is used in the sample code, it is impossible to port the sample code. However, in the case of the porting from RL78/G23 to RL78/G15, if the applicable component is the Voltage Detector (Config_LVD), it is possible to port the sample code by setting the Selectable Power-on-reset Circuit on SC's [System] tab.	2.10.4(1)
	Fails. Please upgrade to the latest version	[Cause] BSP version used on the source RL78 does not support the destination RL78. [Action] Upgrade BSP to the latest version.	2.10.4(3)
(Each component) Note 1	Success.	[Cause] The configuration of the source RL78 was migrated to the destination RL78. [Action] No action required.	-
	Failed. Setting does not exist	[Cause] The interrupt source used on the source RL78 does not exist on the destination RL78. [Action] Check the configuration of applicable component and the impact on the sample code. If the sample code is affected by the unconverted settings, it is impossible to port the sample code.	2.10.4(1)
System Settings	Success.	[Cause] The configuration of the source RL78 was migrated to the destination RL78. [Action] No action required.	-
	Failed. Setting does not exist	[Cause] The setting item that exist for the source RL78 does not exist for the destination RL78. [Action] Check the configuration of applicable item and the impact on the sample code. However, in most cases, the sample code can be ported without affecting its behavior because almost items are for the debug settings.	2.10.4(1)

Note 1. Make sure of "Config_PORT" (Port) in "2.10.5 Pins"

2.10 Change settings and resources

This section explains the actions required based on the migration report. “Clock settings”, “System Settings”, “Interrupt”, “Software Component” and “Pins” are explained with examples.

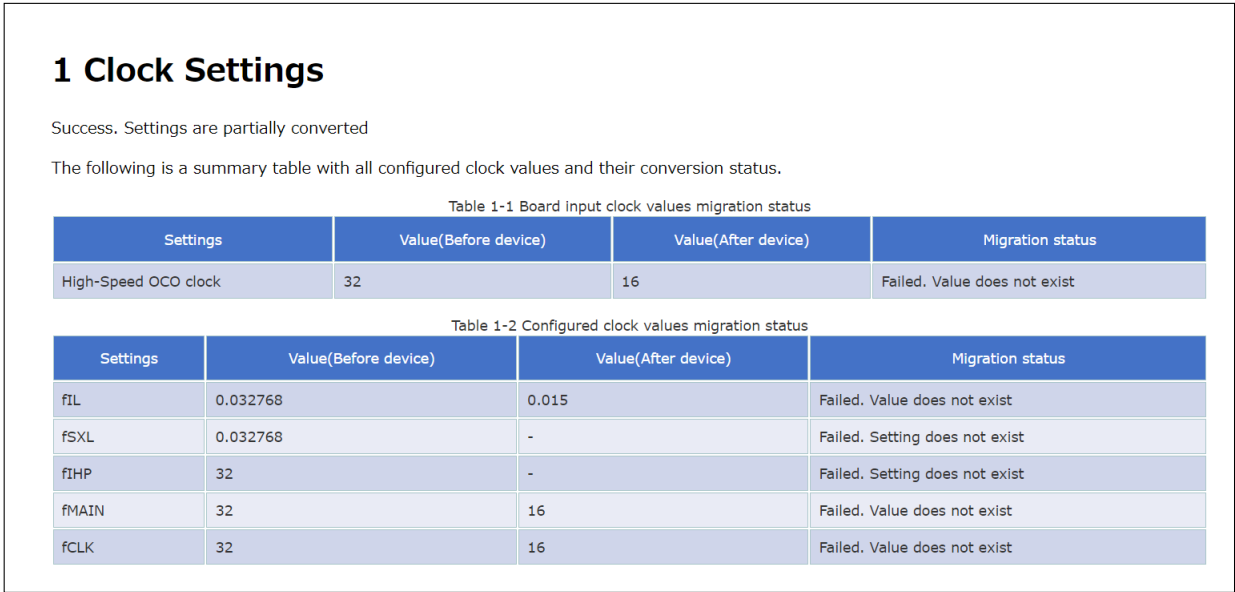
2.10.1 Clock Settings

Configure the operating frequency and operating voltage according to the destination RL78.

The procedure of the check and change is explained using the following case as an example.

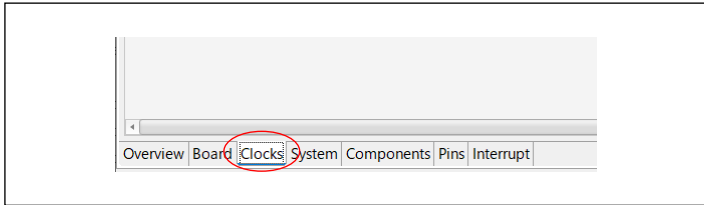
- The source RL78: fCLK = 32MHz
- The destination RL78: fCLK=16MHz

Figure 2-53 Example of Migration report: Clock Settings (1/2)



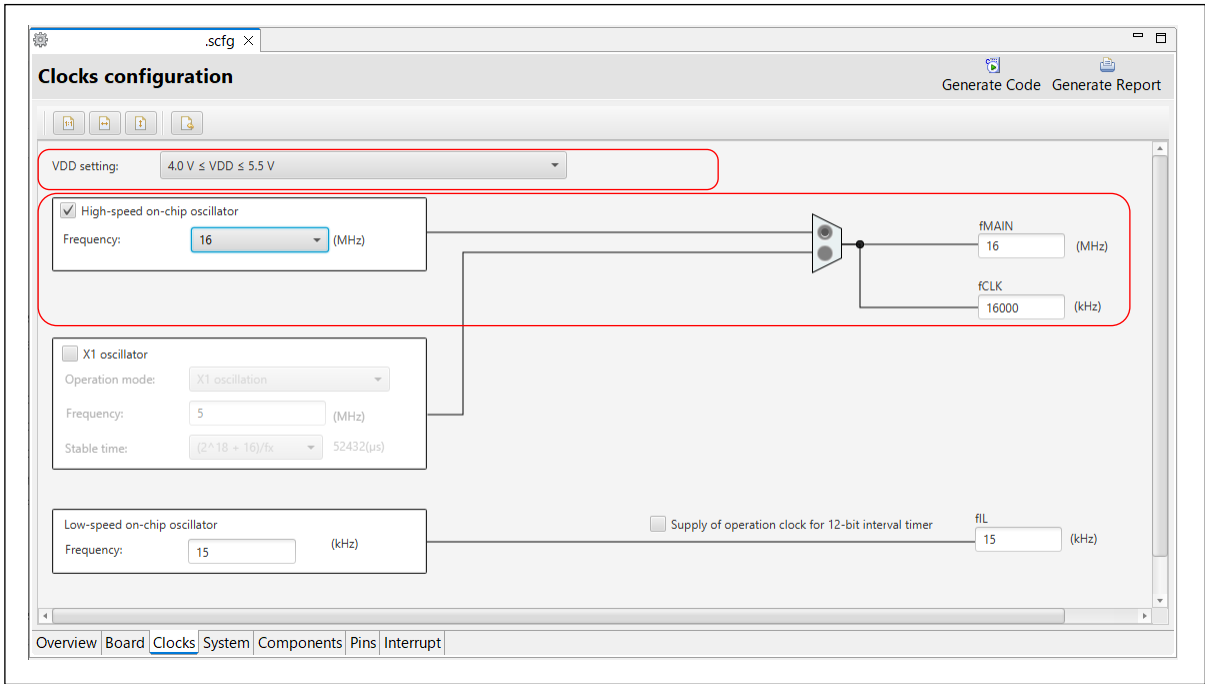
On the Smart Configurator (SC), click the [Clocks] tab.

Figure 2-54 Selection of [Clocks] tab



Set the operating frequency and operating voltage according to the destination RL78.

Figure 2-55 [Clocks] tab



Note. When designing and implementing an actual circuit, make sure to satisfy the hardware's electrical specifications of the destination RL78.

The settings shown in Table-3 are BSP items. BSP items differ for each RL78. BSP items can be changed on the [Components] tab - [r_bsp], if needed.

Figure 2-56 Example of Migration report: Clock Settings (2/2)

Table 1-3 Configured macro definitions migration status			
Settings	Value(Before device)	Value(After device)	Migration status
BSP_CFG_SUBCLK_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_HOCO_DIVIDE	0	-	Failed. Value does not exist
BSP_CFG_SUBSYSCLK_SOURCE	1	-	Failed. Setting does not exist
BSP_CFG_FCLK_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_MOCO_DIVIDE	0	-	Failed. Setting does not exist
BSP_CFG_MOCO_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_XT1_OSCMODE	0	-	Failed. Setting does not exist
BSP_CFG_HISYSCLK_OPERATION	1	1	Success.

2.10.2 System Settings

Configure the debug settings according to the destination RL78.

Figure 2-57 Example of Migration report: System Settings

5 System Settings

Success. Settings are partially converted

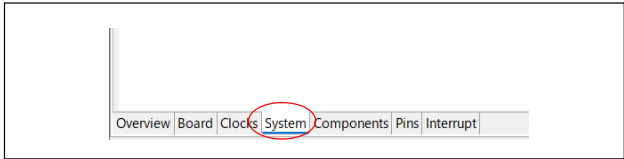
The following is a summary table with all configured system values and their conversion status.

Table 5-1 On-chip debug setting migration status

Settings	Value(Before device)	Value(After device)	Migration status
On-chip debug operation setting	Use emulator	Use emulator	Success.
Emulator setting	E2 Lite	E2 Lite	Success.
Pseudo-RRM/DMM function setting	Used	Used	Success.
Start/Stop function setting	Unused	Unused	Success.
Monitoring point function setting	-	-	Success.
Trace function setting	Used	-	Failed. Setting does not exist
Use security ID	check	check	Success.
Security ID	0x00000000000000000000	0x00000000000000000000	Success.
Security ID authentication failure setting	Do not erase flash memory data	-	Failed. Setting does not exist

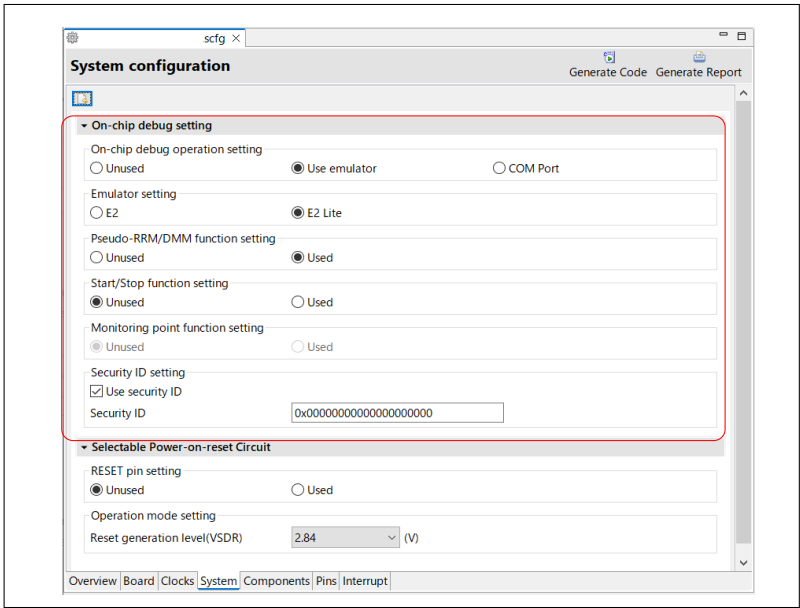
On the Smart Configurator (SC), click the [System] tab.

Figure 2-58 Selection of [System] tab



Set items according to the debugging environment to be used.

Figure 2-59 Selection of [System] tab



2.10.3 Interrupt

Check if the interrupts used in the sample code is migrated. Check the interrupts used in the sample code by the configuration report generated in "2.7 Generate report of SC configuration".

Interrupts not used in the sample code do not need to be migrated. Also, if you change the resource in "2.6 Change resources", make sure that the interrupts after the resource change have been migrated.

Open the configuration report generated in "2.7 Generate report of SC configuration". Then, check the interrupt used in the sample code at the "Interrupt Settings".

Example: INTTM00 is used in the sample code.

Figure 2-60 Example of configuration report: Interrupt Settings

7. Interrupt Settings			
The table below shows used interrupts' information:			
Vector Number	Interrupt	Priority	Bank specify
11	INTTM00	Level 3 (low)	None

Next, open the migration report generated in "2.8 Change the device". Check the migration status of the interrupt. Make sure that the interrupt INTTM00 used in the sample code has been migrated with "state=Used".

Remark. The migration status of the interrupt is indicated for each vector number (default priority number of the interrupt). Therefore, even if the interrupt name is the same between the source RL78 and destination RL78, if the vector number is different, the result is not displayed on the same line.

Figure 2-61 Example of migration report: Interrupt

2 Interrupt					
The following is a summary table with all interrupt assignments and their conversion status					
Table 2-1 Interrupt Migration Status					
Vector Number	Old Interrupt Assignment	New Interrupt Assignment	Status (Before device change)	Status (After device change)	Interrupt Migration Status
11	INTELCL	-	State= Not Use Bank specify=None Priority= Level 3 (low)	-	Failed. Setting does not exist
13	INTIIC00	INTIIC00	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
14	INTTM00	INTTM00	State= Used Bank specify=None Priority= Level 3 (low)	State= Used Bank specify=None Priority= Level 3 (low)	Success.
15	INTSRE0	INTSRE0	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.

2.10.4 Software Component

The r_bsp (BSP: Board Support Package) and peripherals are included in the “Software Component” of the migration report.

Check and configure peripheral components first, next check the r_bsp. Because code generation is required if changing the r_bsp.

Figure 2-62 Example of migration report: Software Component

4 Software Component

The following is a summary table with all configurations and their conversion status.

Table 4-1 Components migration status

	Configurations	Migration status
(3)	r_bsp	Success.Settings are partially converted
(2)	Config_COMP0	Success.Settings are partially converted
(1)	Config_LVD0	Fails.Component is not supported in this device
	Config_TAU0_1	Success.Settings are fully converted
(2)	Config_WDT	Success.Settings are partially converted

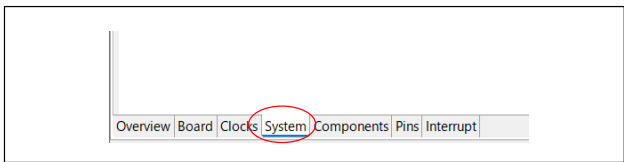
(1) Migration status: Fails

The sample code cannot be ported because the destination RL78 does not support the LVD (Voltage Detector). Check if the destination RL78 has the similar functionality.

For example, if the destination RL78 is the RL78/G15, the Selectable Power-on-reset Circuit (SPOR) can detect the voltage instead of the LVD.

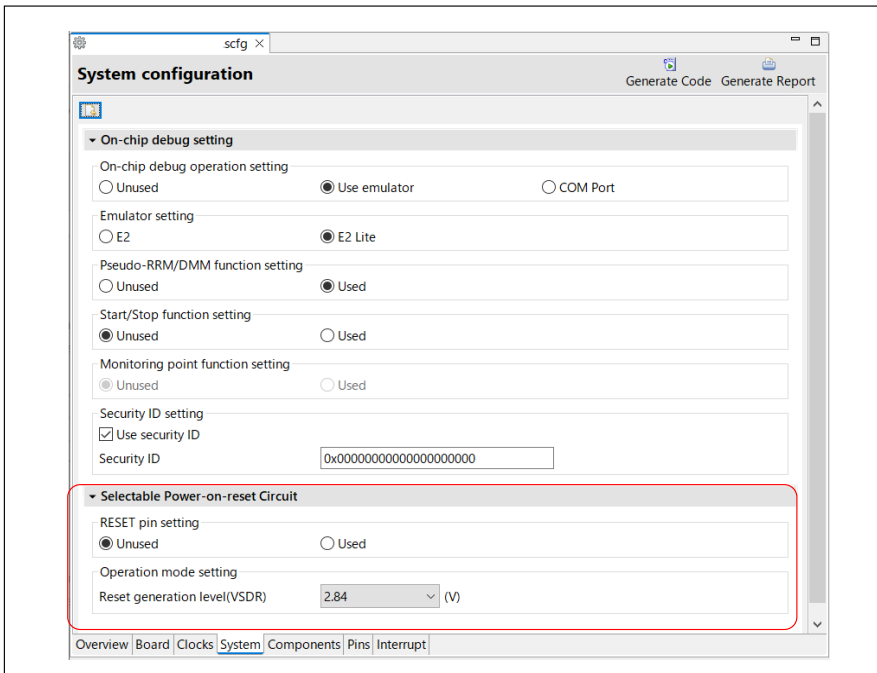
On the Smart Configurator (SC), click the [System] tab.

Figure 2-63 Selection of [System] tab



Specify the reset generation level according to the destination RL78.

Figure 2-64 [System] tab (for RL78/G15)



Note. When designing and implementing an actual circuit, make sure to satisfy the hardware's electrical specifications of the destination RL78.

(2) Migration status: Partially converted

Check the details of applicable components. Check whether the settings that could not be migrated affect the operation of the sample code. If it can be handled by changing the setting value, etc., change the setting value on the setting screen of the applicable component on SC. If it affects the operation of the sample code, it is not possible to port the sample code. Please develop a new program for the destination RL78.

Figure 2-65 Example of a component with some settings migrated: Config_COMP0

4.2 Config_COMP0

Success. Settings are partially converted

Table 4-2-1 Setting migration status

Settings	Value(Before device)	Value(After device)	Migration status
Speed setting	Low speed	Low speed	Success.
Reference voltage	IVREF0	IVREF0	Success.
Voltage range	0 ~ VDD-1.4V	-	Failed. Setting does not exist
Edge setting	Rising edge	Rising edge	Success.
Enable digital filter	Uncheck	Uncheck	Success.
Sampling clock	fCLK	fCLK	Success.
Enable output (VCOUT0)	Uncheck	Uncheck	Success.
Output polarity	Normal	Normal	Success.
Use comparator 0 interrupt (INTCMP0)	Check	Check	Success.
Priority	Level 3 (low)	Level 3 (low)	Success.

