

RX Smart Configurator

R20AN0535EJ0100

Rev.1.00

Jan 25, 2019

User's Guide: IAREW

Introduction

This application note describes the basic usage of the RX Smart Configurator (hereafter called the Smart Configurator), and the procedure for importing its output files to IAR Embedded Workbench.

References to the Smart Configurator and Integrated Development Environment (IDE) in this application note apply to the following versions.

- RX Smart Configurator V2.0.0 or later
- IAR Embedded Workbench for RX V4.10 or later

Target device and support compiler

Refer to the following URL for the range of supported devices:

<https://www.renesas.com/smart-configurator>

Contents

1. Overview	4
1.1 Purpose	4
1.2 Features	4
2. Installation and uninstallation	5
2.1 Installing the Smart Configurator	5
2.2 Uninstalling the Smart Configurator	5
3. Operating the Smart Configurator.....	6
3.1 Procedure for Operations.....	6
3.2 Starting the Smart Configurator	7
3.3 Create and loading a configuration file	8
3.3.1 Creating a New Configuration File	8
3.3.2 Opening an Existing Configuration File	8
3.4 Window.....	9
3.4.1 Main menu	10
3.4.2 Toolbar	10
3.4.3 Smart Configurator view	11
3.4.4 MCU Package view	11
3.4.5 Console view	12
3.4.6 Configuration Problems view	12
4. Setting of Peripheral Modules	13
4.1 Board setting	13

4.1.1	Selecting the device	エラー! ブックマークが定義されていません。
4.1.2	Selecting the board.....	14
4.1.3	Import of board configuration	15
4.1.4	Export of board configuration	15
4.2	Clock settings.....	16
4.3	Software component settings.....	17
4.3.1	Adding component	17
4.3.2	Switching between the component view and hardware view	18
4.3.3	Removing a component	19
4.3.4	Component configuration settings	20
4.3.5	Component resource change	21
4.4	Pin settings.....	24
4.4.1	Assign pins to resources	25
4.4.2	Pin setting using MCU package	26
4.4.3	Export pin settings	27
4.4.4	Import pin settings.....	27
4.4.5	Pin setting using board pin configuration information.....	28
4.4.6	Pin filter feature.....	28
4.5	Interrupt settings.....	29
4.5.1	Changing the interrupt priority level and fast interrupt setting	29
4.5.2	Changing the interrupt priority level and fast interrupt setting	30
5.	Managing Conflicts	31
5.1	Resource conflicts	31
5.2	Resolving pin conflicts.....	32
6.	Generating Source Code	33
6.1	Generating Source Code File.....	33
6.2	Configuration of Generated Files and File Names.....	33
6.3	Initializing Clocks.....	35
6.4	Initializing Pins	36
6.5	Initializing Interrupts.....	37
6.6	Backing up Generated Source Code.....	37
7.	Loading generated files in Integrated development environment	38
7.1	Loading in IAR Embedded Workbench.....	38
8.	Creating User Programs	39
8.1	Adding Custom Code in the Case of Code Generator	39
9.	Generating Reports.....	41
9.1	Report on Configuration.....	41
9.2	Configuration of Pin Function List and Pin Number List (in csv Format).....	42

9.3 Image of MCU Package (in png Format)	42
10. Help	43
10.1 Help.....	43
11. Documents for Reference	44

1. Overview

1.1 Purpose

This application note describes the basic usage of the RX Smart Configurator (hereafter called the Smart Configurator), and the procedure for importing its output files to IAR Embedded Workbench.

Refer to the User's Manual of IAR Embedded Workbench for how to use them.

1.2 Features

The Smart Configurator is a utility for combining software to meet your needs. It handles the following two functions to support the embedding of drivers from Renesas in your systems: Generating driver code and making pin settings.

2. Installation and uninstallation

This section describes the installation and uninstallation.

2.1 Installing the Smart Configurator

Download the Smart Configurator from the URL below.

<https://www.renesas.com/smart-configurator>

After activating the installer, install the Smart Configurator and the plug-in by following the procedure of the installer. You will require administrator privileges to do this.

2.2 Uninstalling the Smart Configurator

If you wish to uninstall the Smart Configurator, select “Smart Configurator for RX” from [Programs and Features] in the control panel.

3. Operating the Smart Configurator

3.1 Procedure for Operations

Figure 3-1 Operating procedure, shows the procedure for generating a source file using Smart Configurator and loading it into IAR Embedded Workbench. For the operation of IAR Embedded Workbench, refer to relevant document of IAR.

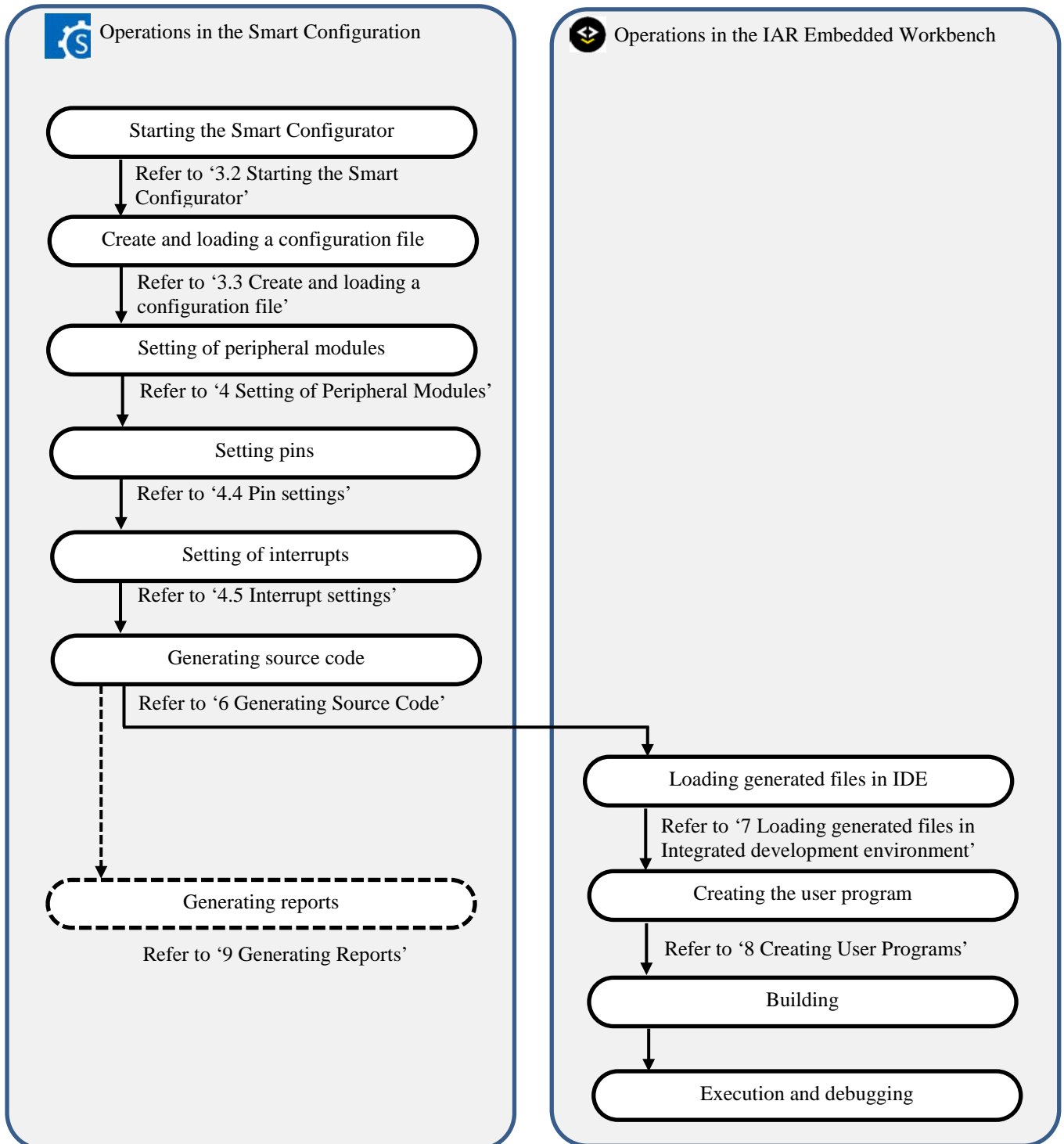


Figure 3-1 Operating Procedure

3.2 Starting the Smart Configurator

Select [Smart Configurator for RX Vx.x.x] of [Renesas Electronics Smart Configurator] from the Windows start menu. The main window of the Smart Configurator will be starting.