Figure 2-66 Example of a component with some settings migrated: Config_WDT

4.4 Config_WDT

Success. Settings are partially converted

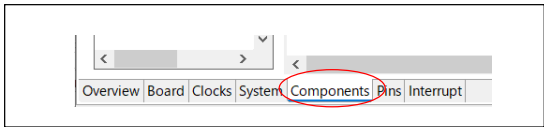
Table 4-4-1 Setting migration status

Settings	Value(Before device)	Value(After device)	Migration status
Operation in HALT/STOP mode setting	Enabled	Enabled	Success.
Overflow time	2 ¹⁷ /fL	(2 ¹⁶ -1)/fL	Failed. Value does not exist
Window open period	100	-	Failed. Setting does not exist
Enable interval interrupt when 75% of overflow time + 3/(4 × fL) (INTWDTI)	Check	Check	Success.
Priority	Level 3 (low)	Level 3 (low)	Success.

The following describes how to display the setting screen for each component.

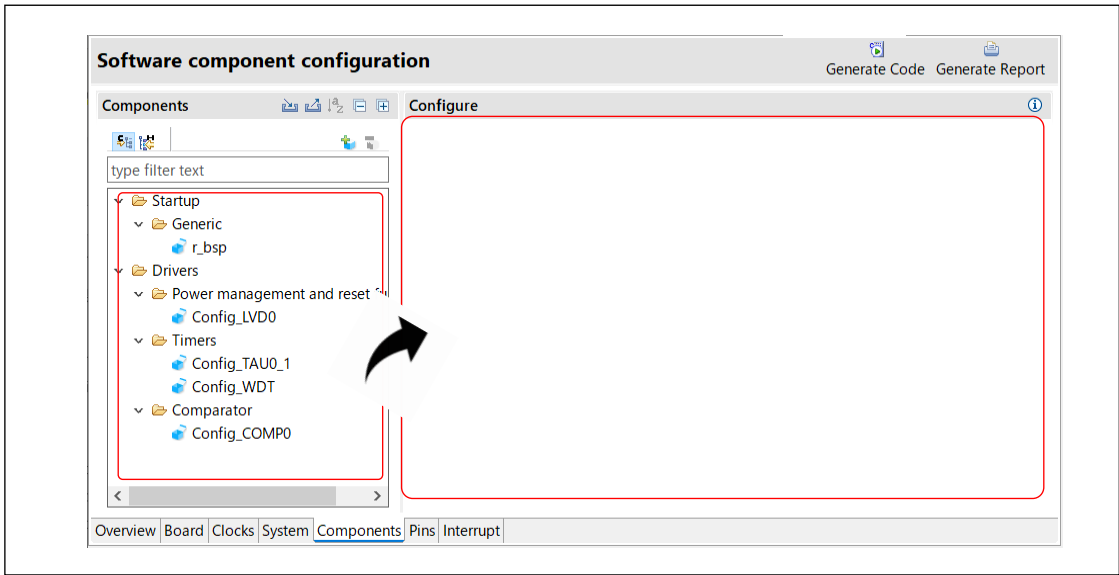
On the Smart Configurator (SC), click the [Components] tab.

Figure 2-67 Selection of [Components] tab



The tree on the left shows the components (peripherals) used. Click a component in the tree to display the settings screen on the left.

Figure 2-68 Display Component settings screen

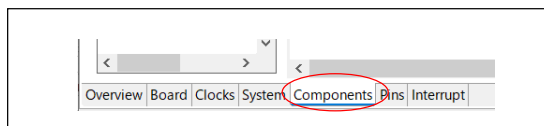


(3) Update BSP

Update the Board Support Package (BSP) used in SC to the latest version.

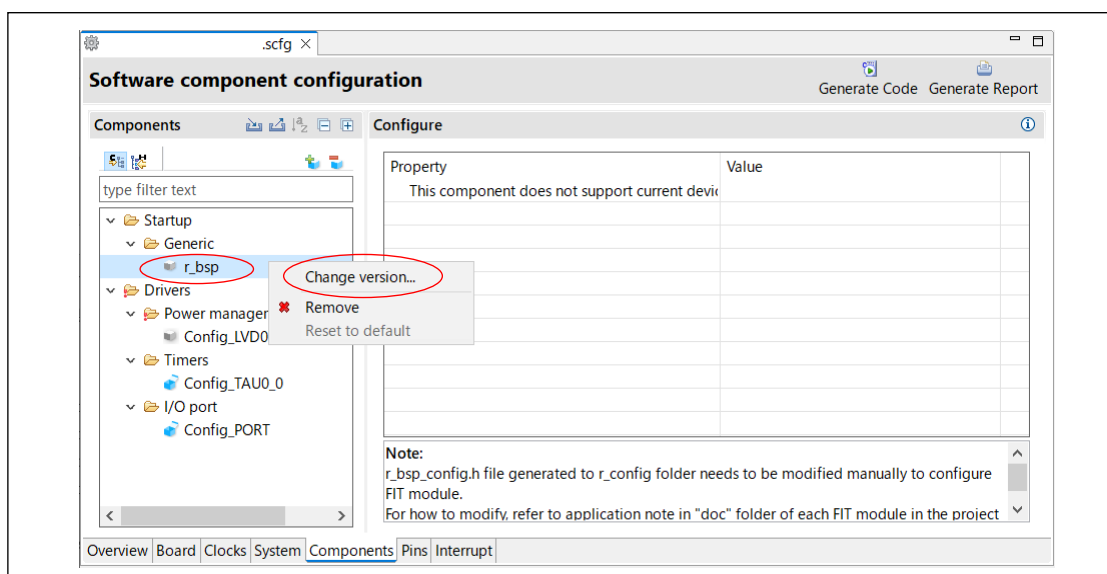
On the Smart Configurator (SC), click the [Components] tab.

Figure 2-69 Selection of [Components] tab



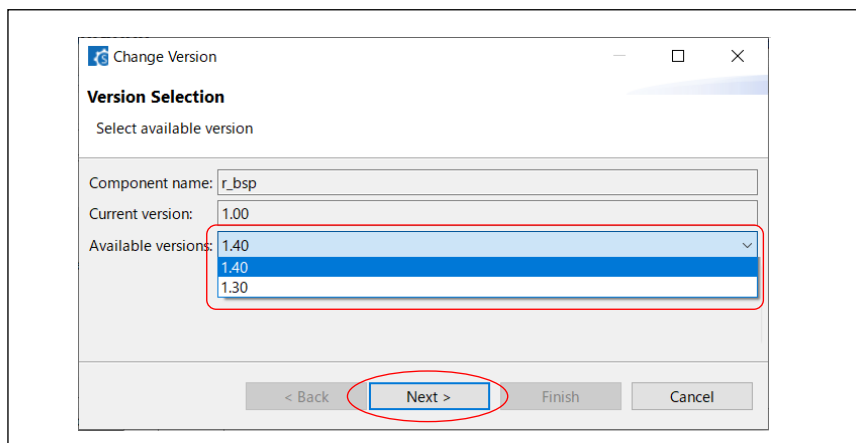
Right-click the “r_bsp” and click the [Change version] menu.

Figure 2-70 Change BSP version



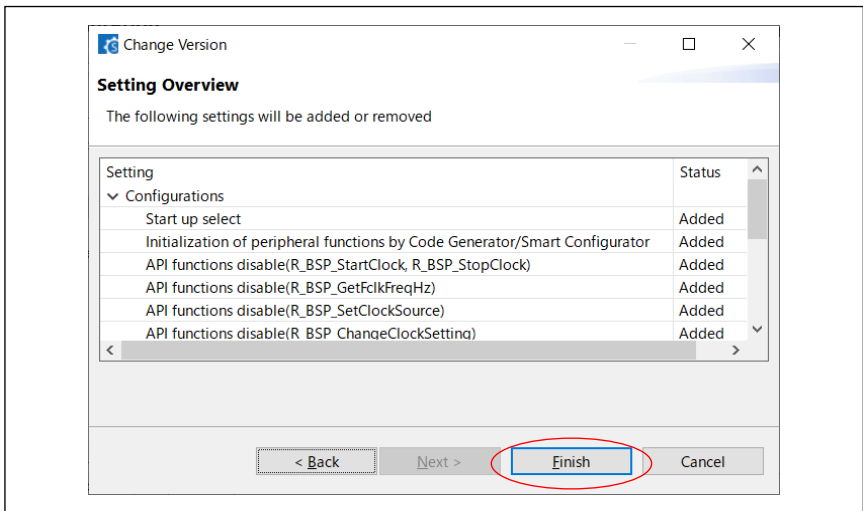
On the [Change version] dialog, specify the latest version listed in the dropdown list at the [Available versions]. Then click the [Next] button.

Figure 2-71 [Change Version] dialog: Version Selection



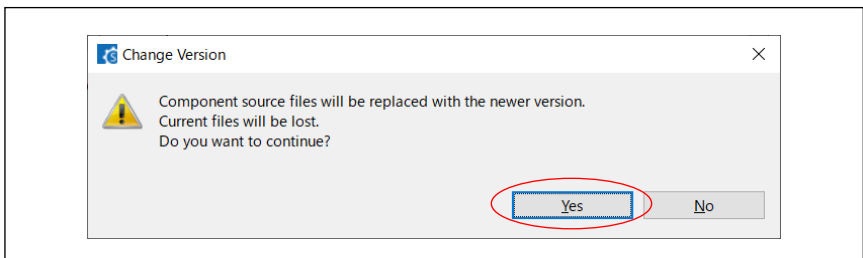
Click the [Finish] button.

Figure 2-72 [Change Version] dialog: Setting Overview



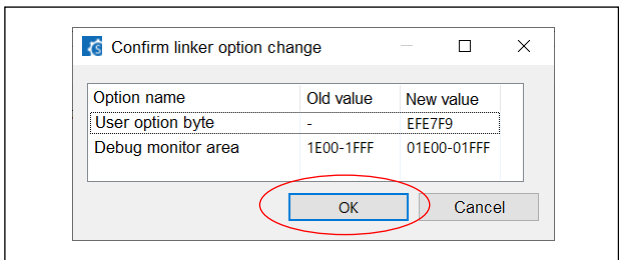
On the [Change Version] dialog, click the [Yes] button. The latest BSP files are generated.

Figure 2-73 [Change Version] dialog: Confirmation



Remark. In the case of CS+ for CC, some CC-RL options are set according to the destination RL78. The dialog appears to show the changed options. In the case of e² studio also, some CC-RL options are set according to the destination RL78. However, the dialog does not appear. In the case of IAR EW for Renesas RL78, the dialog does not appear, and the settings are reflected in BSP files.

Figure 2-74 [Change Version] dialog (CS+ for CC)



2.10.5 Pins

Regardless of whether the result of the migration report is “Success” or “Failed”, check the setting status of pins on the [Pins] – [Pin Number] of SC for all pins to be used.

Figure 2-75 Example of Migration report: Pins

3 Pins			
The following is a summary table with all pin assignments and their conversion status.			
Table 3-1 Pin Migration Status			
Function	Pin Number (Before)	Pin Number (After)	Status
TOOL0	5	1	Success
INTP0	9	8	Success
(1) P50	33	-	Discarded. Function is not available
(2) TxD0	44	6	Success
(2) TI00	62	6	Success

(1) Migration status: Discarded

Figure 2-75 shows that the sample code was not migrated because there is no P50 in the destination RL78. Use another pin in the destination RL78.

The procedure of the change procedure is explained using the following case as an example.

The source RL78: P50

The destination RL78: P04

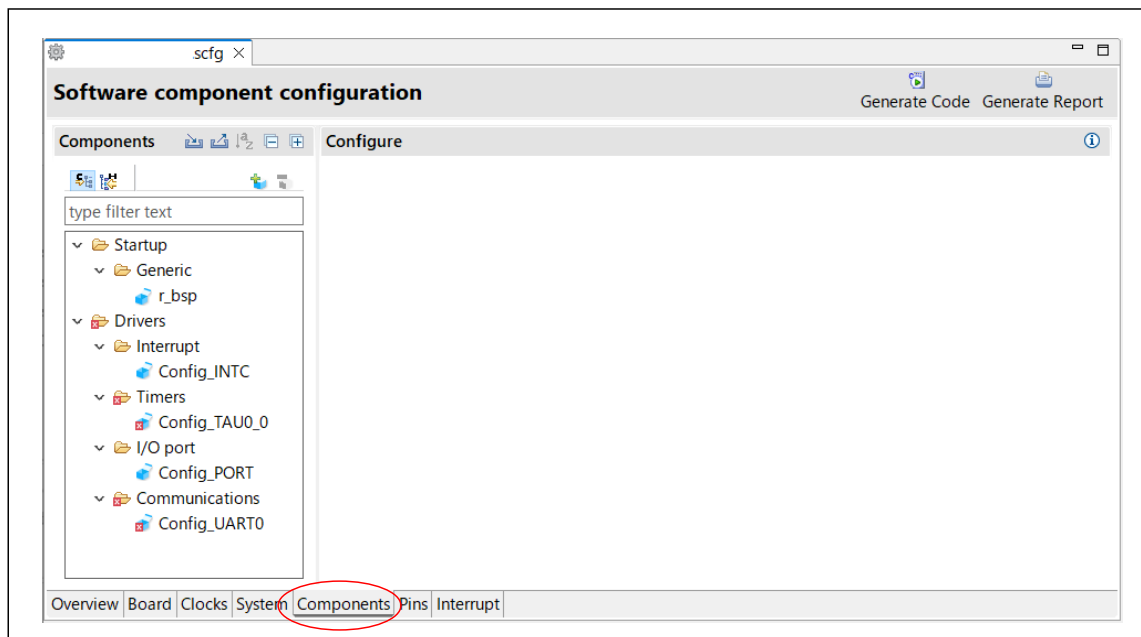
Open the configuration report generated in “2.7 Generate report of SC configuration”. Then, check the port setting used in the sample code at the “Config_PORT”. P50 is used as the output port with an initial value of 1.

Figure 2-76 Example of configuration report: Config_PORT

5.4. Config_PORT	
Component: Ports	
Use status: Used	
Version: 1.3.0	
P50:	
Setting name	Value
Use status	Out
Pull-up	Uncheck
Input buffer OFF	Uncheck
N-ch	Uncheck
Output 1	Check
Output current value	Hi-Z

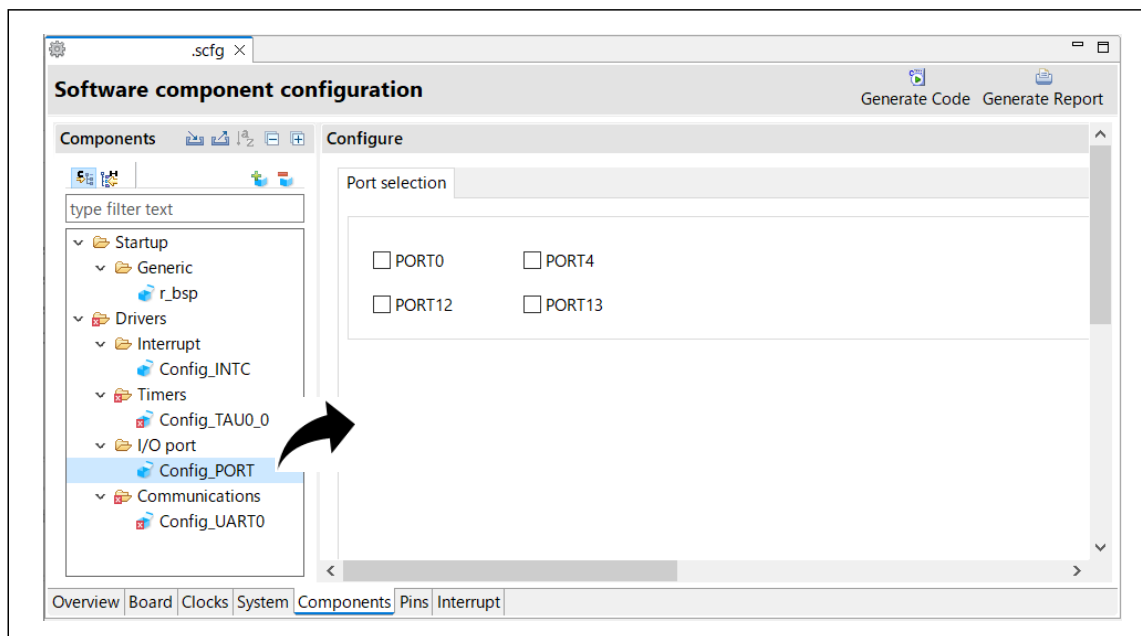
Open the [Components] tab of SC.

Figure 2-77 [Components] tab



Click the [Config_PORT].

Figure 2-78 [PORT] component



Place a check mark in the [PORT0]. Then, place check marks in the [Out] and the [Output 1].

Figure 2-79 Configure port (1/2)

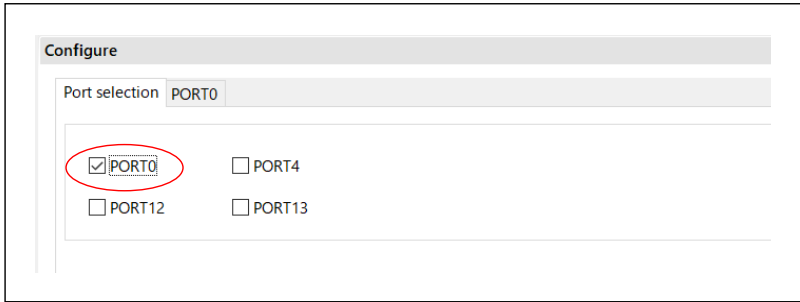
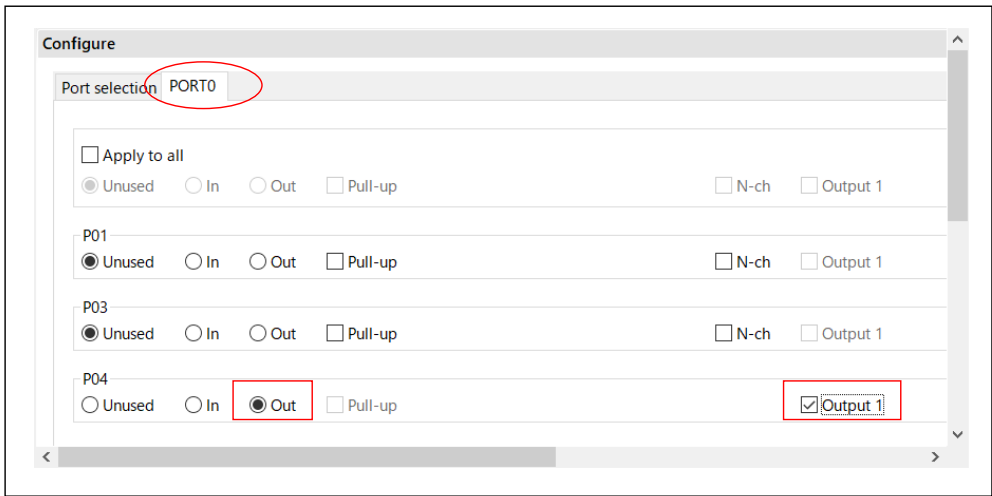
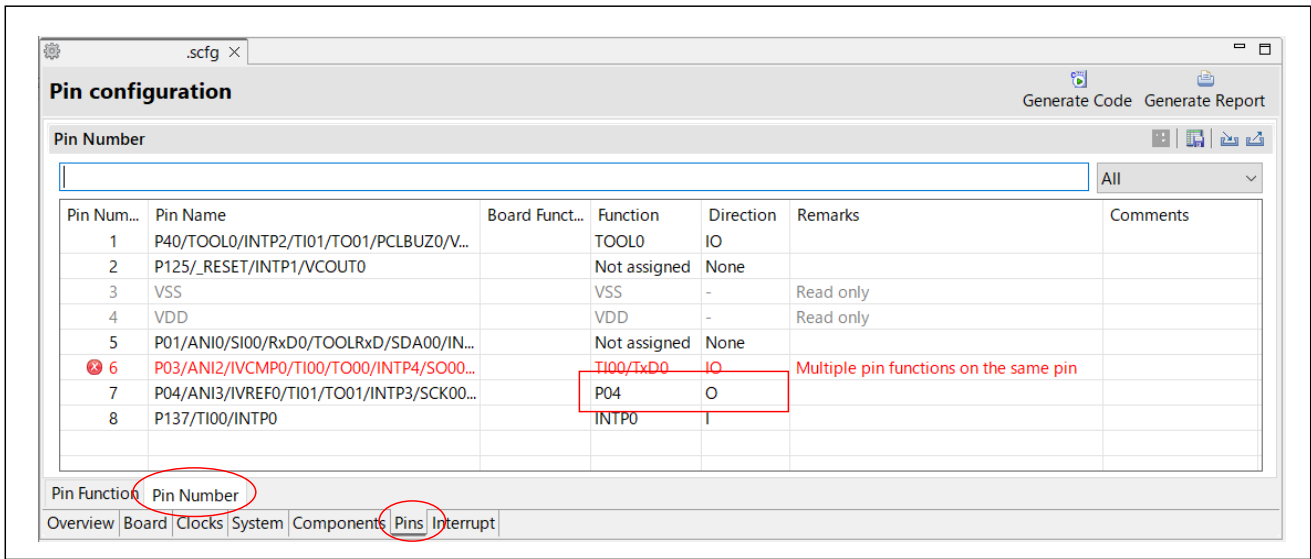


Figure 2-80 Configure port (2/2)



On the [Pins] – [Pin Number] tab, make sure that P04 is assigned.

Figure 2-81 [Pins] – [Pin Number] tab



(2) Migration status: Success

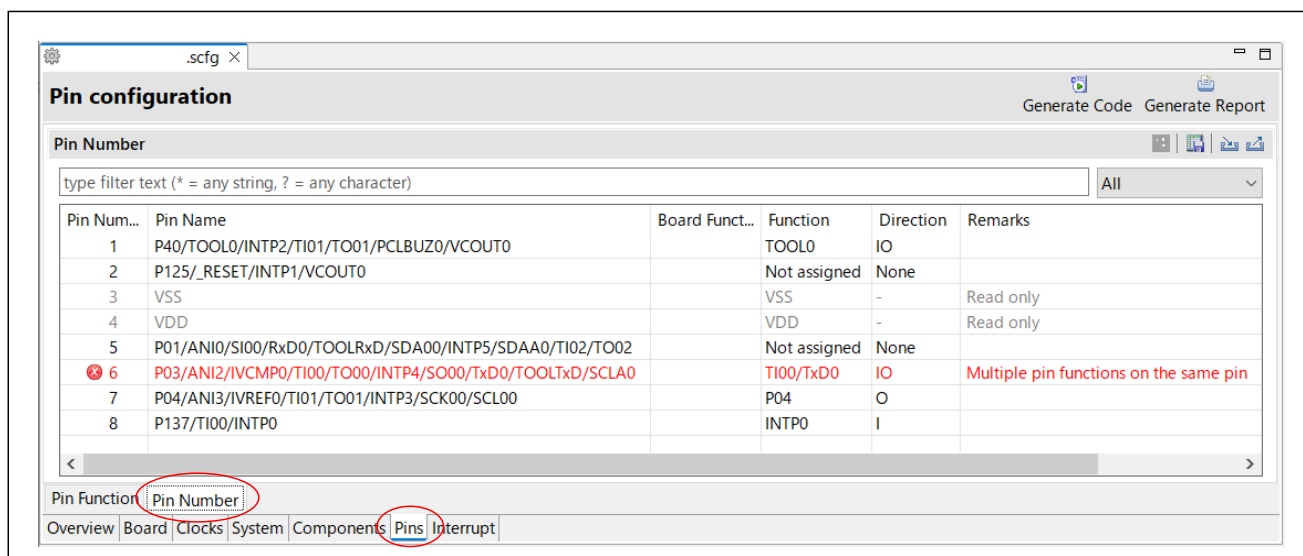
Check whether the pin which was migrated successfully have been assigned to the same pin number (overlap). If the pins overlap, change the assignment to another pin. If it cannot be assigned to another pin, it is not possible to port the sample code. It is necessary to change the specifications of the system or change the destination RL78. If there is no overlap, the following steps in this section are not needed.

Figure 2-82 Example of migration report: Pins (Overlap)

Function	Pin Number (Before)	Pin Number (After)	Status
TOOL0	5	1	Success
INTP0	9	8	Success
P50	33	-	Discarded. Function is not available
TxD0	44	6	Success
TI00	62	6	Success

Click the [Pins] – [Pin Number] tab.

Figure 2-83 [Pins] – [Pin Number] tab: Overlap



Check whether TI00 or TxD0 can be assigned to another pin. Input “TI00” or “TxD0” into the top text box.

Figure 2-84 Pin search (1)

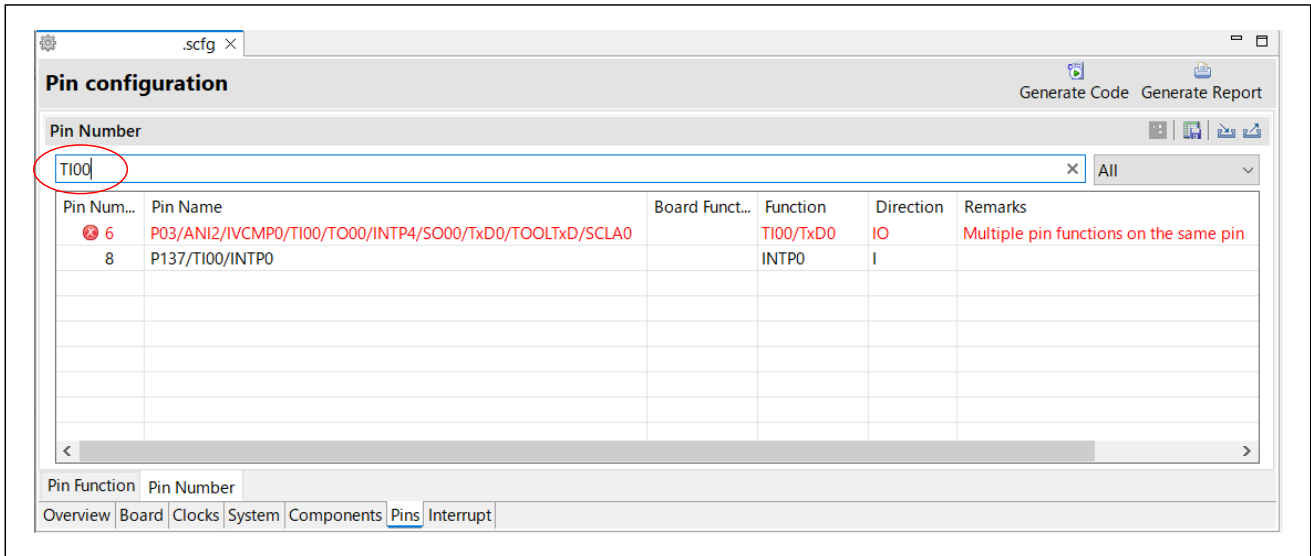
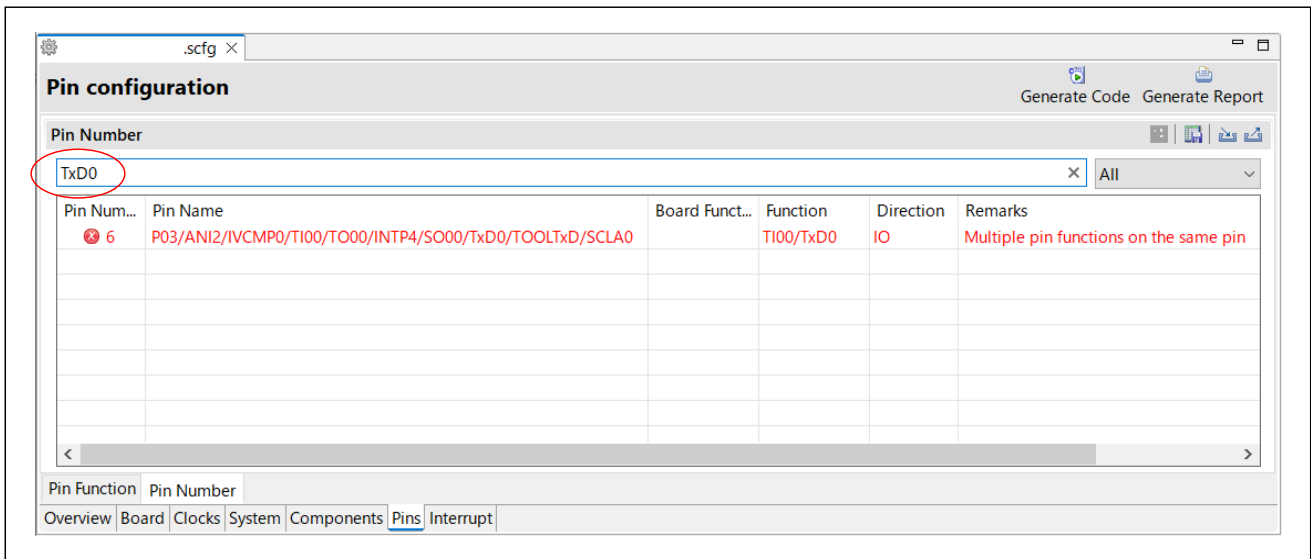


Figure 2-85 Pin search (2)



TI00 can be assigned to the pin number 6 or 8, however the pin number 8 is used as INTP0. TxD0 can be assigned to the pin number 6 only. Because it is not possible to use the TI00, TxD0 and INTP0 at the same time by the current pin assignment, change the pin assignment.

Example 1 of change: Change INTP0 to another channel (INTPx)

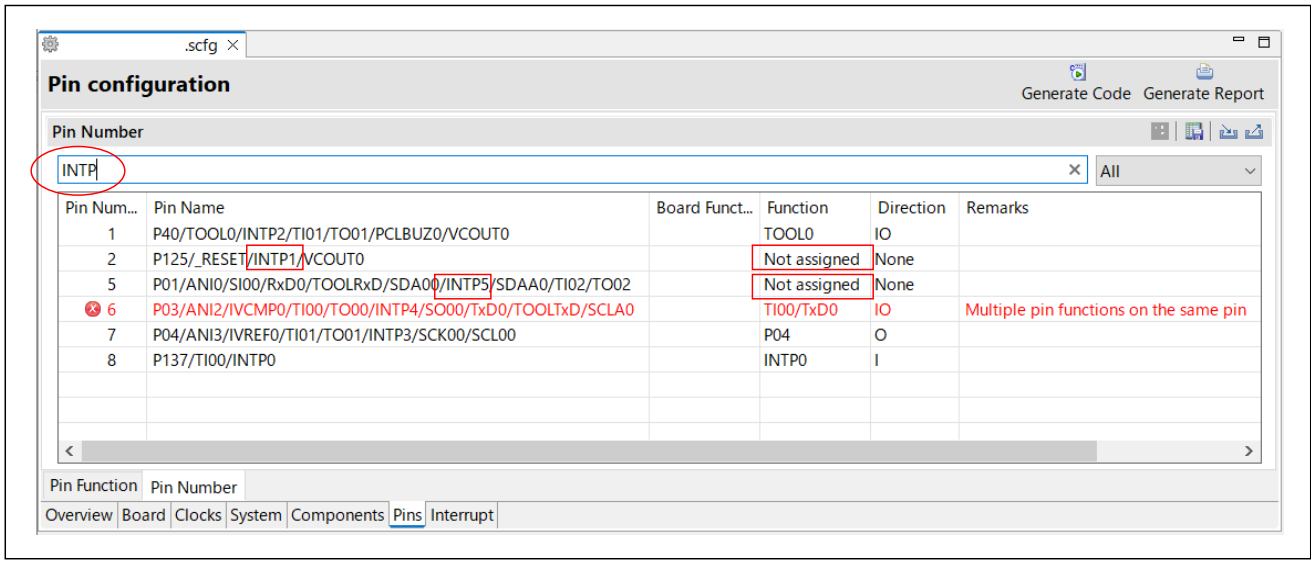
Example 2 of change: Change TI00 to another channel (TIxx)

(a) Example 1 of change

Check whether INTP0 can be changed to another INTPx. Input “INTP” into the top text box of the [Pins] – [Pin Number] tab..

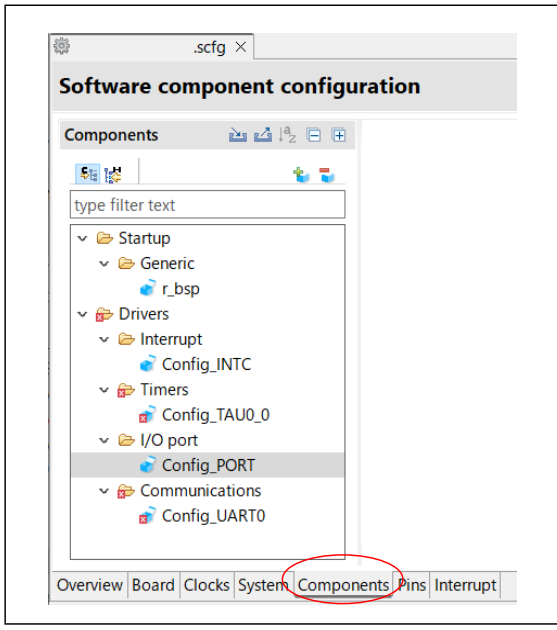
As the search result, the pin number 2 has INTP1 and the pin is free, and the pin number 5 has INTP5 and the pin is free.

Figure 2-86 Pin search (3)



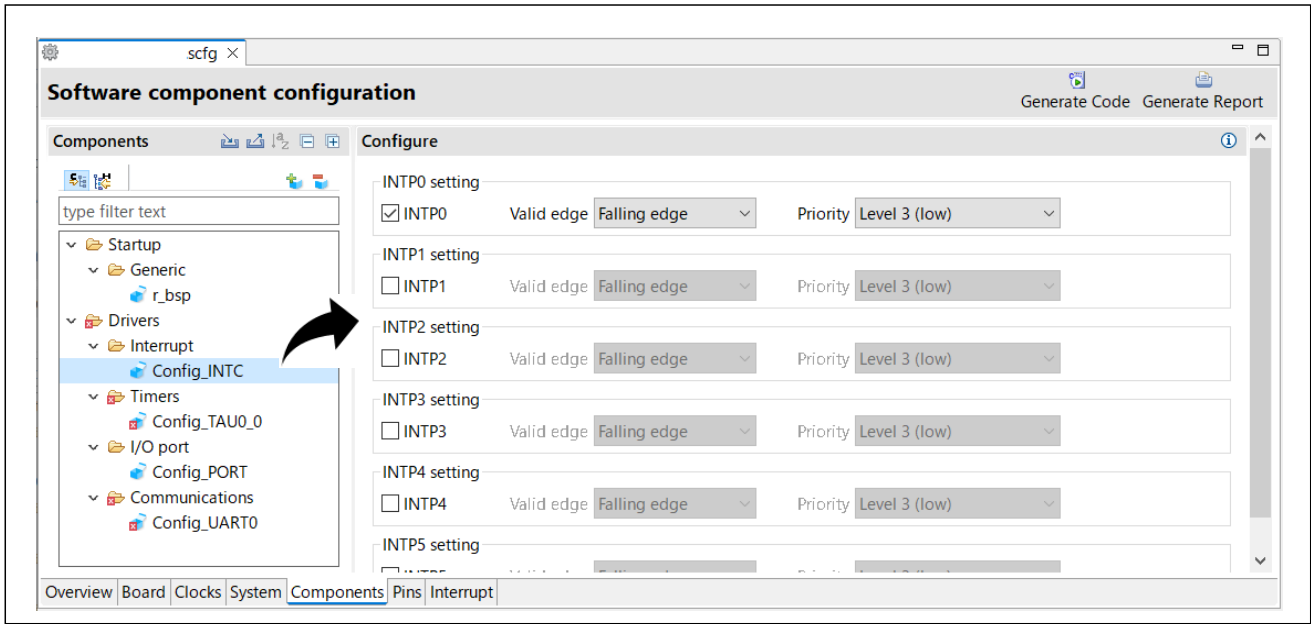
Change the channel to be used from INTP0 to INTP5. Open the [Components] tab.