Note: Please replace Vx.x.x with your version.

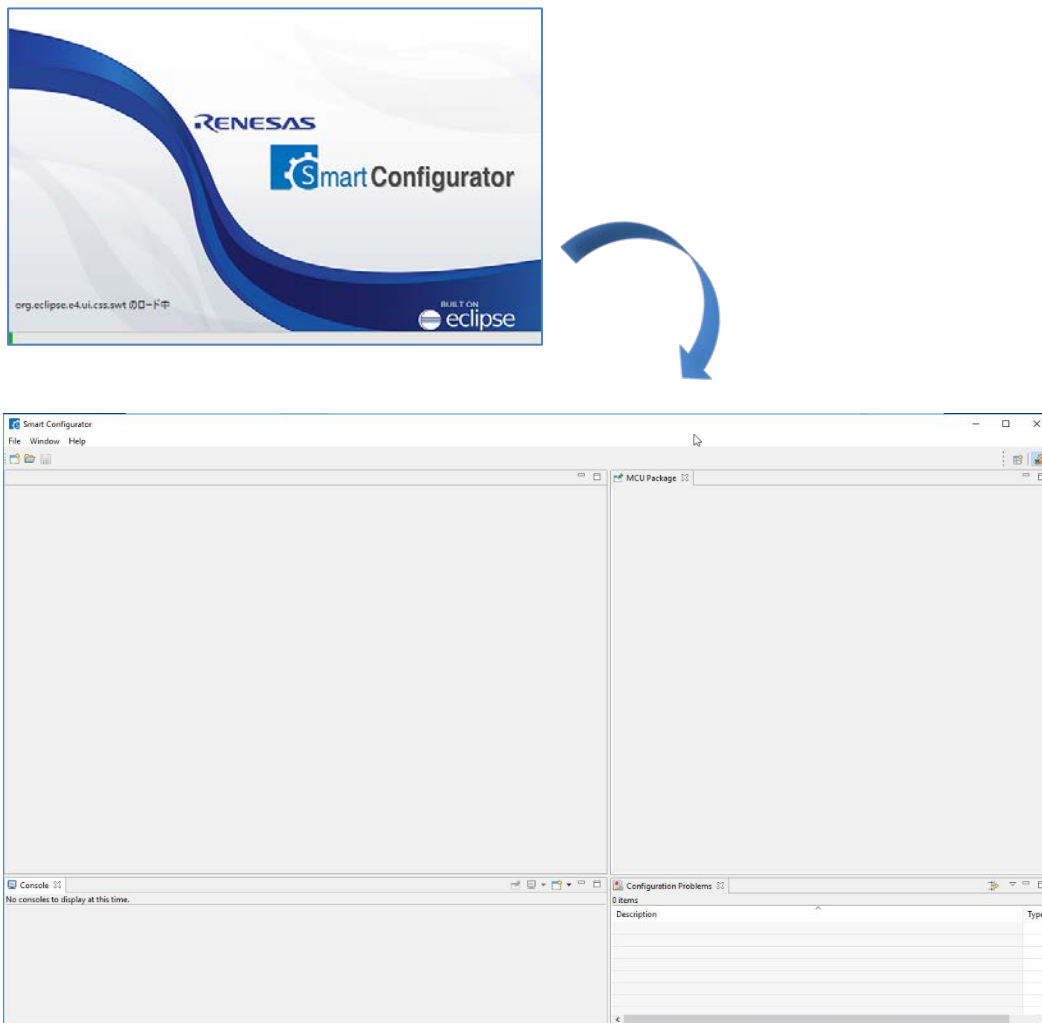



Figure 3-2 Starting of Smart Configurator

3.3 Create and loading a configuration file

Smart Configurator saves and refers to the configuration file (*.scfg) the configuration information of the microcontroller, build tool, peripheral function, pin function etc. used in the project.

3.3.1 Creating a New Configuration File

On the main window, click the  [New Configuration File] button to display the [New Smart Configuration File] dialog box.

- (1) In [Platform:], select the device.
- (2) In [Toolchain:], select [IAR EWRX Toolchain].
- (3) In [File name:], enter the file name.
- (4) Confirm [Location:]. If you want to change it, click [Browse] and select the save destination.
- (5) Click [Finish] to create the configuration file.

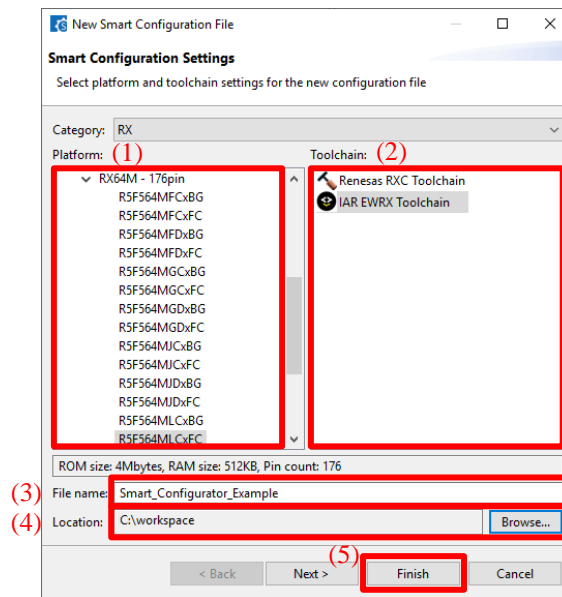


Figure 3-3 Create a Configuration File

3.3.2 Opening an Existing Configuration File

On the main window, click the  [Opening an Existing Configuration File] button to display the [Open] dialog box. Select the file and click [Open].

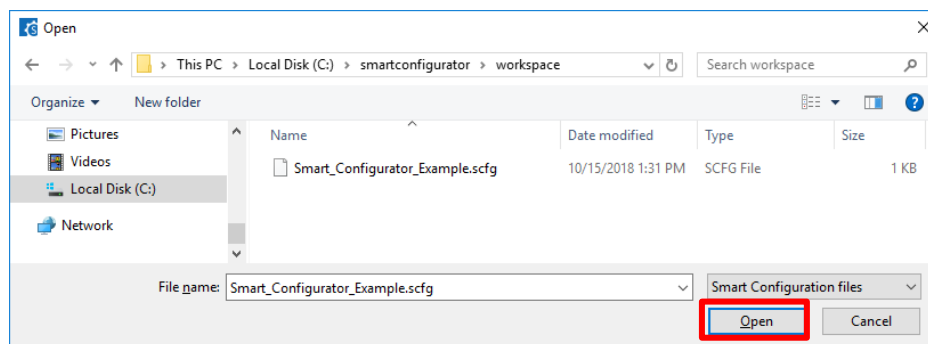


Figure 3-4 Opening an Existing Configuration File

3.4 Window

The main window is displayed when the Smart Configurator is started. The configuration of the window is shown in Figure 3-5, Main Window.