Figure 2-87 Selection of [Components] tab



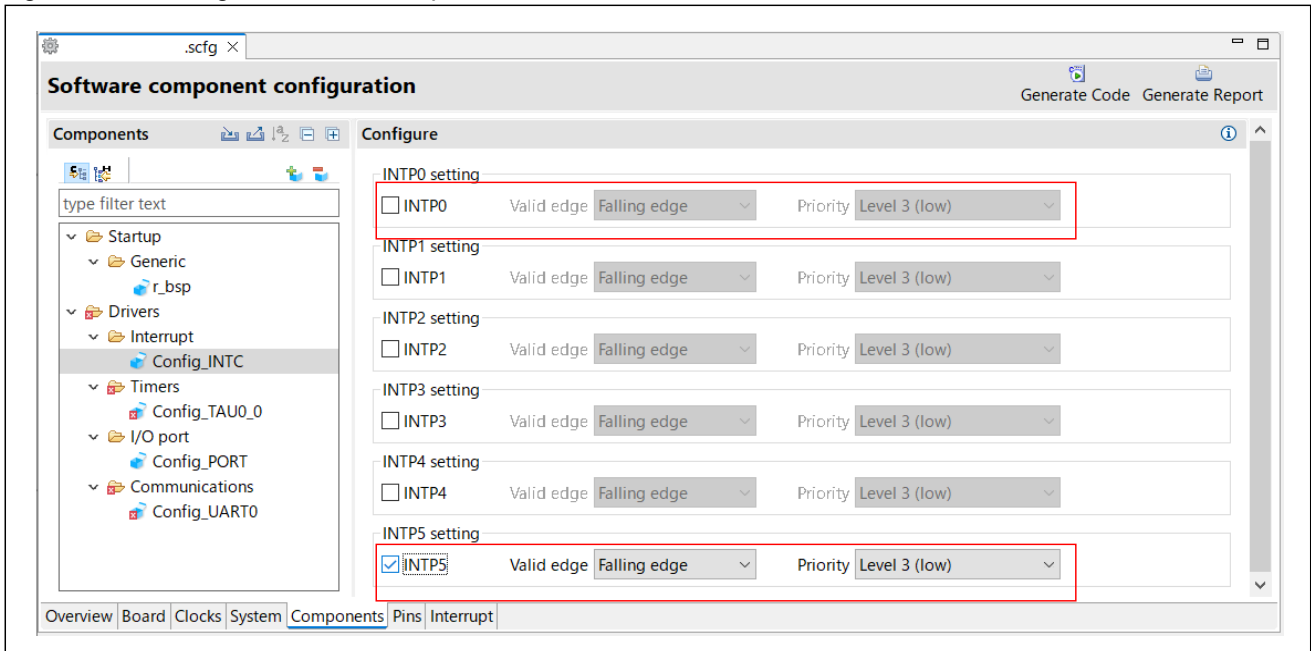
Click the [Config_INTC].

Figure 2-88 [Interrupt controller] component



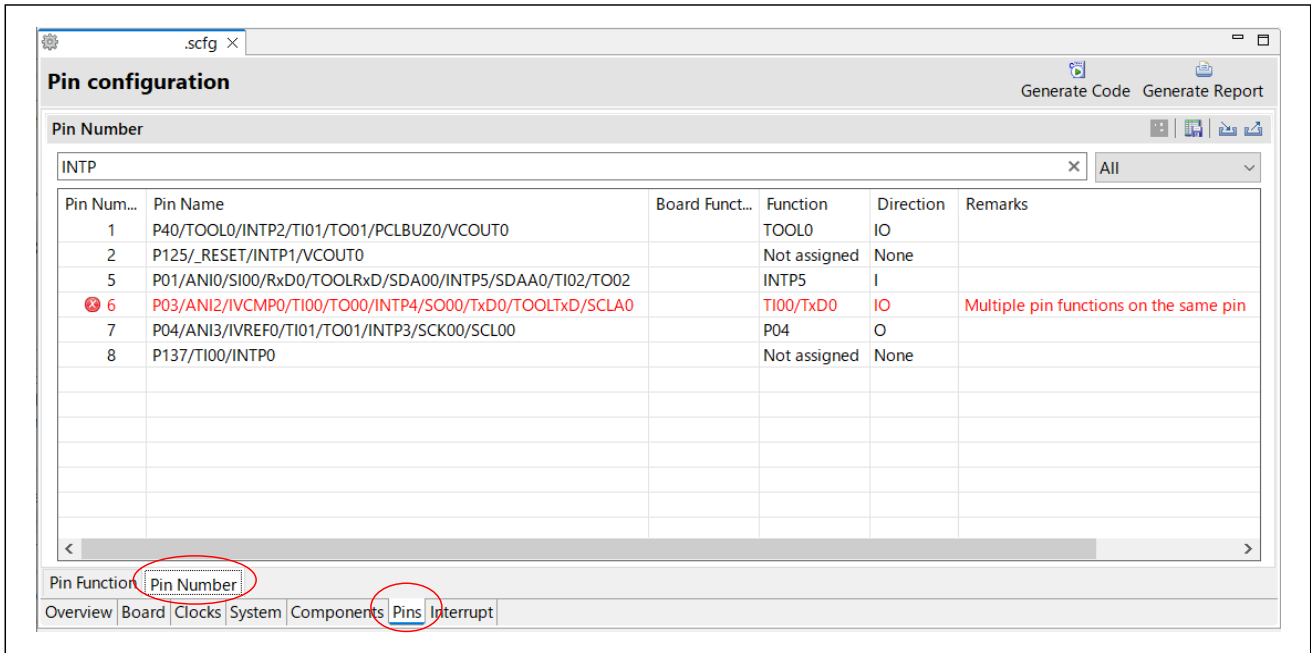
Uncheck the INTP0, place the check mark in the INTP5 and configure the same setting as the INTP0.

Figure 2-89 Change external interrupt



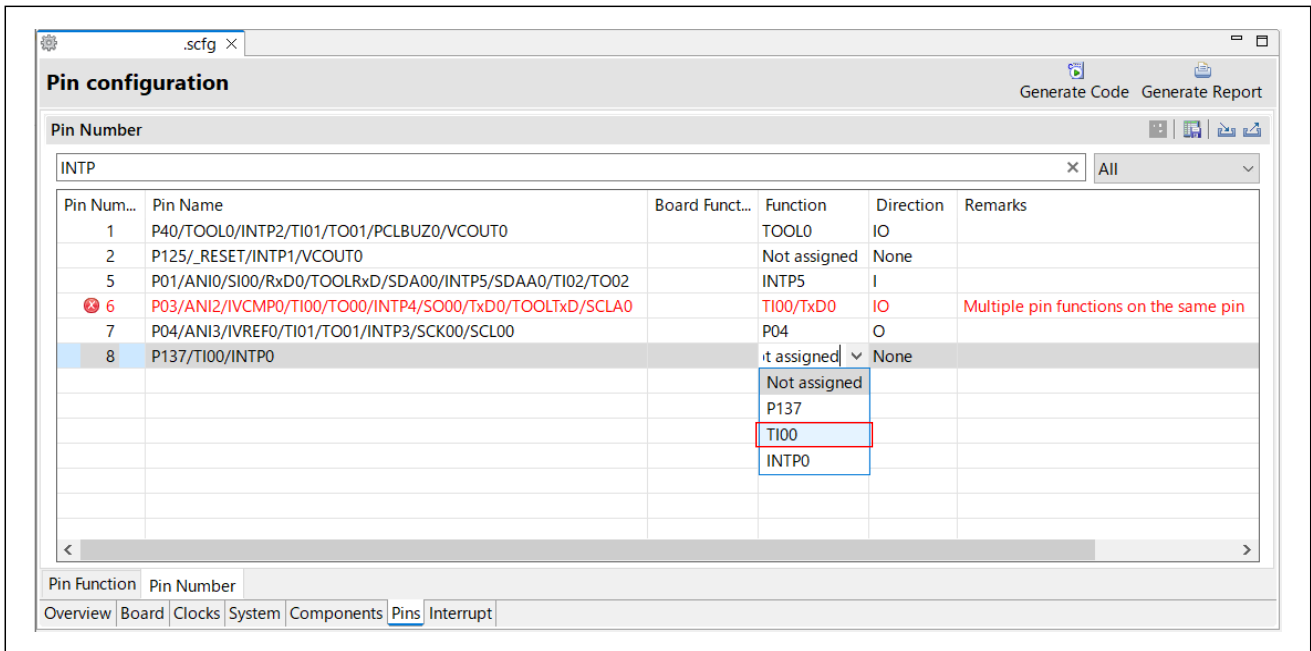
Open the [Pins] – [Pin Number] tab.

Figure 2-90 [Pins] – [Pin Number] tab: Overlap



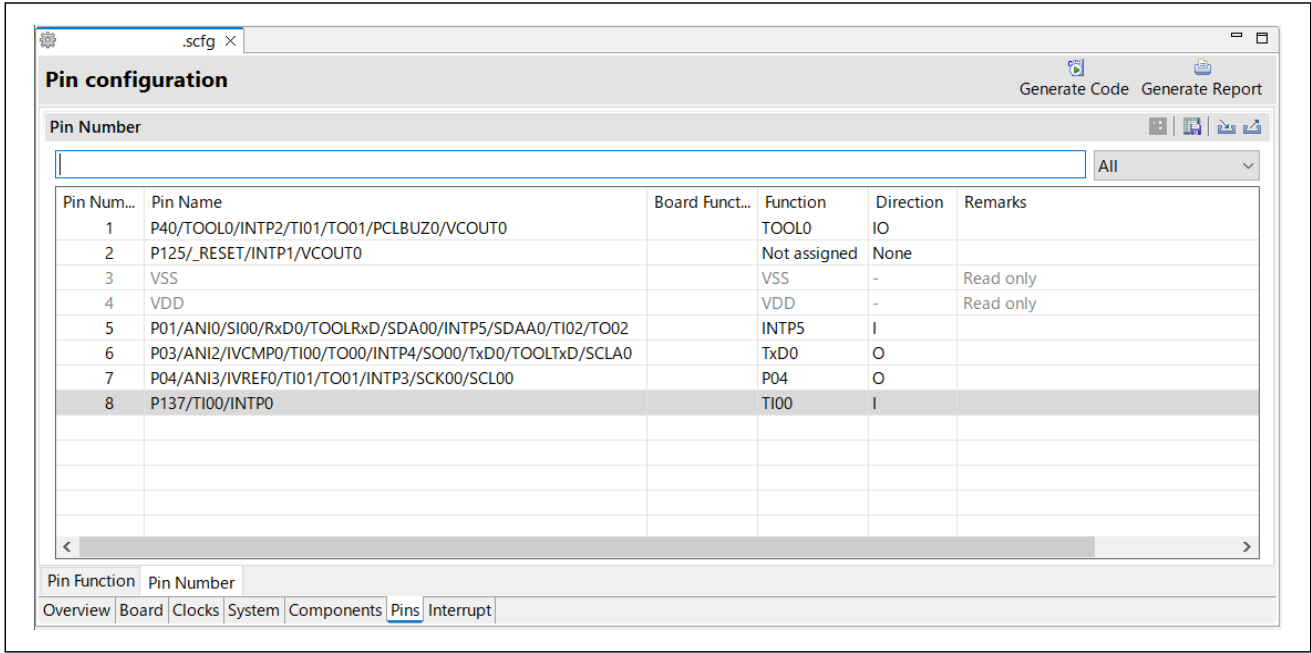
Assign TI00 to the pin number 8 that is free. Specify “TI00” on the [Function] column of the pin number 8.

Figure 2-91 [Pins] – [Pin Number] tab: Change function of pin



Make sure that the function overlap is resolved. (Delete any text in the top text box.)

Figure 2-92 [Pins] – [Pin Number] tab: no overlap

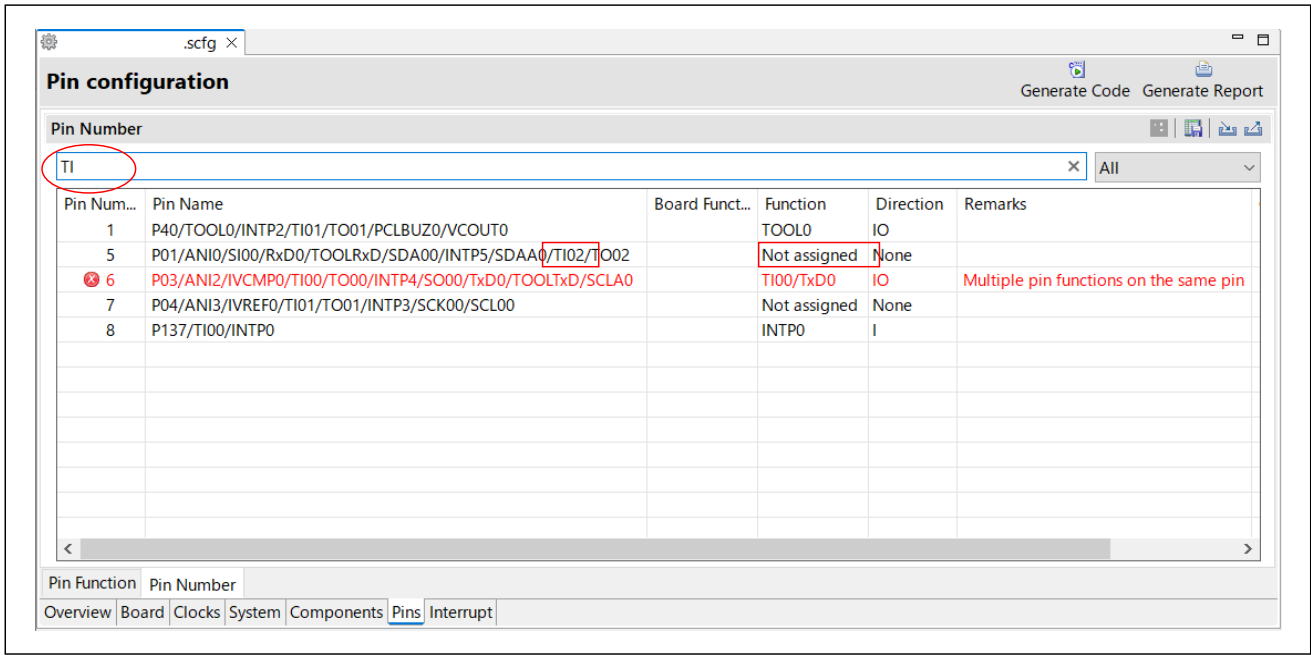


(b) Example 2 of change

Check whether TI00 can be changed to another TIxx. Input “TI” into the top text box of the [Pins] – [Pin Number] tab.

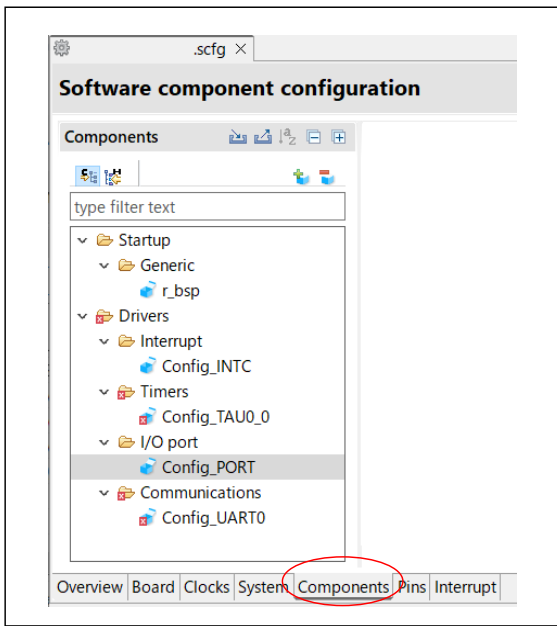
As the search result, the pin number 5 has TI02 and the pin is free.

Figure 2-93 Pin search (4)



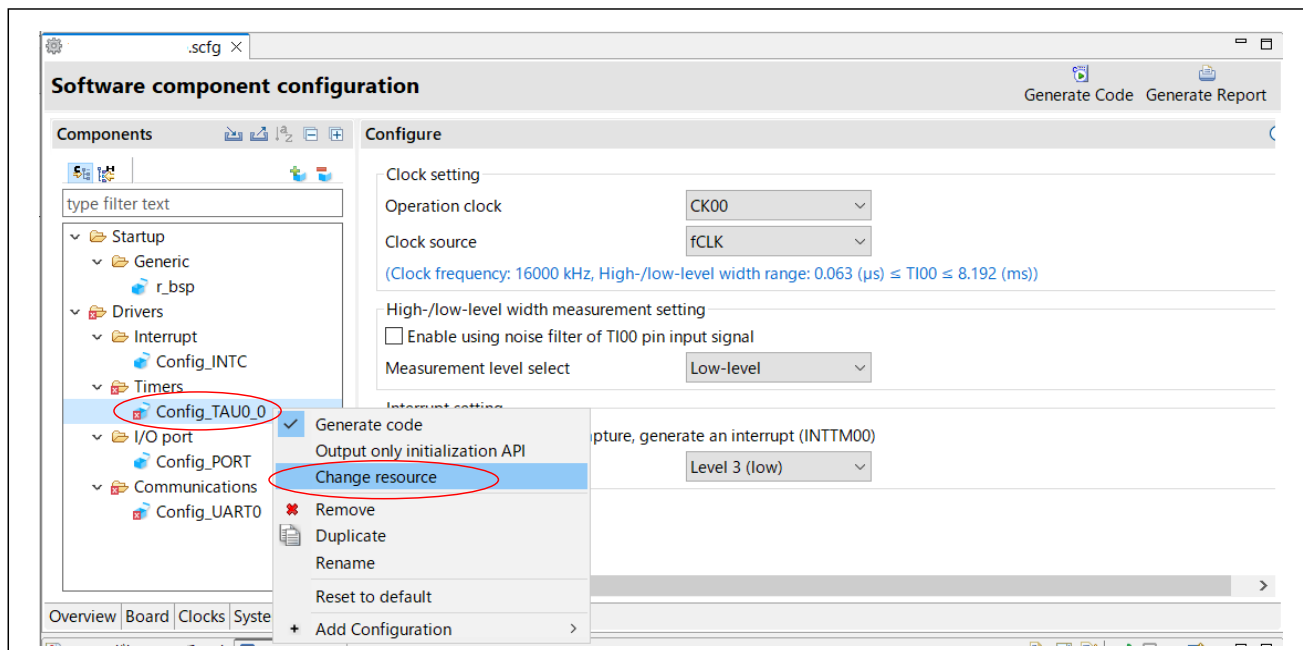
Change the channel to be used from TI00 to TI02. Open the [Components] tab.