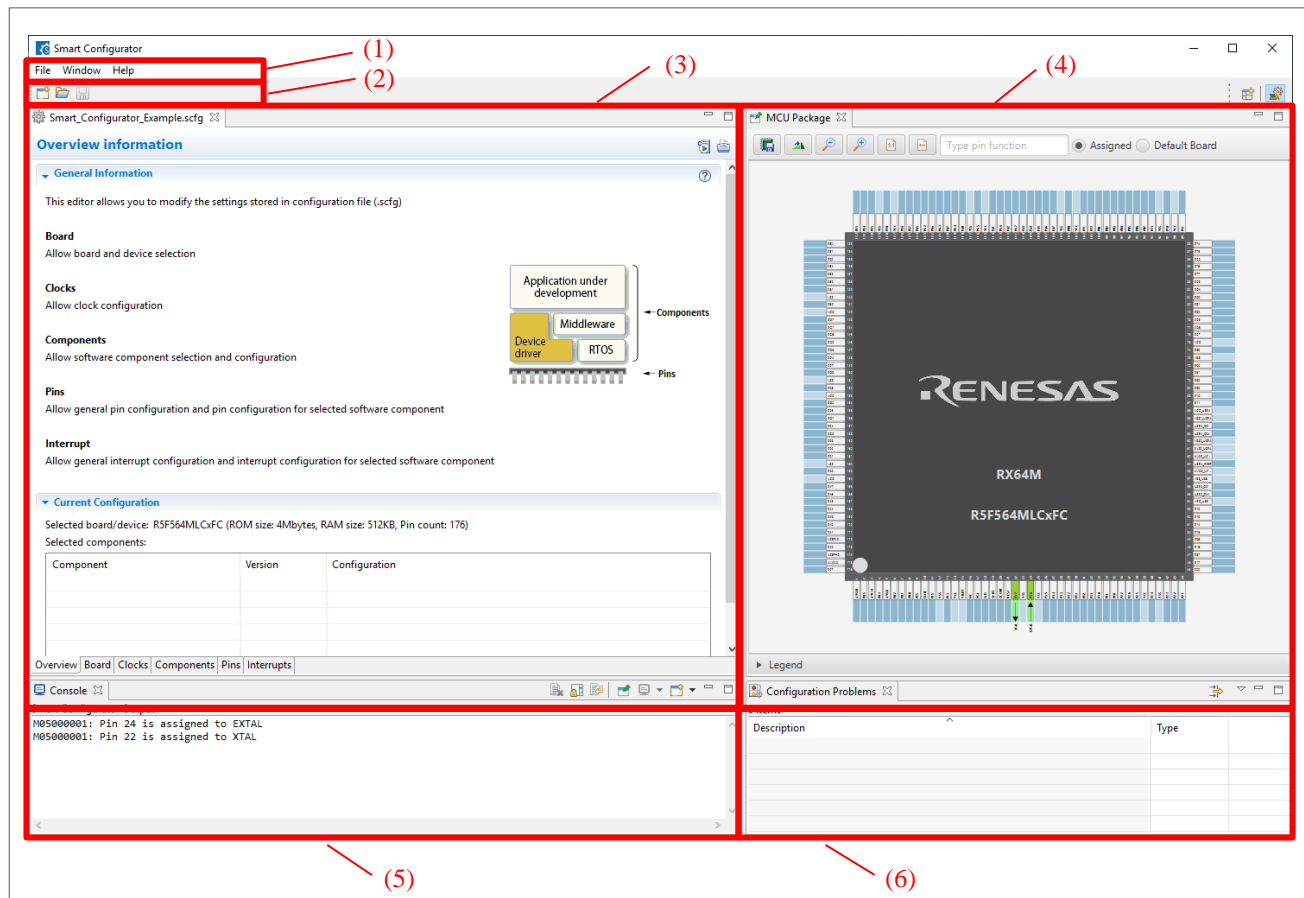


Figure 3-5 Main Window

- (1) Menu bar
- (2) Main tool bar
- (3) Smart Configurator view
- (4) MCU Package view
- (5) Console view
- (6) Configuration Problems view

3.4.1 Main menu

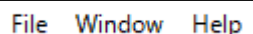


Table 3-1, Main Menu Items, lists the items of the main menu.

Table 3-1 Main Menu Items




Menu		Details
File	New	The dialog box [New Smart Configuration File], which is used to create a new configuration file, is displayed.
	Open	The dialog box [Open], which opens an existing configuration file, is displayed.
	Save	Saves a configuration file with the same name.
	Restart	Smart Configurator is re-started.
	Exit	Execution of the Smart Configurator is terminated.
Window	Preference	The dialog box [Preference], which is used to specify the properties of the configuration file, is displayed.
	Show view	The dialog box [Show view], which is used to set the view of the window, is displayed.
Help	Help Contents	The help menu is displayed.
	About	The version information is displayed.

3.4.2 Toolbar



Some functions of the main menu are allocated to the buttons on the toolbar. Table 3-2, Toolbar Buttons and Related Menu Items, shows the description of those tool buttons.

Table 3-2 Toolbar Buttons and Related Menu Items

Toolbar button	Related menu item
	[File] → [New Smart Configuration File]
	[File] → [Open]
	[File] → [Save]

3.4.3 Smart Configurator view

The Smart Configurator view consists of six pages: [Overview information], [Board], [Clocks], [Components], [Pins], and [Interrupts]. Select a page by clicking on a tab; the displayed page will be changed.

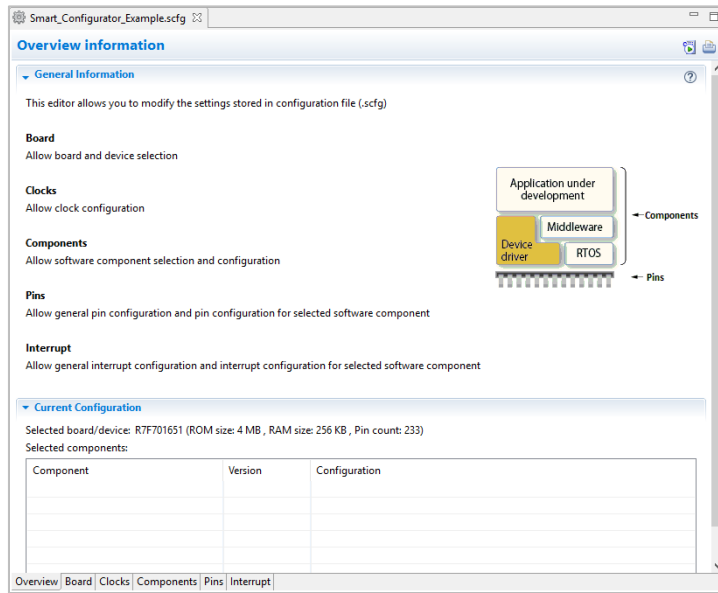


Figure 3-6 Smart Configurator View

3.4.4 MCU Package view

Display the MCU package. You can save rotation, enlargement, reduction, and MCU package view of the display to the image file. You can also confirmation pin assignment and change it.

Two types of package view can be switched between [Assigned] and [Default Board]. [Assigned] displays the assignment status of the pin setting, and [Default Board] displays the initial pin setting information of the board. To specify Board, set [Board:] on the [Board] page.

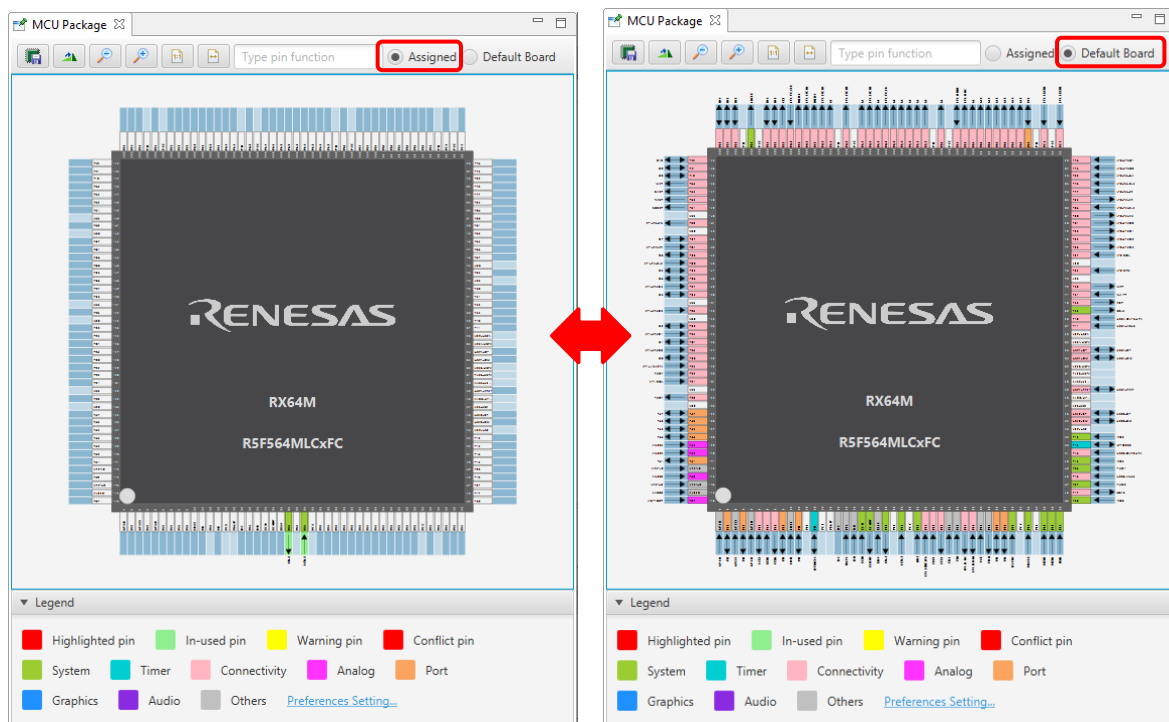


Figure 3-7 MCU Package View

3.4.5 Console view

The console displays details of changes to the configuration made in the Smart Configurator or MCU Package view.

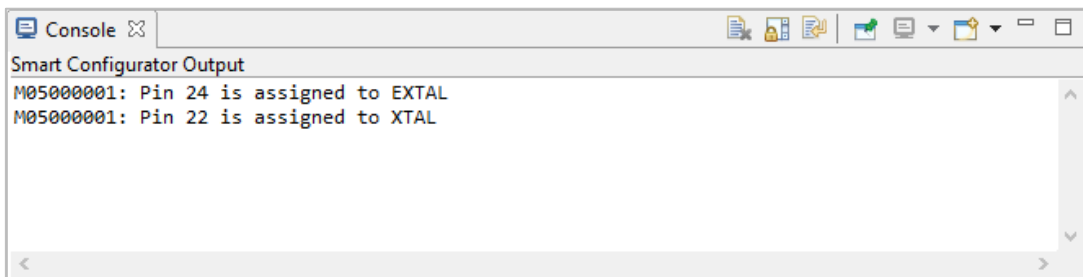


Figure 3-8 Console View

3.4.6 Configuration Problems view

The Configuration Problems view displays problems with peripheral functions, interrupts, and pin conflicts.

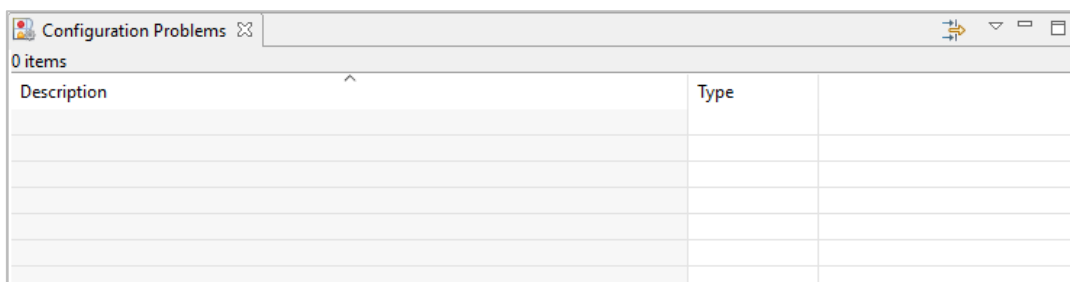


Figure 3-9 Configuration Problems View

4. Setting of Peripheral Modules

You can select peripheral modules from the Smart Configurator view.

4.1 Board setting

On the [Board] page, you can select boards and change devices.

4.1.1 Selecting the device

Click on the [...] button to select a device.

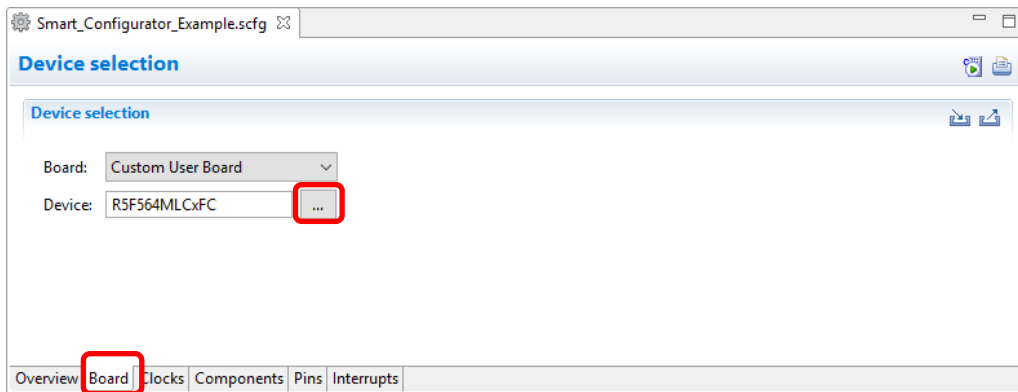


Figure 4-1 Selecting the Device

The following message is displayed when changing the device. For each button operation, refer to "Table 4-1, Device change confirmation operation list".

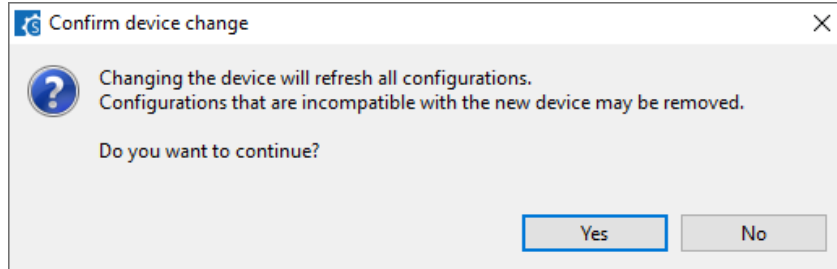


Figure 4-2 Confirm Device Change

Table 4-1 Device Change Confirmation Operation List

Button	Operation explanation
Yes	Change to the selected device.
No	It does not change the device.
Save and continue	After saving the current configuration contents to the configuration file, change to the selected device.
Continue	Changes to the selected device without saving the current configuration contents to the configuration file.
Cancel	It does not change the device.

4.1.2 Selecting the board

Click on the [...] button to select a device. When peripheral functions are configured by board selection, pins are automatically set according to board connection.