Figure 2-94 Selection of [Components] tab



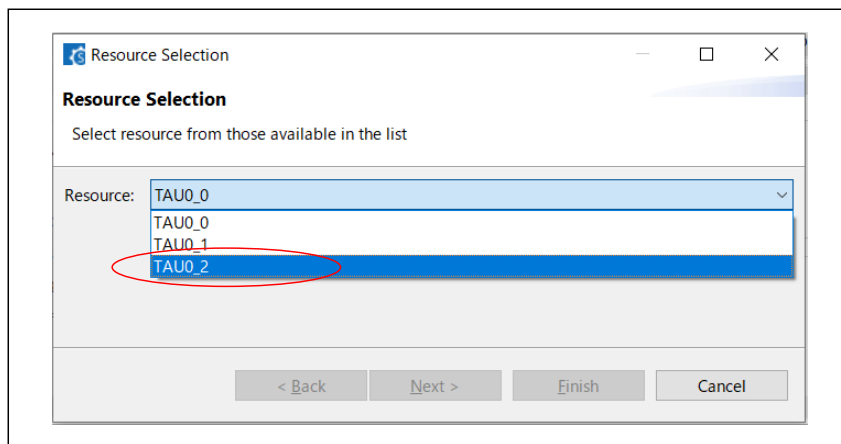
Right click the [Config_TAU0] and click the [Change resource] menu.

Figure 2-95 Change resource



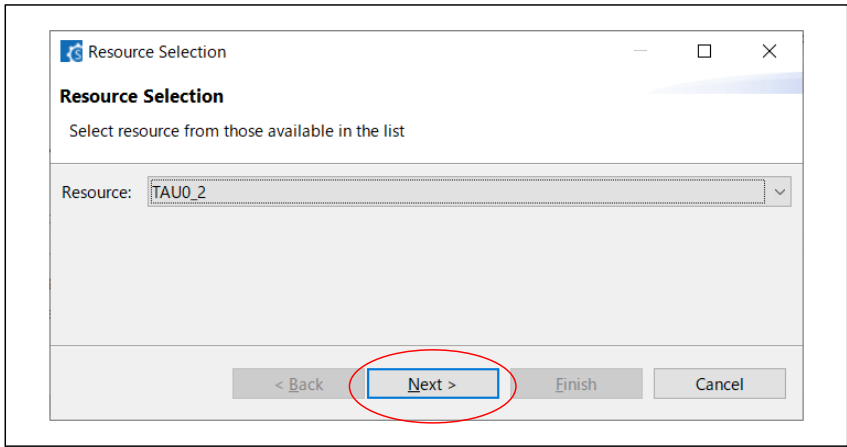
Specify “TAU0_2” in the [Resource:] on the [Resource Selection] dialog.

Figure 2-96 [Resource Selection] dialog (1/3)



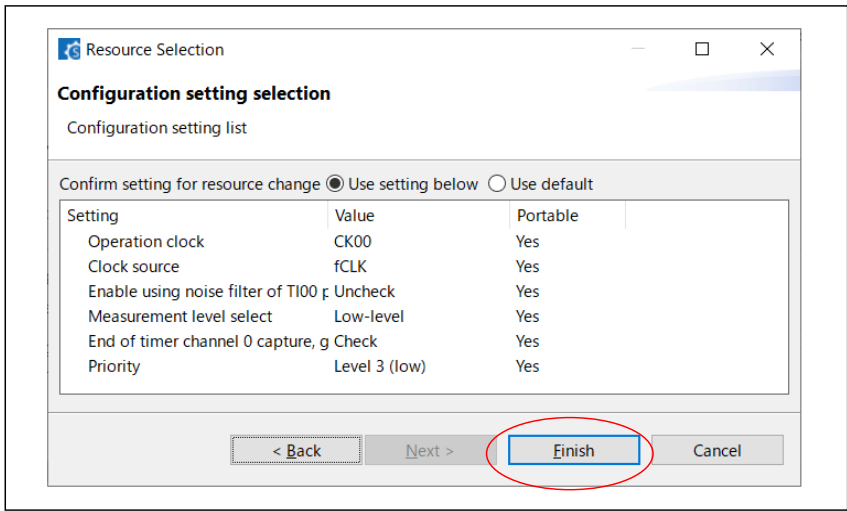
Click the [Next] button.

Figure 2-97 [Resource Selection] dialog (2/3)



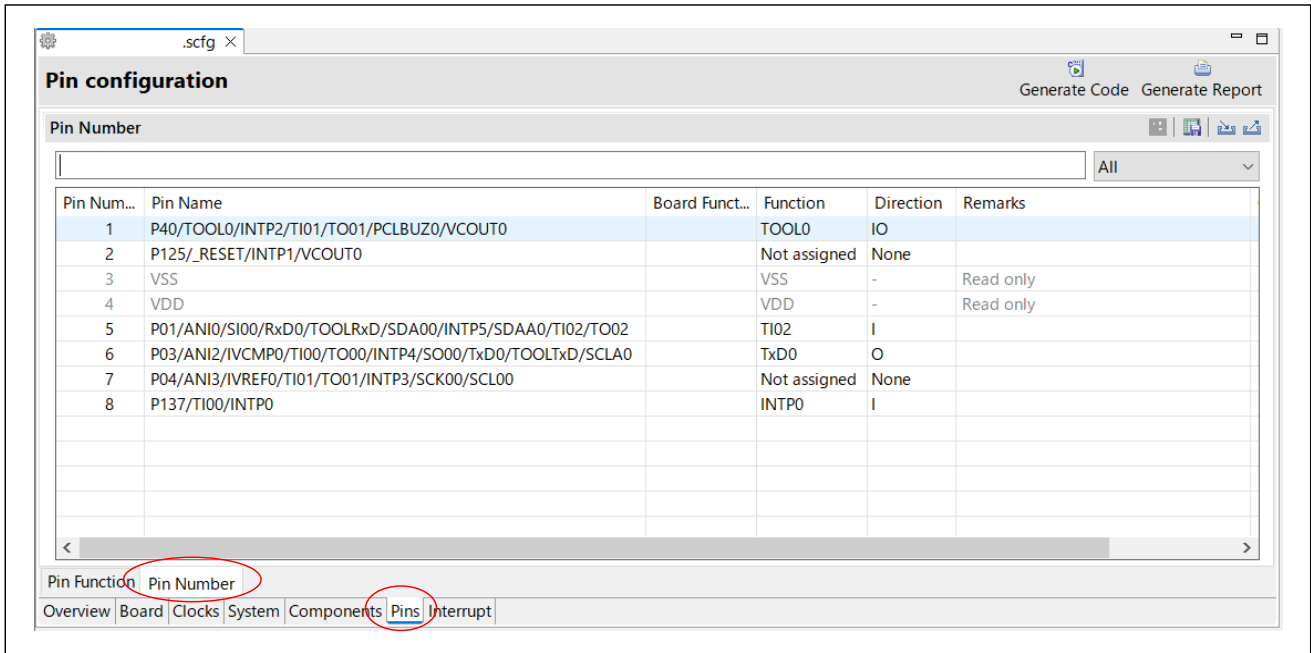
Click the [Finish] button without changing any settings.

Figure 2-98 [Resource Selection] dialog (3/3)



Make sure that the function overlap is resolved. (Delete any text in the top text box.)

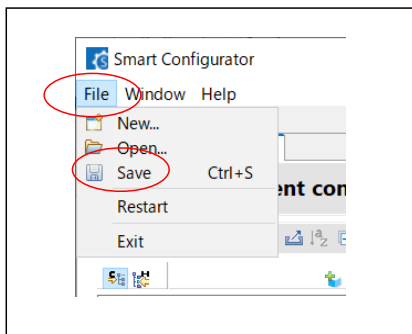
Figure 2-99 [Pins] – [Pin Number] tab: no overlap



2.11 Save the SC settings

Save the settings of Smart Configurator (SC). Click the [File] – [Save] menu on SC. (In the case of e² studio, click the [file] – [Save] menu of e² studio.)

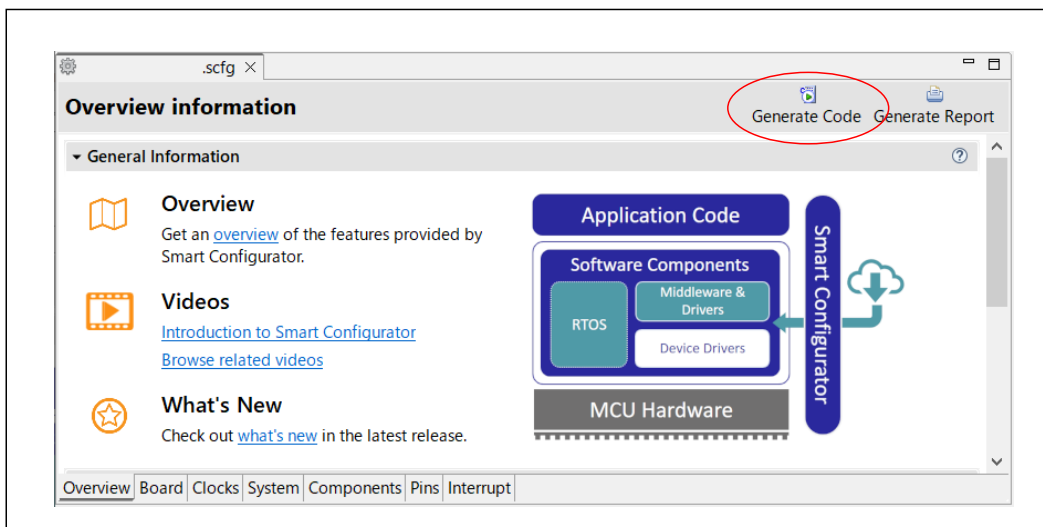
Figure 2-100 Save SC settings



2.12 Code generation

Generate the code on Smart Configurator (SC) after the device change and change settings or resources. Click the [Generate Code] button on SC.

Figure 2-101 Generate Code



2.13 Edit the code

If you find that the SC generate code has been modified in “2.5 Check code changes”, or if any resource is changed in “2.10 Change settings and resources”, it is necessary to edit the generated file.

<The part that needs editing.>

- (1) In “2.5 Check code changes”, if there are parts that have been modified by commenting out etc.
-> In the applicable files of the destination RL78, make the same modifications as the source RL78.
- (2) In “2.10 Change settings and resources”, if the resource of I/O port is changed, the subject to edit is the codes that handle the input value or the output value of the I/O port.
-> In files of the destination RL78, modify the port name in the code.
- (3) In “2.10 Change settings and resources”, if the channel of the peripheral function is changed, the subject to edit is the codes that call the peripheral's start or stop API generated by SC..
-> In files of the destination RL78, modify the API name. However, the body of the API function is replaced with code corresponding to the changed channels in “2.12 Code generation”, therefore, it is not necessary to edit the body of the API functions.

However, the subject to (2) and (3) are only the user code in the user code comment shown in Figure 1-2 in main.c or SC generated files.

Examples of actions for (2) and (3) are shown below.

RL78 Software Porting Guide

Porting sample code generated by Smart Configurator (CS+, e2 studio, IAR)

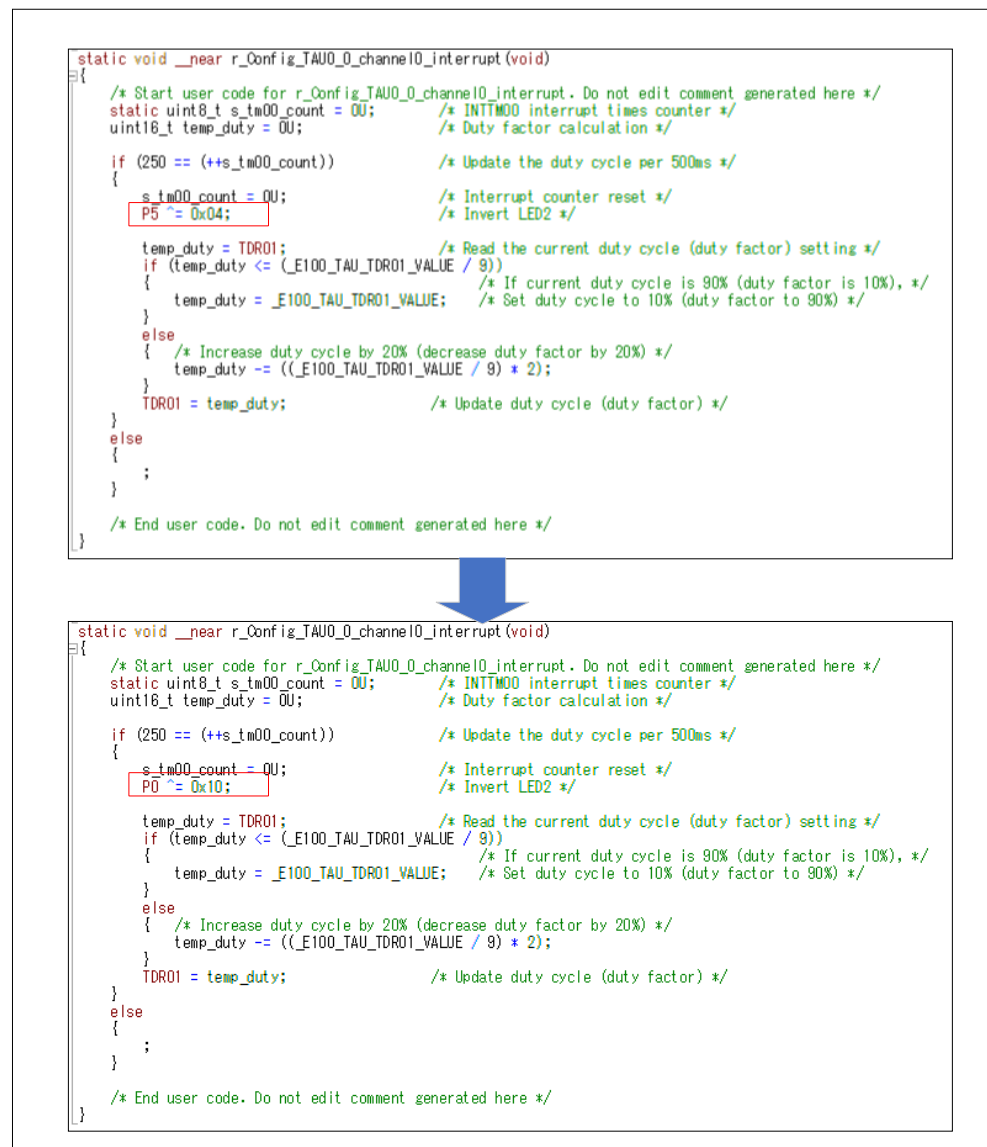
(2) When the I/O port is changed

Example of change: P52 is used in the resource RL78. Changed to use P04 in the destination RL78.

In the project file of the destination RL78, check the location of the codes that access to P52. Search it by the search string as "P5".

From the search results, edit the code for P52 in the user code comments in main.c or SC generated file according to P04.

Figure 2-102 Code modification: Port



After the edit, save the file.

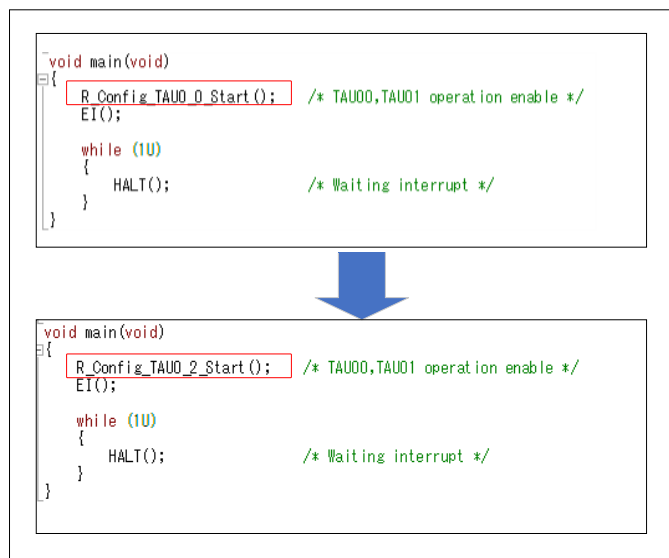
(3) When the channel of peripheral is changed

Example of change: TAU0_0 (channel 0 of TAU0) is used in the resource RL78. Changed to use TAU0_2 (channel 2 of TAU0) in the destination RL78.

In the project file of the destination RL78, check the location of the code that calls the API of TAU0_0. Search it by the search string as "R_Config_TAU0_0". Uncheck the options related to the exact check when searching.

From the search results, in the user code comments in main.c or SC generated file, edit the code that calls the API starting with "R_Config_TAU0_0" to the code starting with "Config_TAU0_2".