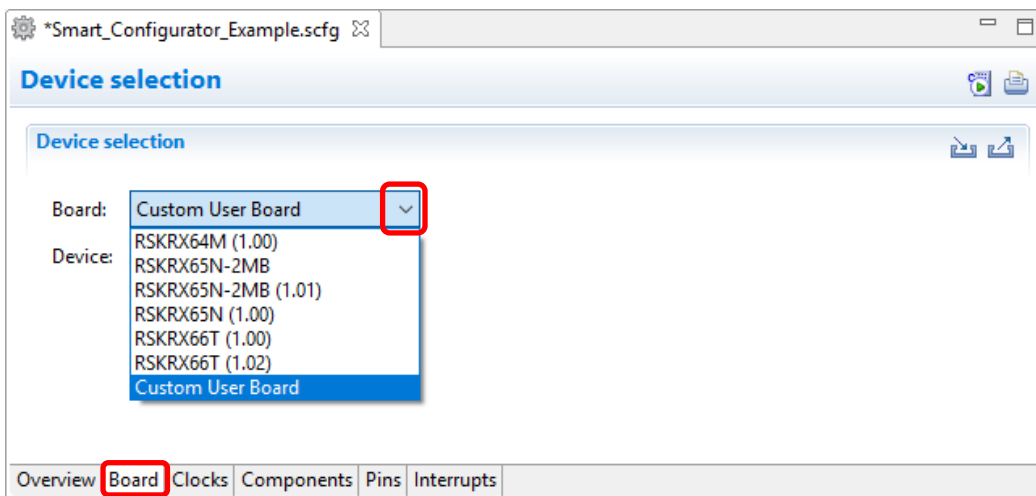


Figure 4-3 Selecting the Board

The following items are changed according to the configuration of the selected board.

- Pin assignment
- Frequency of the main clock
- Frequency of the sub-clock
- Target device

If you change the board, the message shown in “Figure 4-2” or the following message will be displayed. For each button operation, refer to "Table 4-2, Board Change Confirmation Operation List".

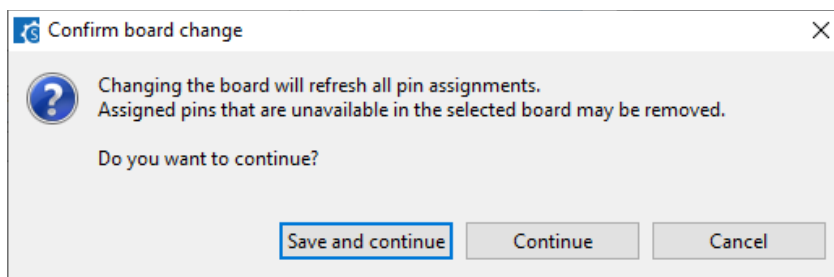



Figure 4-4 Confirm Board Change

Table 4-2 Board Change Confirmation Operation List

Button	Operation explanation
Save and continue	After saving the current configuration contents to the configuration file, change to the selected device.
Continue	Changes to the selected device without saving the current configuration contents to the configuration file.
Cancel	It does not change the device.

4.1.3 Import of board configuration

The board setting is defined in bdf (Board Description File). Follow the procedure below to import board configuration.

- (1) Click on the [ (Import board setting)] button and select a desired bdf file.
- (2) The board of the imported settings is added to the board selection menu.

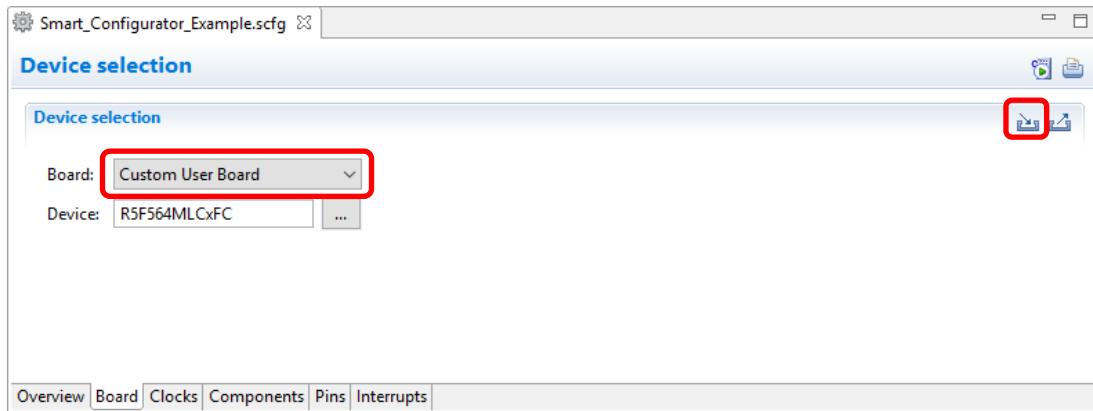



Figure 4-5 Import of Board Configuration (bdf format)

Once a board setting file is imported, the added board is also displayed in the board selection menu of other projects for the same device group.

4.1.4 Export of board configuration

The current main clock frequency, sub clock frequency and pin assignment settings can be exported as board configuration. Follow the procedure below to export the board configuration.

- (1) Click on the [ (Export board setting)] button on the [Board] tabbed page.
- (2) Select the output location and specify a name (Display Name) for the file to be exported.

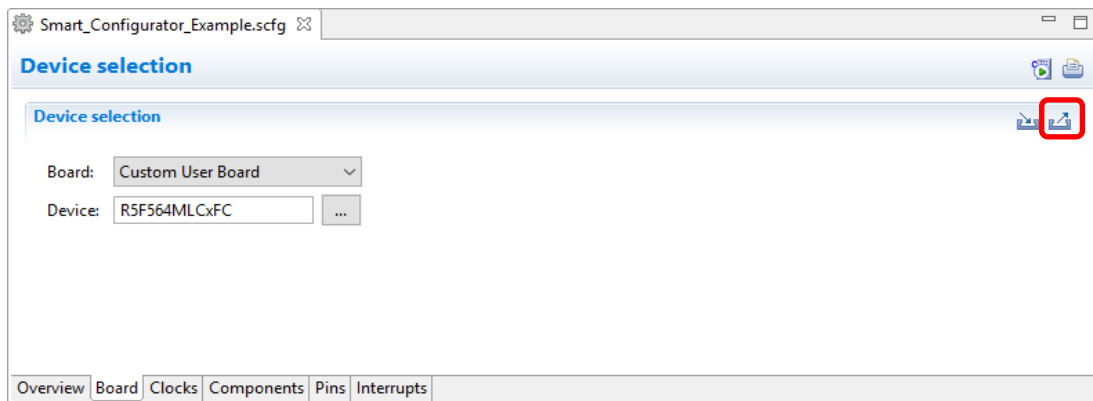


Figure 4-6 Export of Board Configuration (bdf format)

4.2 Clock settings

On the [Clocks] page, set the clock. The [Clocks] page setting is used as the clock source for each component. Set the clock before configuring the component.

The clocks setting is performed in the following procedure.

- (1) Set the clock oscillator circuit.
 - (a) When you move the mouse on the screen, the clock signal is displayed in blue.
 - (b) Click on the screen to select the clock selector.
- (2) Sets the clock source to be supplied to the CPU and peripheral functions.
 - (a) When you move the mouse on the screen, the clock signal is displayed in blue.
 - (b) Click on the screen to select the clock selector.

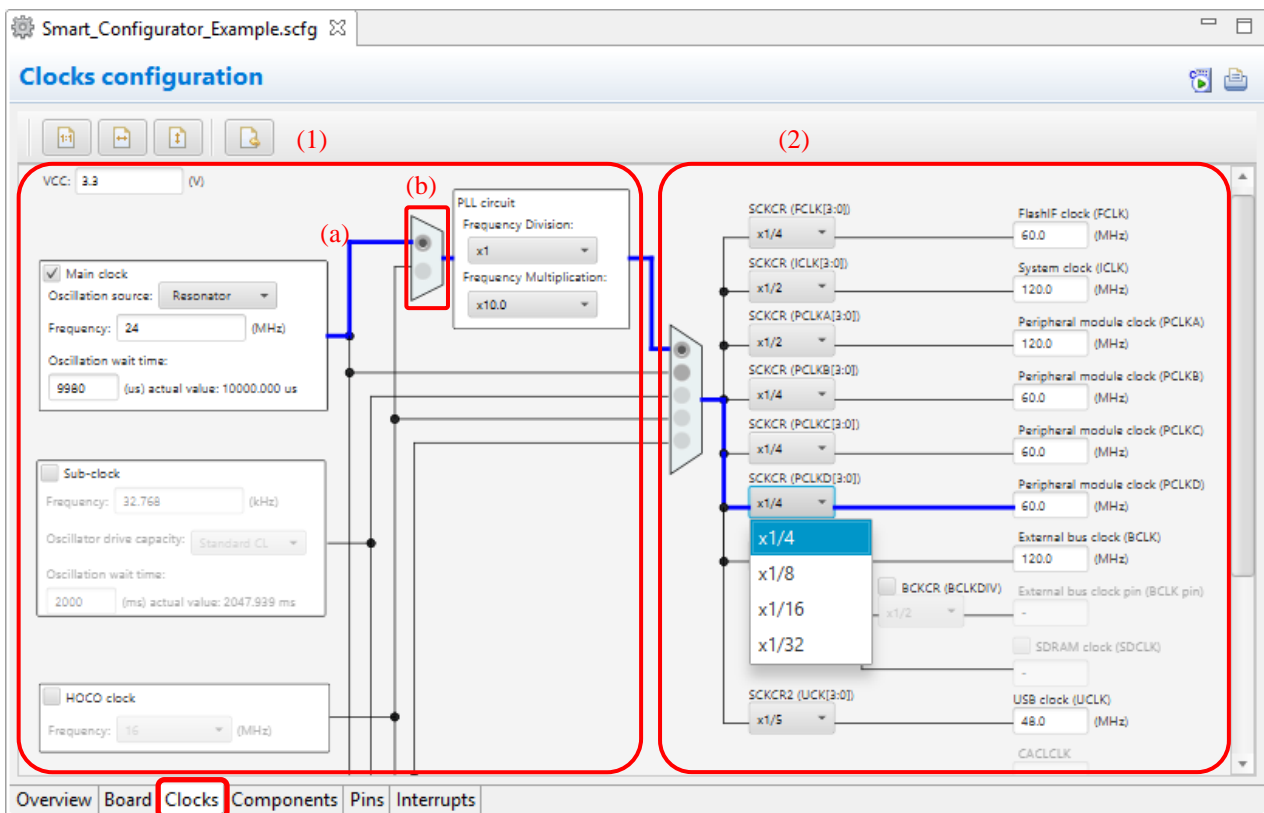


Figure 4-7 Clock Settings

4.3 Software component settings

Drivers can be combined as software components on the [Components] page. Added components are displayed in the component tree at the left of the page.

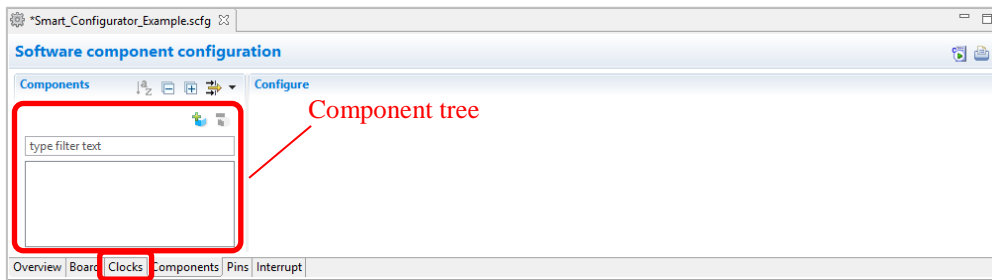


Figure 4-8 Component Page

4.3.1 Adding component

The following describes the procedure for adding a component.

- (1) Click on the [ (Add component)] icon.

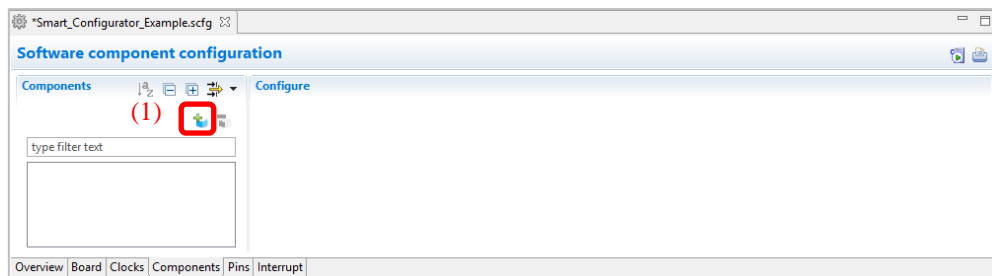


Figure 4-9 Adding Components

- (2) Select a component from the list in the [Software Component Selection] page of the [New Component] dialog box (e.g. Single Scan Mode S12AD).
- (3) Click on [Next].

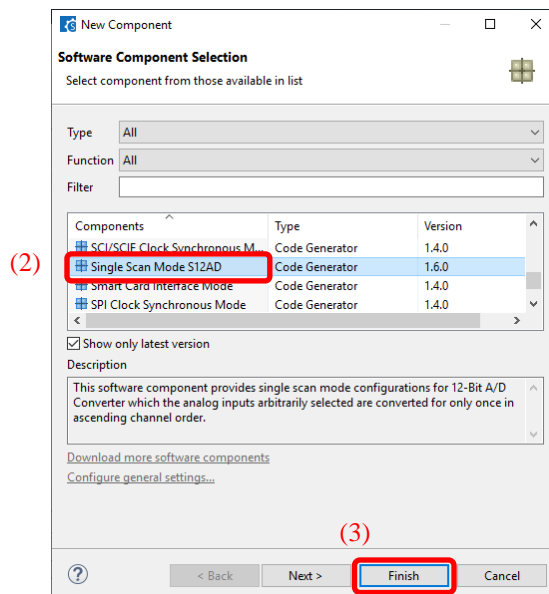


Figure 4-10 Selection of Software Components

- (4) Specify an appropriate configuration name in the [Add new configuration for selected component] page or use the default name (e.g. Config_S12AD0).
- (5) Select a hardware resource or use the default resource (e.g. S12AD0).
- (6) Click on [Finish]. The component is added to the component tree.

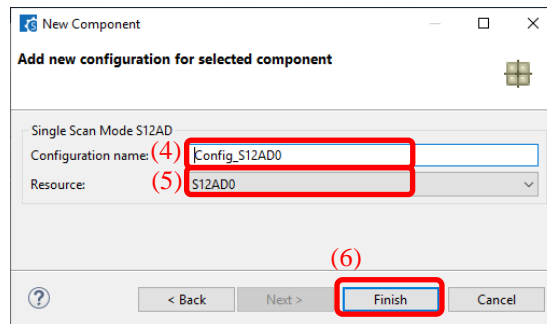


Figure 4-11 Add New Configuration for Selected Component (e.g. S12AD0)

4.3.2 Switching between the component view and hardware view

The Smart Configurator also provides a function for adding a new component by directly clicking a node in the Components tree. To use this function, you need to switch the view of the Components tree from the component view to the hardware view.

- (1) Click on the [View Menu] icon and select [Show by Hardware View]. The Components tree will display the components in a hardware resource hierarchy.

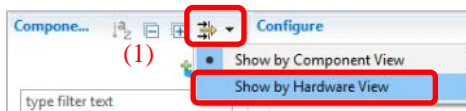


Figure 4-12 Switch to [Show by Hardware View]

- (2) Double-click on a hardware resource node (e.g. S12AD1 under 12-bit A/D converter) to open the [New Component] dialog box.
- (3) Select a component from the list (e.g. Single Scan Mode S12AD) in the [Software Component Selection] page.
- (4) Click the [Next].