Figure 2-103 Code modification: main()



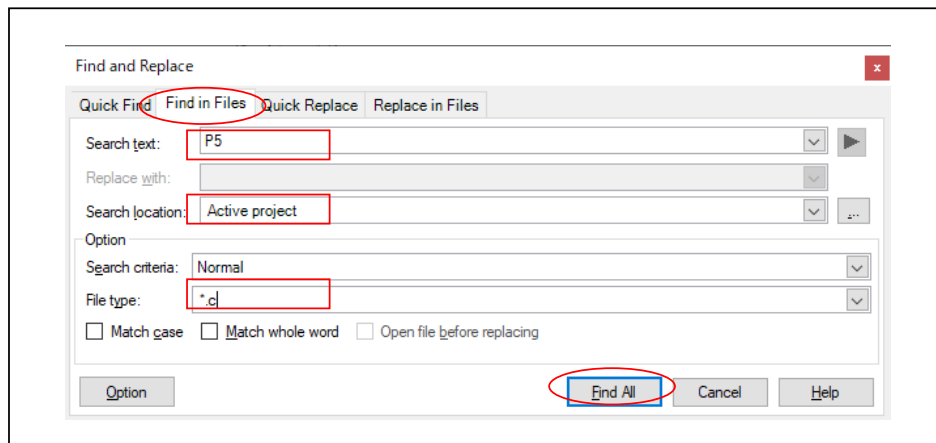
After the edit, save the file.

Reference: How to search on each IDE

- CS+ for CC

Click the [Edit] – [Find] menu of CS+ for CC. On the [Find and Replace] dialog, specify as follows and click the [Find All] button.

Figure 2-104 CS+ for CC [Find and Replace] dialog

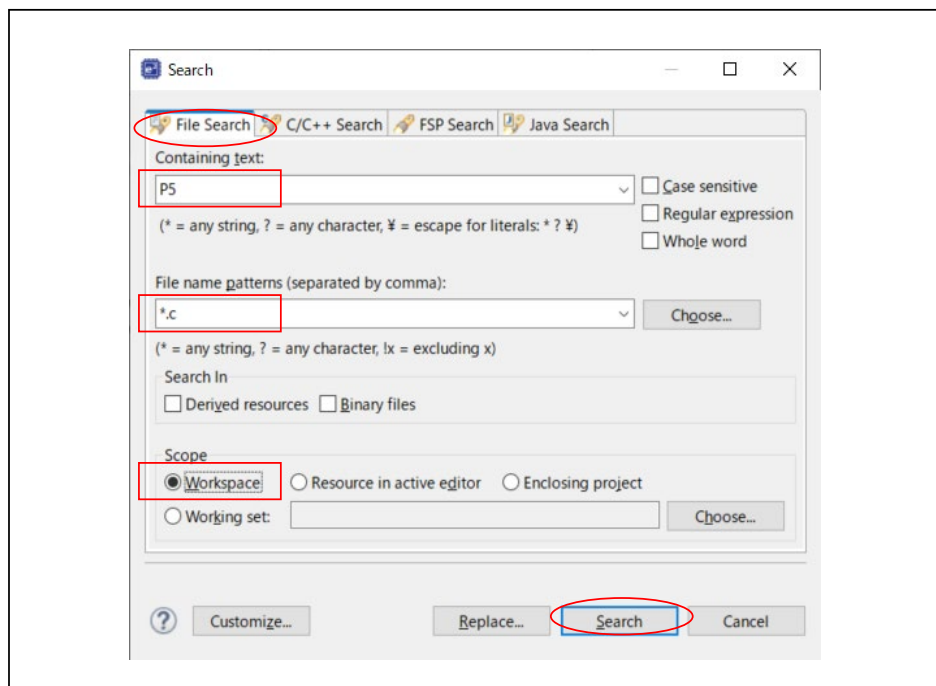


The result is displayed in the [Output] tab.

- e² studio

Click the [Search] - [Search] menu of e² studio. On the [Search] dialog, specify as follows and click the [Search] button.

Figure 2-105 e² studio [Search] dialog

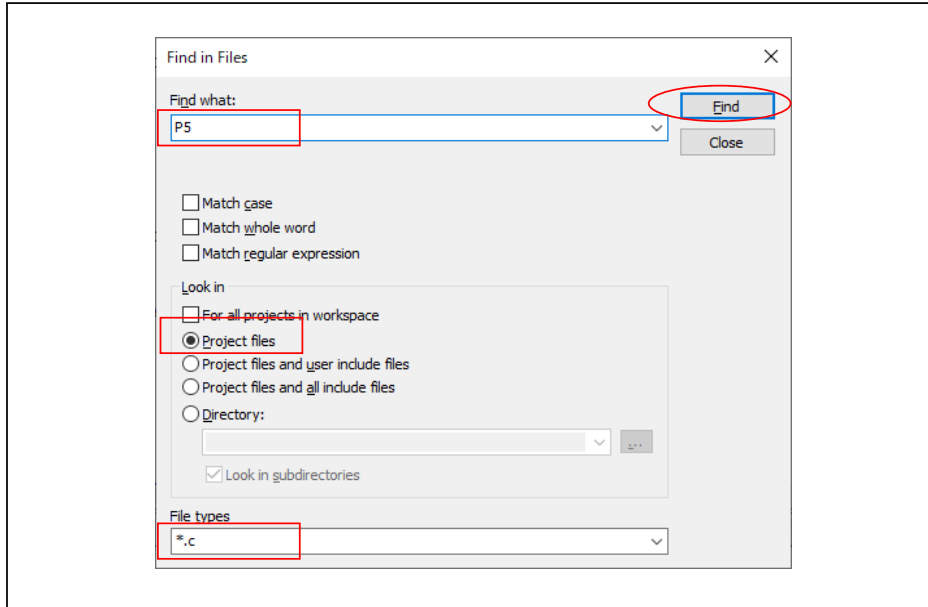


The result is displayed in the [Search] tab.

- IAR EW for Renesas RL78

Click the [Edit] - [Find and Replace] - [Find in files] menu of IAR EW for Renesas RL78. On the [Find in Files] dialog, specify as follows and click the [Find] button.

Figure 2-106 IAR EW for Renesas RL78 [Find in Files] dialog



The result is displayed in the [Find in Files] tab.

2.14 Build project of destination RL78

Build the project of the destination RL78.

- CS+ for CC

Click the [Build] – [Rebuild Project] menu.

- e² studio

Click the [Project] – [Build Project] menu.

- IAR EW for Renesas RL78

Click the [Project] – [Rebuild All] menu.

Note. If any error occurs, take action according to the error message.

3. Example of Porting

This chapter describes a specific example of porting a sample code. Below is the application note used in the porting example.

Table 3-1 Example of porting

Item	Content
Application note (Sample code)	RL78/G23 Timer Array Unit (PWM output) (R01AN5874EJ0100)
Destination device	RL78/G15 (R5F12068)
Integrated development environment	CS+ for CC V8.09.00 from Renesas Electronics Corp.
Compiler	CC-RL V1.12.00 from Renesas Electronics Corp.
Smart Configurator	Smart Configurator for RL78 V1.5.0 from Renesas Electronics Corp.
Overview of porting work	Change component and I/O port

3.1 Advance preparation

Check what resources are used by referring to “2. Operation Confirmation Conditions”, “Table 1-1 Peripheral Function and Use”, “3.2 List of Pins to be Used” and “3.1 Example of Hardware Configuration” in the application note for RL78/G23.

An example configuration for porting to RL78/G15 is shown below.

- Operation Confirmation Conditions

Differences are shown with red color. The RL78/G15 cannot operate with 32MHz, therefore, change the operating frequency to 16MHz. Smart Configurator (SC) changes the related register settings according to the changed operating frequency, therefore, there is no effect on the operation of the sample code. Also, the RL78/G15 does not have the Voltage Detector (LVD), therefore, change it to the Selectable Power-on-reset Circuit (SPOR).

Table 3-2 Operation Confirmation Conditions

Item	RL78/G23	RL78/G15
MCU	RL78/G23 (R7F100GLG)	RL78/G15 (R5F12068)
Board	RL78/G23 Fast Prototyping Board (RTK7RLG230CLG000BJ)	RL78/G15-20p Fast Prototyping Board (RTK5RLG150CLG000BJ)
Operating frequency	High-speed on-chip oscillator clock: 32MHz CPU/peripheral hardware clock: 32MHz	High-speed on-chip oscillator clock: 16MHz CPU/peripheral hardware clock: 16MHz
Operating voltage	3.3 V (can be operated at 1.8 V to 5.5 V) LVD0 detection voltage: Reset mode At rising edge TYP. 1.90 V (1.84 V to 1.95 V) At falling edge TYP. 1.86 V (1.80 V to 1.91 V)	5.0V (can be operated at 2.4V to 5.5V) SPOR detection voltage At rising edge TYP. 2.57 V (2.44 V to 2.68 V) At falling edge TYP. 2.52 V (2.40 V to 2.62 V)

- Peripheral Function and Usage

The same peripheral function and channel used in the sample code of RL78/G23 is used in the RL78/G15.

Table 3-3 Peripheral function to be used

RL78/G23

Peripheral Function	Usage
Timer Array Unit 0 channel 0, channel 1	This unit is used to realize the PWM function by operating channel 0 and channel 1 together and a PWM signal output from the TO01 pin.

RL78/G15

Peripheral Function	Usage
Timer Array Unit 0 channel 0, channel 1	This unit is used to realize the PWM function by operating channel 0 and channel 1 together and a PWM signal output from the TO01 pin.

- Pins to be Used

The differences are shown with red color. The RL78/G15 (R5F12068) does not have P52. Therefore, change P52 to P21.

Table 3-4 Pins to be used

RL78/G23

Pin name	I/O	Function
P16 / TO01	Output	PWM output port (controls LED1 brightness)
P52	Output	Output signal for LED (controls LED2 on/off)

RL78/G15

Pin name	I/O	Function
P04 / TO01	Output	PWM output port (controls LED1 brightness)
P21	Output	Output signal for LED (controls LED2 on/off)

● Hardware Configuration

The differences are shown with red color. The RL78/G15 (R5F12068) does not have P52. Therefore, change P52 to P21.

Figure 3-1 Hardware Configuration (RL78/G23)

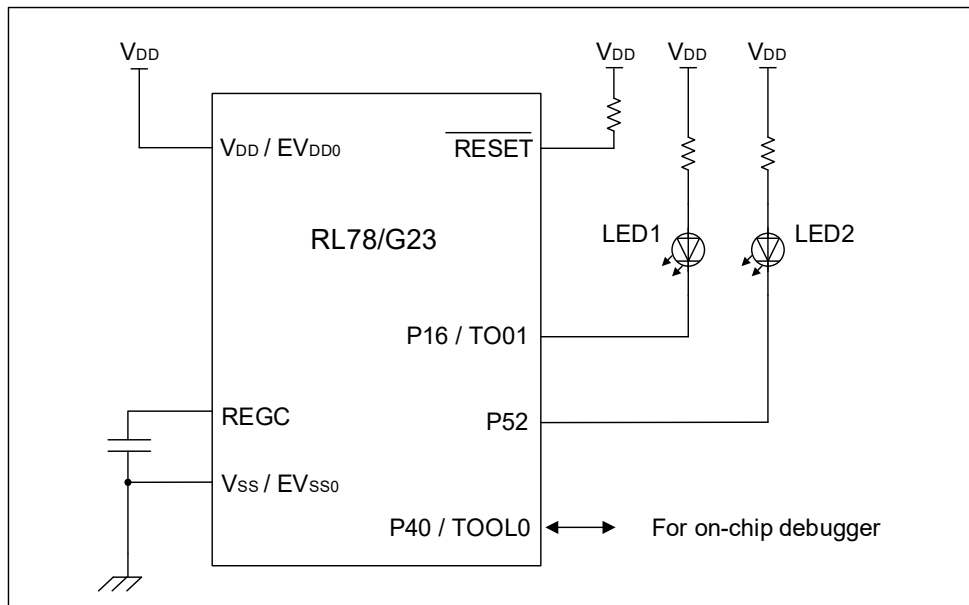
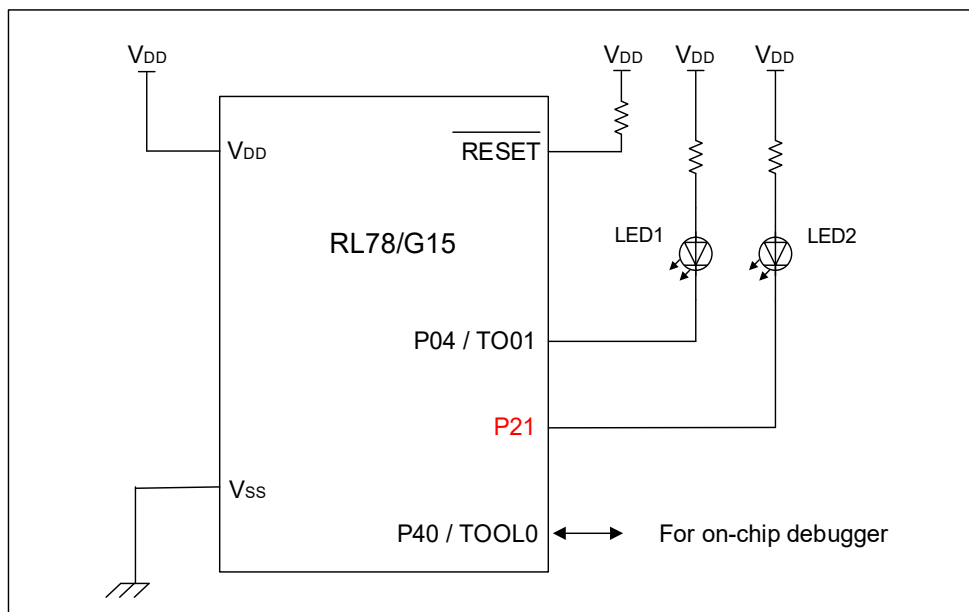


Figure 3-2 Hardware Configuration (RL78/G15)



Note 1. This simplified circuit diagram was created to show an overview of connections only. When actually designing your circuit, make sure the design includes appropriate pin handling and meets electrical characteristic requirements (connect each input-only port to VDD or VSS through a resistor).

2. Connect any pins whose name begins with EVSS to VSS, and any pins whose name begins with EVDD to VDD, respectively.

3. VDD must not be lower than the reset release voltage (VLVD0) that is specified for the LVD0.

3.2 Copy the project

Create a porting project (destination project) by referring to "2.2 Copy the project".

3.3 Open the project

Open the destination project into the development environment by referring to "2.3 Open the project".

3.4 Change the compiler version

Change the compiler version by referring to "2.4 Change the compiler version".

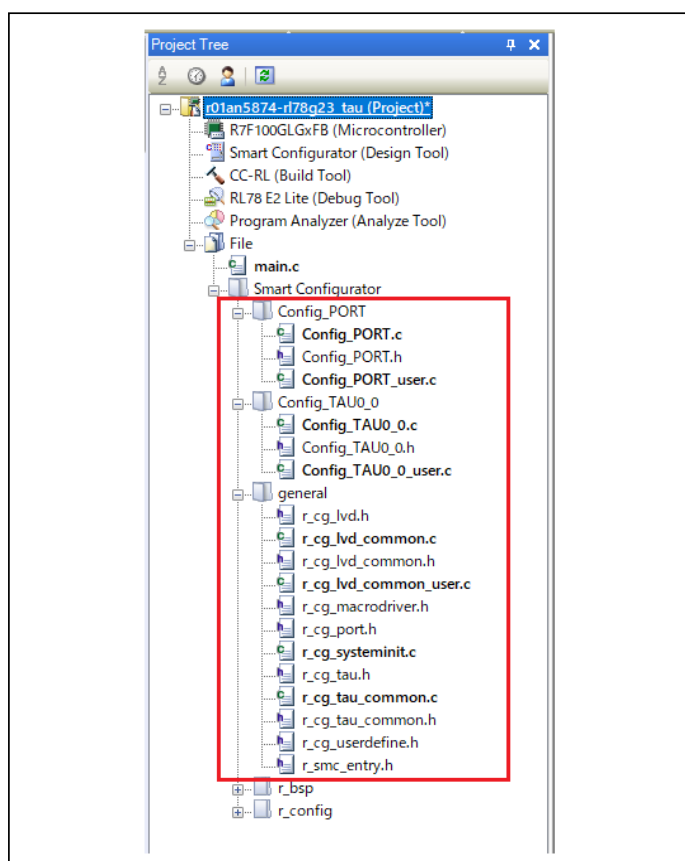
3.5 Check code changes

Check if SC generated code in the source project has modified.

The target are files indicated in Figure 3-3. Open the target files in an editor and check for comments indicating changes for the SC generated code. If there are any changes, make the same changes to the files indicated in Figure 3-3.

In this sample code, there is no change.

Figure 3-3 Check code changes (CS+ for CC screen)



3.6 Change resources

In this sample code, the same channels of the same peripheral function used in the source project are used in the destination project also, therefore, resources are not changed.

3.7 Generate report of SC configuration

Save the SC configuration report of the source project by referring to “2.7 Generate report of SC configuration”.

3.8 Change the device

Change the device of the destination project to “R5F12068xSP” as the destination RL78.