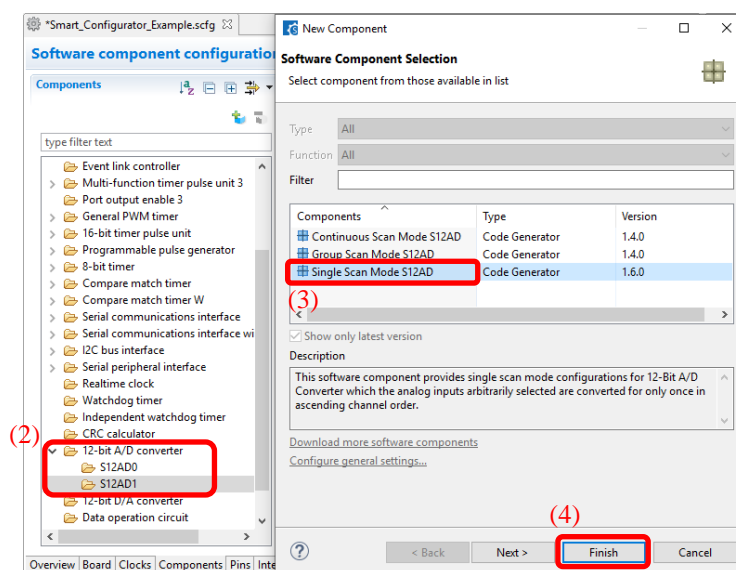


Figure 4-13 Adding CG Components form the Hardware View

- (5) Specify an appropriate configuration name in the [Add new configuration for selected component] page or use the default name (e.g. Config_S12AD1).
- (6) Select a hardware resource or use the default resource (e.g. S12AD1).
- (7) Click on [Finish]. The component is added to the component tree.

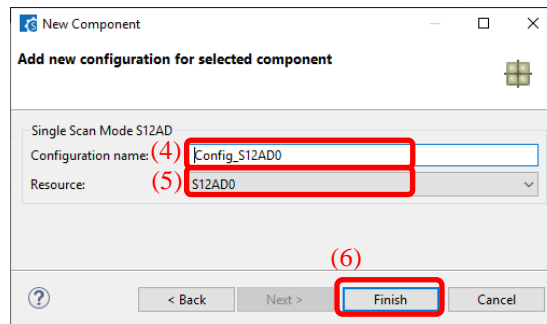


Figure 4-14 Add New Configuration for Selected Component (e.g. S12AD1)

4.3.3 Removing a component

Follow the procedure below to removing a software component.

- (1) Select a software component from the Components tree.
- (2) Click on the [Remove component] icon. The selected software is removed from the component tree. The selected software component will be removed from the Components tree.

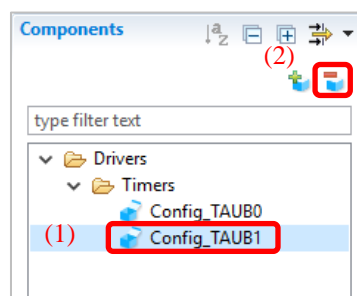


Figure 4-15 Removing a Component

4.3.4 Component configuration settings

Follow the procedure below to setting the component configuration.

- (1) Click the component in the component tree. (e.g. Config_S12AD0).
- (2) Configure the driver in the [Configure] panel to the right of the Components tree. The Figure 4-16 is an example.
 - a. Select AN000.
 - b. Select [A/D conversion start trigger pin] under [Conversion start trigger setting].
 - c. Click on [Advance setting] to expand the view.
 - d. Select [Discharge] for [Charge setting].

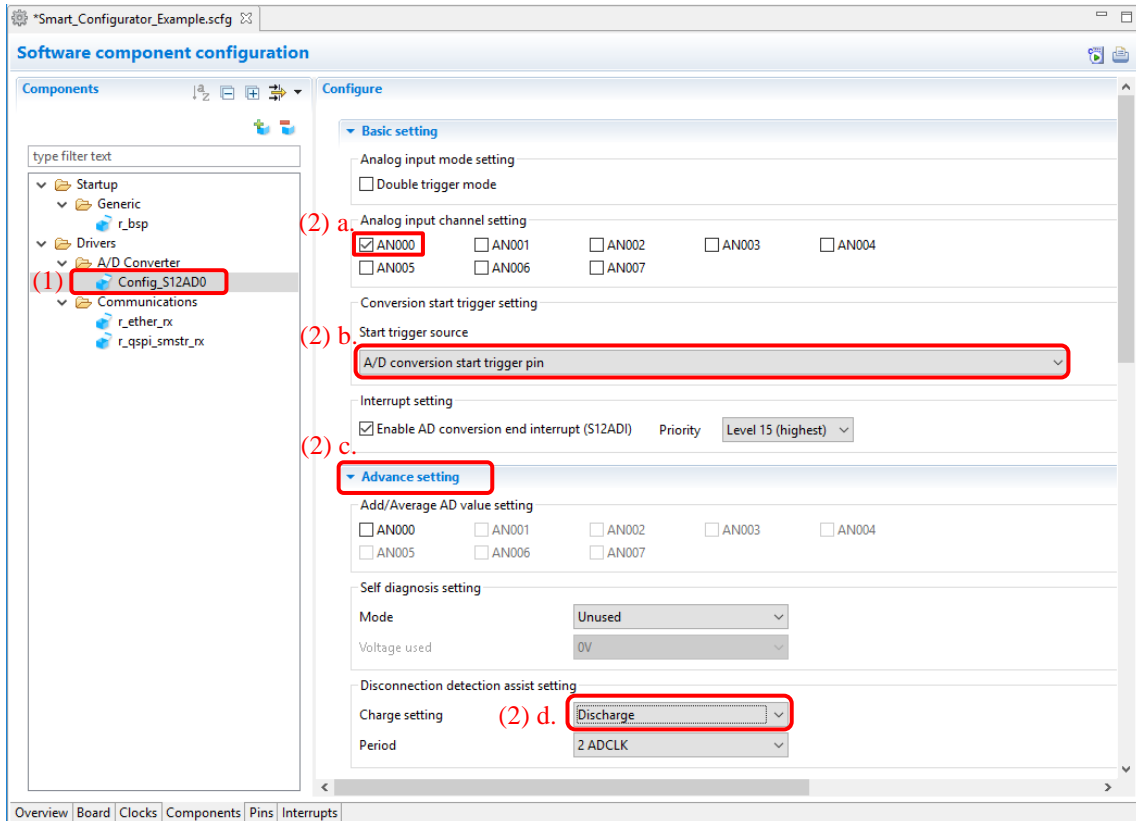


Figure 4-16 Component Configuration Settings

The code generation of the component is set to enabled by default.

Right click on the component and click [Generate code], it changes to [Generate code] and no code is generated.

Clicking [Generate code] will change to [Generate code] and generate code.

4.3.5 Component resource change

You can change the resource of the component (e.g. change from S12AD0 to S12AD1). Compatible configurations can be migrated from the current resource to the newly selected resource.

Follow the procedure below to change the resource.

- (1) Right-click on a component (e.g. Config_S12AD0).
- (2) Select [Change resource] from the context menu.

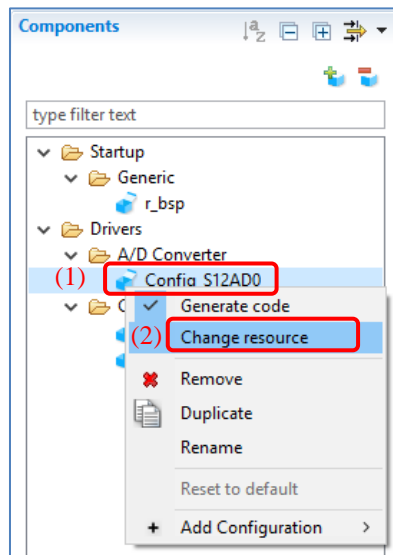


Figure 4-17 Resource Change

- (3) Select a new resource in the [Resource Selection] dialog box (e.g. S12AD1).
- (4) The [Next] button will be active; click on it.