3.9 Check the SC migration report

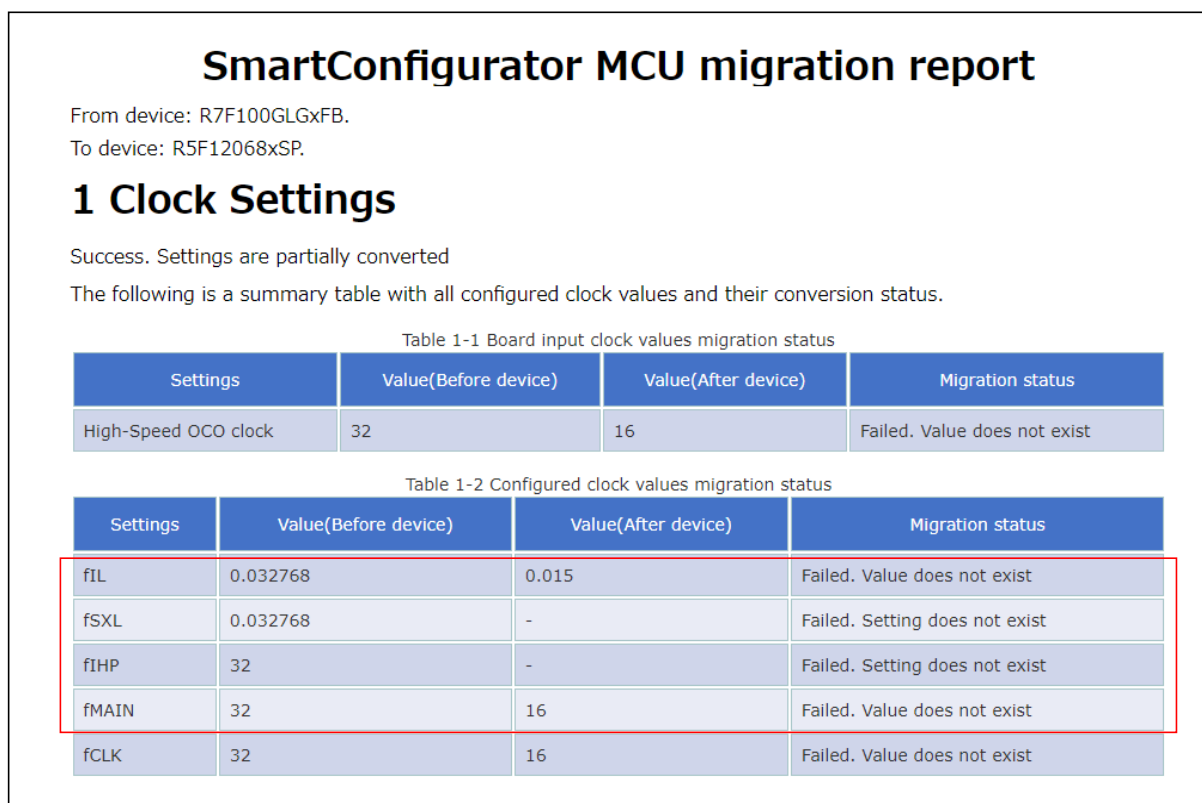
Open the migration report by referring to “2.9 Check the SC migration report”.

3.10 Change settings and resources

Check the migration status of the SC settings from the source project to the destination project in the migration report, then take action according to the example configuration described in "3.1 Advance preparation".

3.10.1 Clock settings

Figure 3-4 Migration report: Clock Settings (1)



- In this sample code, fSXL, fIH and fMAIN are not used. Therefore, “Fails” is no problem.
- fCLK was changed from 32MHz to 16MHz, however it does not affect the operation of this sample code.
- There is no report about the operating voltage, however, change the operating voltage according to the example configuration.

Open the [Clocks] tab of SC. Specify “ $2.4\text{V} \leq \text{VDD} \leq 5.5\text{V}$ ” at the “VDD setting”.

Figure 3-5 [Clocks] tab: VDD setting

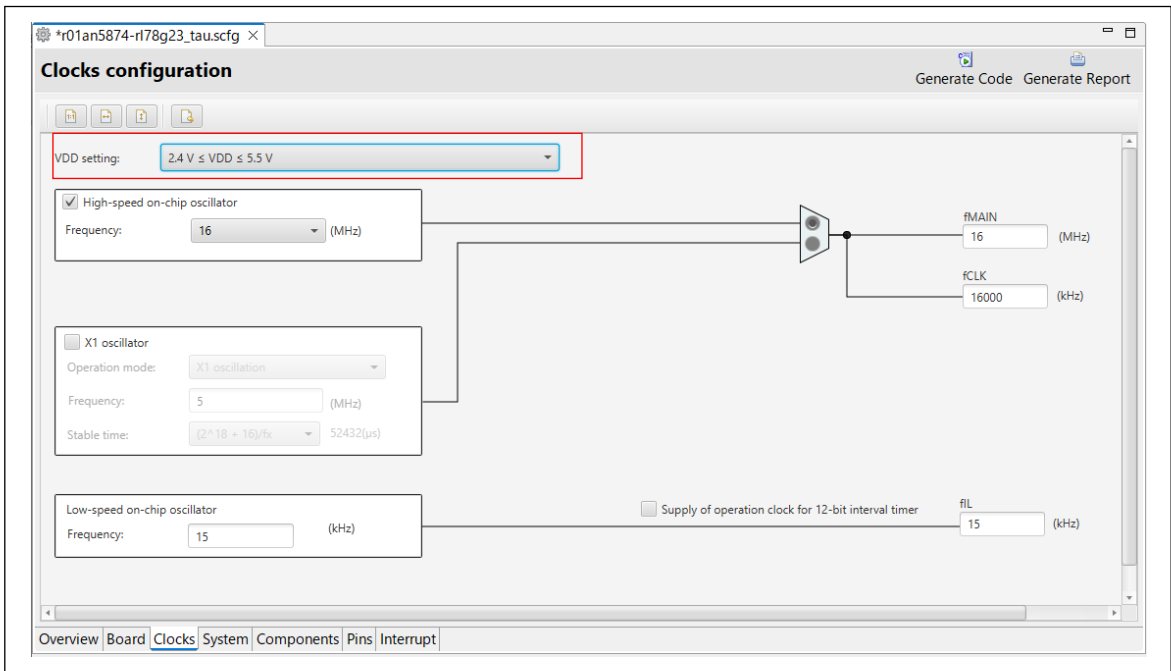


Figure 3-6 Migration report: Clock Settings (2)

Table 1-3 Configured macro definitions migration status			
Settings	Value(Before device)	Value(After device)	Migration status
BSP_CFG_SUBCLK_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_HOCO_DIVIDE	0	-	Failed. Value does not exist
BSP_CFG_SUBSYSCLK_SOURCE	1	-	Failed. Setting does not exist
BSP_CFG_FCLK_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_MOCO_DIVIDE	0	-	Failed. Setting does not exist
BSP_CFG_MOCO_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_XT1_OSCMODE	0	-	Failed. Setting does not exist
BSP_CFG_HISYSCLK_OPERATION	1	1	Success.
BSP_CFG_SUBCLK_OPERATION	1	-	Failed. Setting does not exist
BSP_CFG_MOSC_DIVIDE	0	-	Failed. Setting does not exist
BSP_CFG_OCOCLK_SOURCE	0	-	Failed. Setting does not exist
BSP_CFG_HISYSCLK_SOURCE	0	0	Success.
BSP_CFG_FMX_START_ON_STARTUP	0	-	Failed. Setting does not exist
BSP_CFG_WAKEUP_MODE	0	-	Failed. Setting does not exist
BSP_CFG_FSX_START_ON_STARTUP	0	-	Failed. Setting does not exist
BSP_CFG_ALLOW_FSUB_IN_STOPHALT	0	-	Failed. Setting does not exist
BSP_CFG_FIH_START_ON_STARTUP	1	1	Success.
BSP_CFG_RTC_OUT_CLK_SOURCE	1	-	Failed. Setting does not exist
BSP_CFG_FIM_START_ON_STARTUP	0	-	Failed. Setting does not exist
BSP_CFG_MAINCLK_SOURCE	0	-	Failed. Value does not exist

- The settings shown in the above Table 1-3 are BSP (Board Support Package) setting items. In this porting example, BSP is changed to the latest version in “3.10.4(3)”, therefore do nothing here.

3.10.2 System settings

Figure 3-7 Migration report: System Settings

5 System Settings			
Success. Settings are partially converted			
The following is a summary table with all configured system values and their conversion status.			
Table 5-1 On-chip debug setting migration status			
Settings	Value(Before device)	Value(After device)	Migration status
On-chip debug operation setting	Use emulator	Use emulator	Success.
Emulator setting	E2 Lite	E2 Lite	Success.
Pseudo-RRM/DMM function setting	Used	Used	Success.
Start/Stop function setting	Unused	Unused	Success.
Monitoring point function setting	-	-	Success.
Trace function setting	Used	-	Failed. Setting does not exist
Use security ID	check	check	Success.
Security ID	0x00000000000000000000	0x00000000000000000000	Success.
Security ID authentication failure setting	Do not erase flash memory data	-	Failed. Setting does not exist

- The "Trace function setting" and "Security ID authentication feature setting" settings are not supported in the RL78/G15. There is no problem because it does not affect the operation of the sample code.

3.10.3 Interrupt

Check if the interrupts used in the sample code is migrated. Check the interrupts used in the sample code by the configuration report generated in "3.7 Generate report of SC configuration".

Figure 3-8 Configuration report: Interrupt Settings

7. Interrupt Settings			
The table below shows used interrupts' information:			
Vector Number	Interrupt	Priority	Bank specify
14	INTTM00	Level 3 (low)	None

- INTTM00 is used in the source project.

Figure 3-9 Migration report: Interrupt

2 Interrupt

The following is a summary table with all interrupt assignments and their conversion status

Table 2-1 Interrupt Migration Status

Vector Number	Old Interrupt Assignment	New Interrupt Assignment	Status (Before device change)	Status (After device change)	Interrupt Migration Status
0	INTWDTI	INTWDTI	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
11	INTELCL	INTTM00	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Used Bank specify=None Priority= Level 3 (low)	Success.
12	INTSMSE	INTTM01	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
13	INTST0	INTAD	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
13	INTCSI00	INTAD	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
13	INTIIC00	INTAD	State= Not Use Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.
14	INTTM00	INTP6	State= Used Bank specify=None Priority= Level 3 (low)	State= Not Use Bank specify=None Priority= Level 3 (low)	Success.

- Interrupt vector number 14 "INTMM00" (in the RL78/G23) in the source project was migrated to interrupt vector number 11 "INTTM00" (in the RL78/G15) in the destination project.

3.10.4 Software Component

Check whether peripheral components used in the sample code were migrated.

Figure 3-10 Migration report: Software Components

4 Software Component

The following is a summary table with all configurations and their conversion status.

Table 4-1 Components migration status

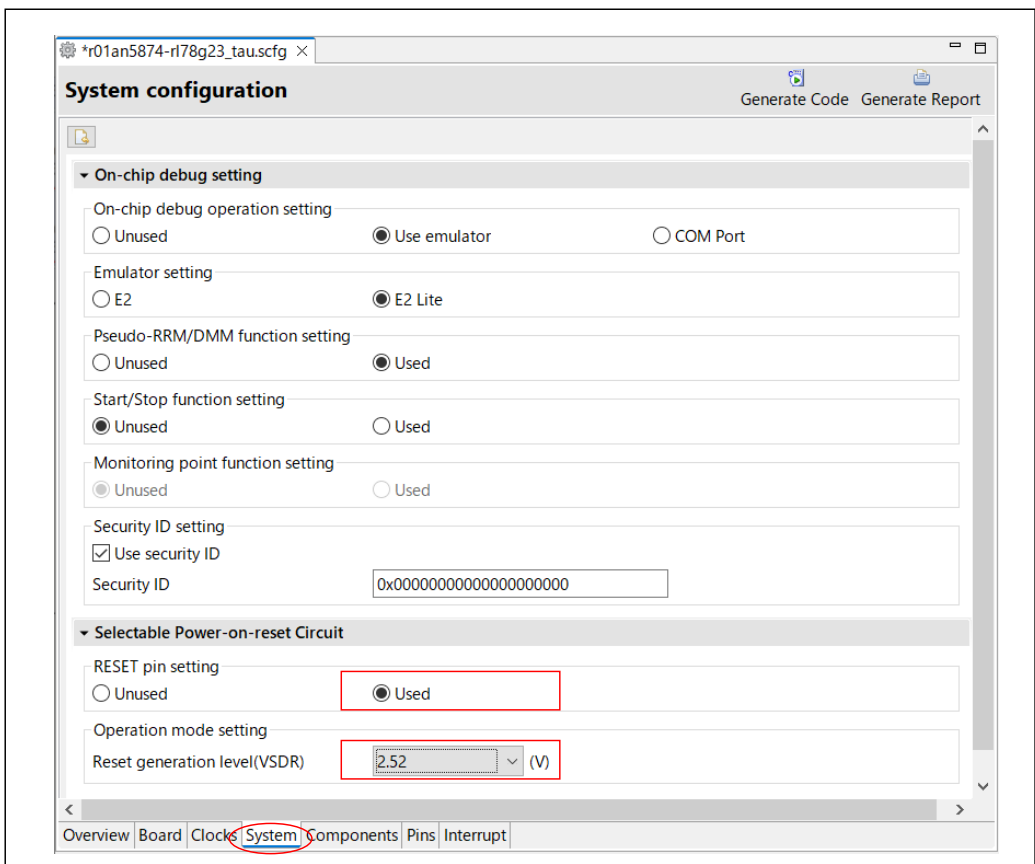
	Configurations	Migration status
(3)	r_bsp	Fails.Please upgrade to the latest version
(1)	Config_LVD0	Fails.Component is not supported in this device
	Config_TAU0_0	Success.Settings are fully converted
(2)	Config_PORT	Success.Settings are partially converted

(1) Config_LVD0

In the RL78/G15, use the Selectable Power-on-reset Circuit (SPOR) instead of the Voltage Detector (LVD).

On the Smart Configurator (SC), click the [System] tab.

Figure 3-11 System tab: Selectable Power-on-reset Circuit



- Specify the “Used” at the “RESET pin setting”.
- Specify the “2.52” at the “Reset generation level(VSDR)”

(2) Config_PORT

About the port, check it at "3.10.5 Pins".

(3) r_bsp

Change the BSP version to the latest version by referring to "2.10.4 (3)Update BSP".

3.10.5 Pins

Figure 3-12 Migration report: Pins

3 Pins			
The following is a summary table with all pin assignments and their conversion status.			
Table 3-1 Pin Migration Status			
Function	Pin Number (Before)	Pin Number (After)	Status
TOOL0	5	4	Success
P52	35	-	Discarded. Function is not available
TO01	40	15	Success

- Change the setting from P52 (Output port) to P21 (Output port) according to the example configuration.

Check the setting of P52 by the configuration report generated in "3.7 Generate report of SC configuration".

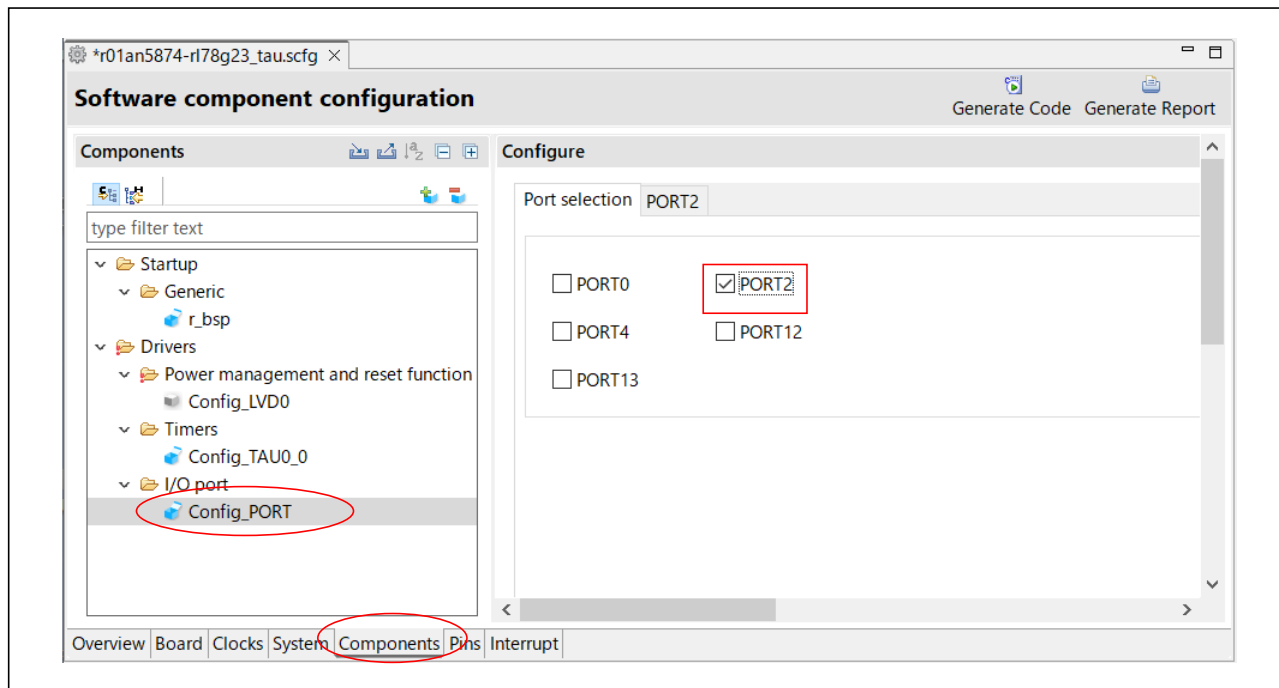
Figure 3-13 Configuration report: Config_PORT

P52:	
Setting name	Value
Use status	Out
Pull-up	Uncheck
Output 1	Check

- P52 is set to Output port and 1 as the initial value in the source project.

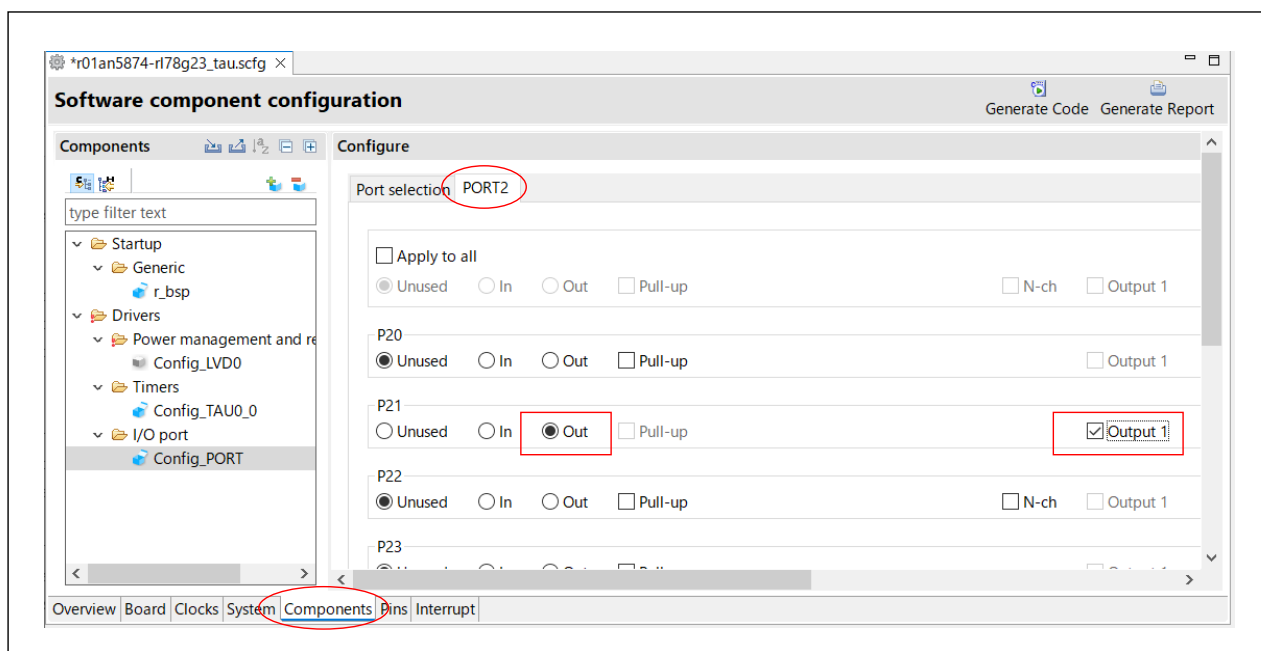
On the Smart Configurator (SC), click the [Components] tab. Click the [Config_PORT] and place a check mark in the [PORT2].

Figure 3-14 [Components] tab: Config_PORT



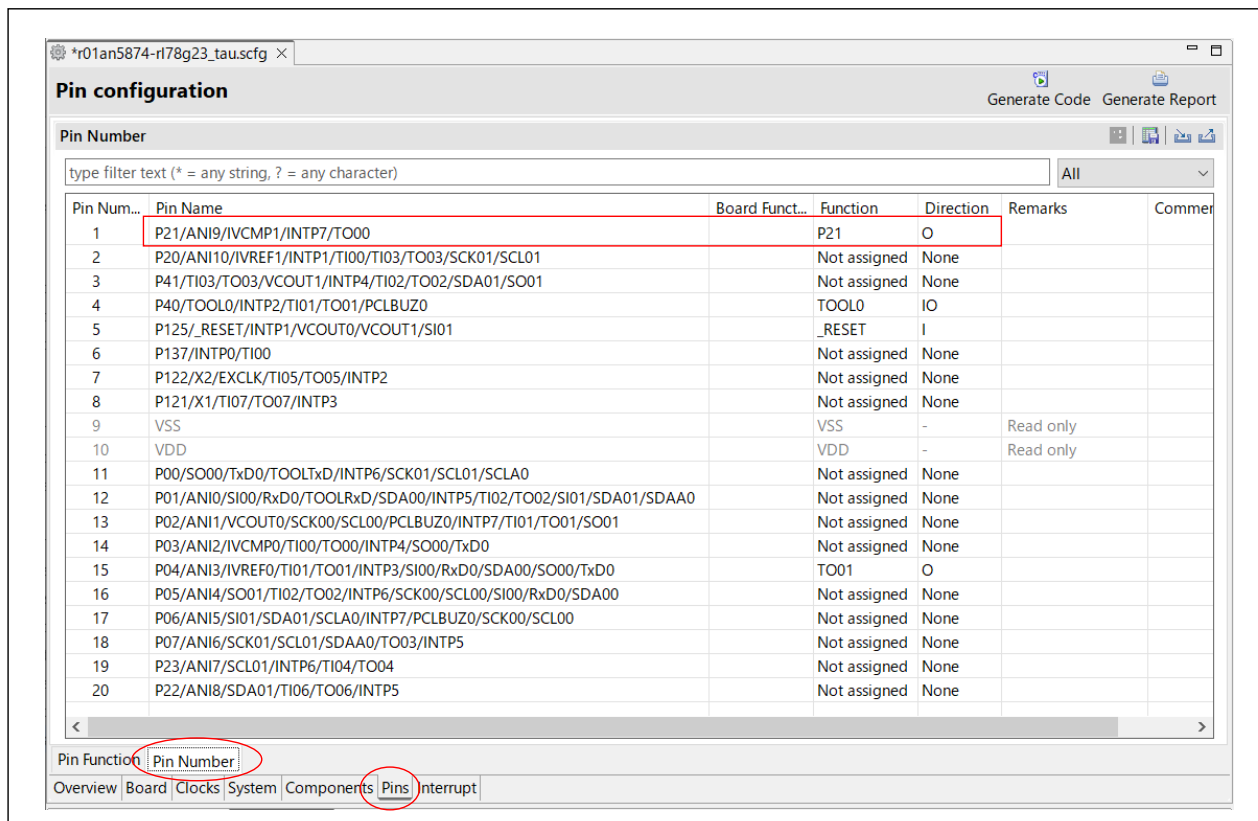
Click the [PORT2] tab. Then, place check marks in the [Out] and the [Output 1] of P21

Figure 3-15 [Components] tab: PORT2



Make sure that P21 is set to the output port in the [Pins] – [Pin Number] tab.

Figure 3-16 [Pins] – [Pin Number] tab



3.11 Save the SC settings

Save the settings of Smart Configurator (SC) by referring to “2.11 Save the SC settings”.

3.12 Code Generation

Generate the code on Smart Configurator (SC) by referring to “2.12 Code generation”.

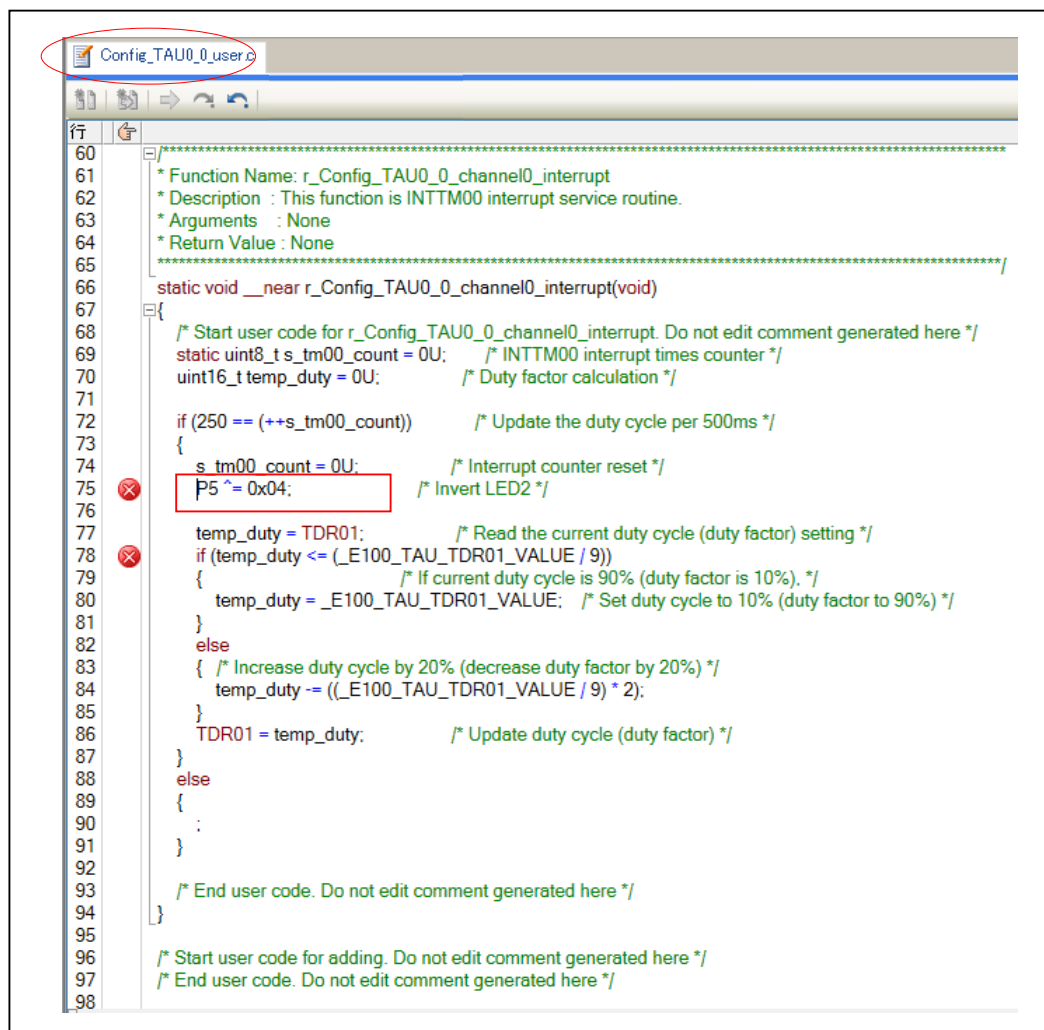
3.13 Edit the code

The output port was changed from P52 to P21 in “3.10.5 Pins”, therefore edit the user codes that set the value for P52. It is not necessary to edit the code other than the user code because the initialization codes such as the port mode register setting are generated by SC according to SC configuration.

In the project file, check the location of the codes that access to P52. Search by the search string as “P5”. (Reference: How to search on each IDE)

From the search results, there is the code for P52 in the user code comments in Config_TAU0_0_user.c. Therefore, edit it.

Figure 3-17 Config_TAU0_0_user.c (Screen of CS+ for CC): Edit 1



- Before change: P5 ^= 0x04; After change: P2 ^= 0x02;

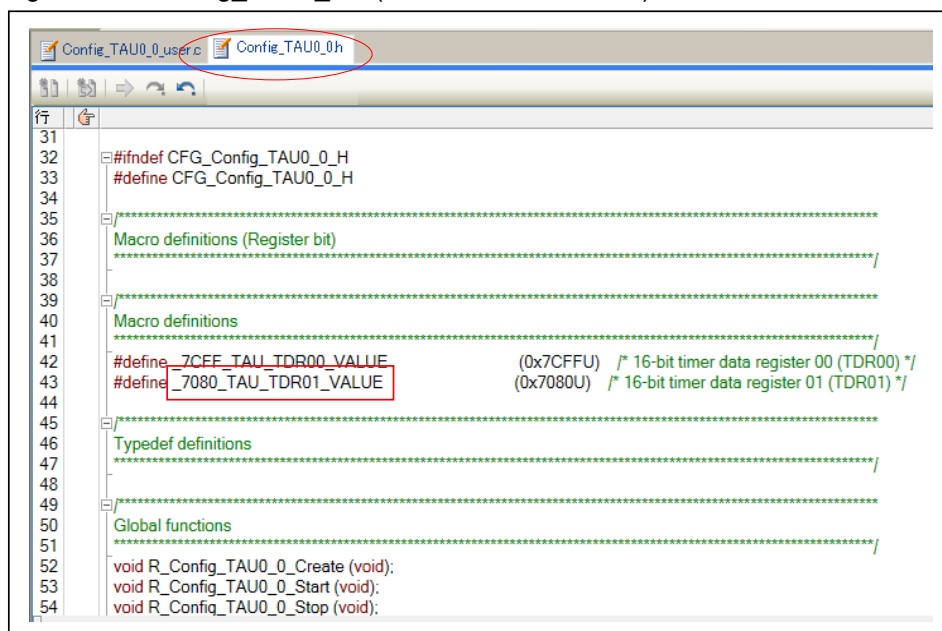
3.14 Build project of destination RL78

Build the destination project by referring to “2.14 Build project of destination RL78”.

As the result of build, the undefinition error occurs for the macro “_E100_TAU_TDR01_VALUE” at the line 78 in Config_TAU0_0_user.c. This macro is used for the initial duty ratio of PWM that uses channel 0 and 1 of the Timer Array Unit0. (Setting value for TRD01 register)

Edit the code according to the initial values of the destination project. Initial values are defined in Config_TAU0_0.h generated by SC.

Figure 3-18 Config_TAU0_0.h (Screen of CS+ for CC)

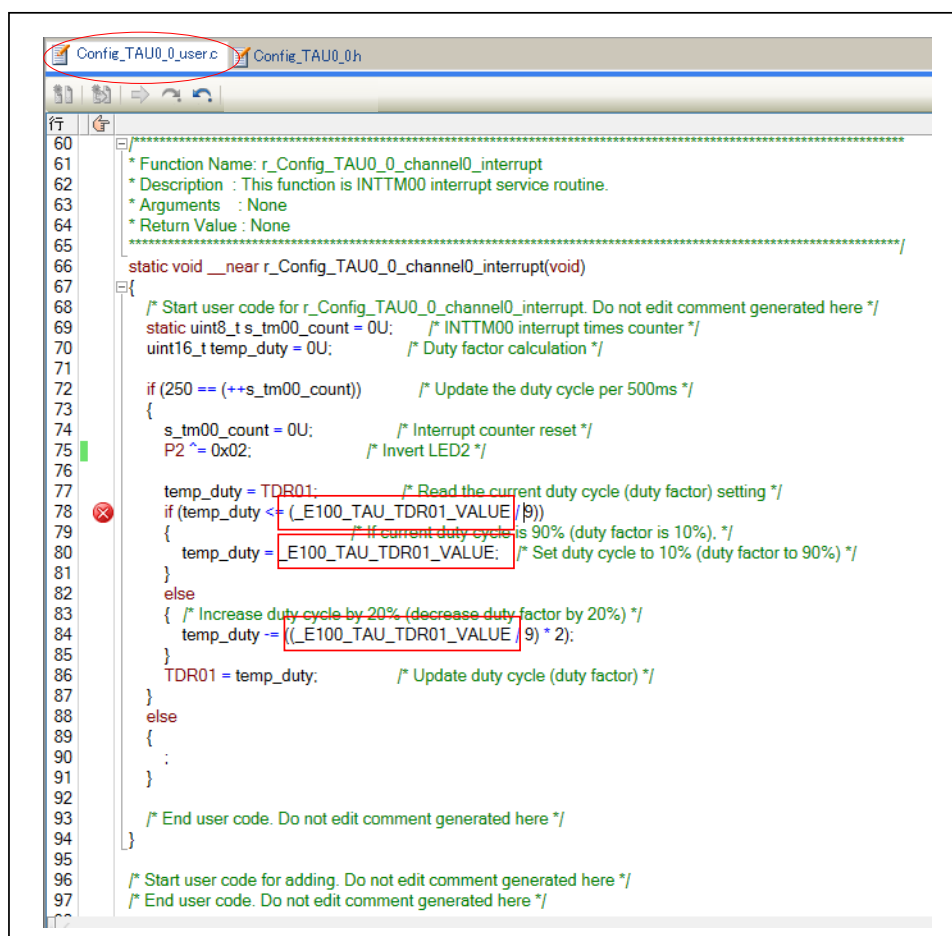


```

31
32 #ifndef CFG_Config_TAU0_0_H
33 #define CFG_Config_TAU0_0_H
34
35
36 Macro definitions (Register bit)
37
38
39
40 Macro definitions
41
42 #define _7CFE_TAU_TDR00_VALUE (0x7CFFU) /* 16-bit timer data register 00 (TDR00) */
43 #define _7080_TAU_TDR01_VALUE (0x7080U) /* 16-bit timer data register 01 (TDR01) */
44
45
46 Typedef definitions
47
48
49
50 Global functions
51
52 void R_Config_TAU0_0_Create (void);
53 void R_Config_TAU0_0_Start (void);
54 void R_Config_TAU0_0_Stop (void);

```

Figure 3-19 Config_TAU0_0_user.c (Screen of CS+ for CC): Edit 2



```

60
61 * Function Name: r_Config_TAU0_0_channel0_interrupt
62 * Description : This function is INTTM00 interrupt service routine.
63 * Arguments : None
64 * Return Value : None
65
66 static void __near r_Config_TAU0_0_channel0_interrupt(void)
67 {
68     /* Start user code for r_Config_TAU0_0_channel0_interrupt. Do not edit comment generated here */
69     static uint8_t s_tm00_count = 0U; /* INTTM00 interrupt times counter */
70     uint16_t temp_duty = 0U; /* Duty factor calculation */
71
72     if (250 == (++s_tm00_count)) /* Update the duty cycle per 500ms */
73     {
74         s_tm00_count = 0U; /* Interrupt counter reset */
75         P2 ^= 0x02; /* Invert LED2 */
76
77         temp_duty = TDR01; /* Read the current duty cycle (duty factor) setting */
78         if (temp_duty <= (_E100_TAU_TDR01_VALUE / 9))
79         {
80             temp_duty = _E100_TAU_TDR01_VALUE; /* Set duty cycle to 10% (duty factor to 90%) */
81         }
82         else
83         {
84             /* Increase duty cycle by 20% (decrease duty factor by 20%) */
85             temp_duty = ((_E100_TAU_TDR01_VALUE / 9) * 2);
86         }
87         TDR01 = temp_duty; /* Update duty cycle (duty factor) */
88     }
89     else
90     {
91         :
92     }
93
94     /* End user code. Do not edit comment generated here */
95 }
96
97 /* Start user code for adding. Do not edit comment generated here */
98 /* End user code. Do not edit comment generated here */

```

- Before change: `_E100_TAU_TDR01_VALUE` After change: `_7080_TAU_TDR01_VALUE`

The error will disappear after re-build. The porting of the sample code is finished.

4. Sample code

Sample code can be downloaded from the Renesas Electronics website.

5. Reference Document

RL78/G23 User's Manual: Hardware (R01UH0896)

RL78 family user's manual software (R01US0015)

Smart Configurator User's Manual: RL78 API Reference (R20AN4852)

RL78 Smart Configurator User's Guide: CS+ (R20AN0580)

RL78 Smart Configurator User's Guide: e² studio (R20AN0579)

RL78 Smart Configurator User's Guide: IAREW (R20AN0581)

RL78 Family Board Support Package Module Using Software Integration System (R20AN5522)

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

Technical update

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

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

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 07, 2023	-	First edition

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/.