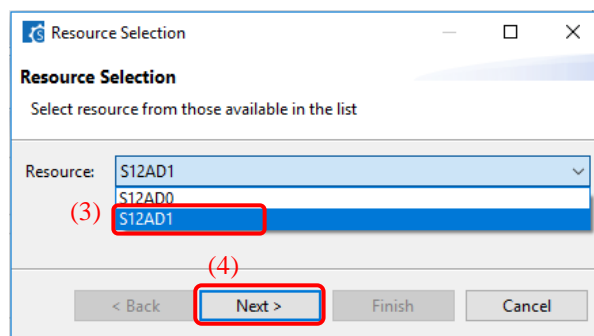


Figure 4-18 Select a New Resource

- (5) The configuration information is displayed on the [Configuration setting selection] page of the [Select Resource] dialog.
- (6) Check the portability of the settings.
- (7) Select whether to use the listed or default settings.
- (8) Click on [Finish].

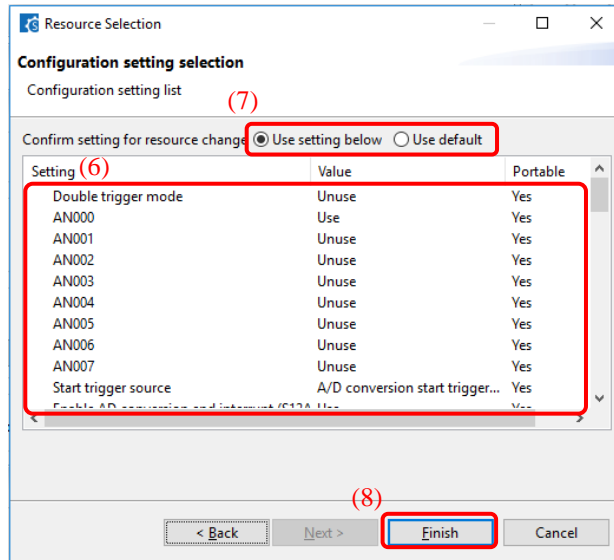


Figure 4-19 Confirm New Resource Settings

The resource is automatically changed (e.g. changed from S12ADI0 to S12ADI1).

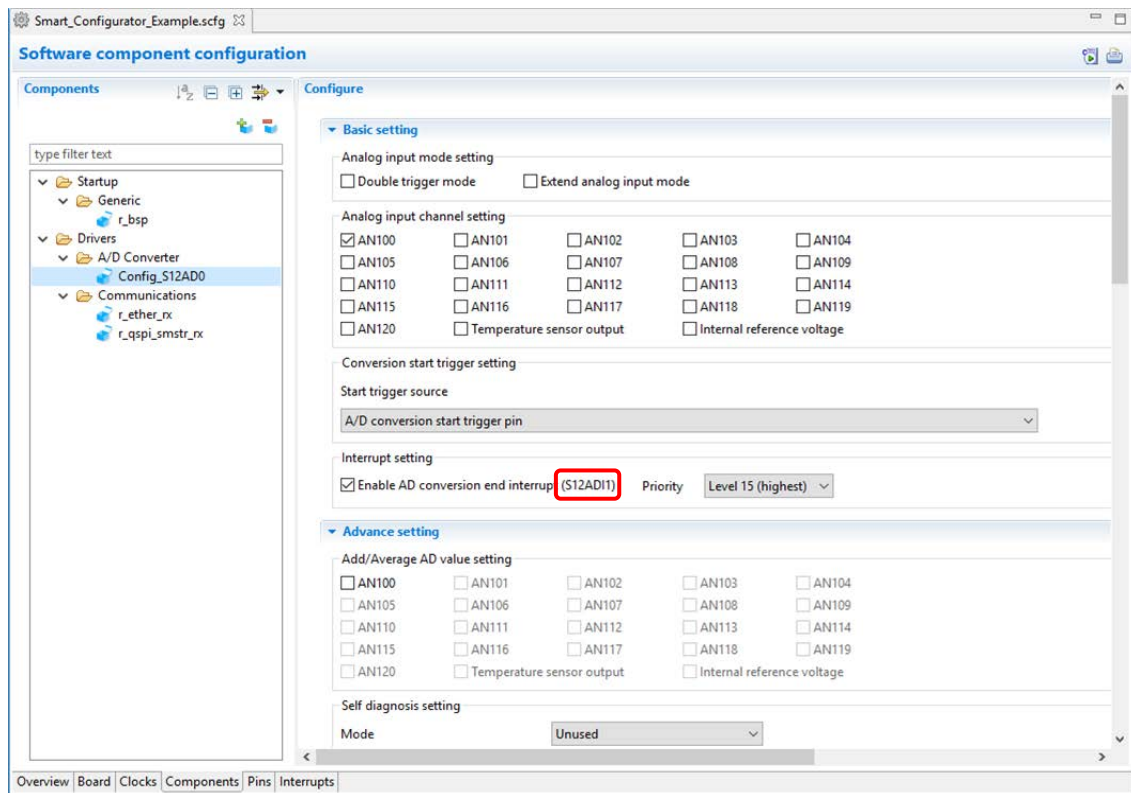


Figure 4-20 Resource Changed Automatically

To change the configuration name, follow the procedure below.

- (9) Right-click on the component.
- (10) Select [Rename] to rename the configuration (e.g. change Config_S12AD0 to Config_S12AD1).

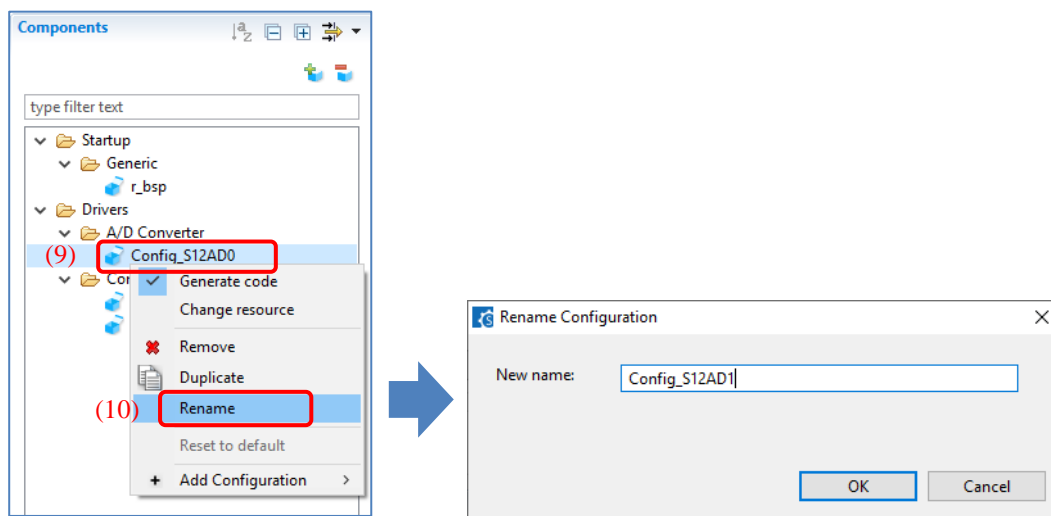


Figure 4-21 Renaming the Configuration

4.4 Pin settings

The [Pins] page is used for allocating pin functions. You can switch the display by clicking on the [Pin Function] and [Pin Number] tabs. The [Pin Function] list shows the pin functions for each of the peripheral functions, and the [Pin Number] list shows all pins in order of pin number.

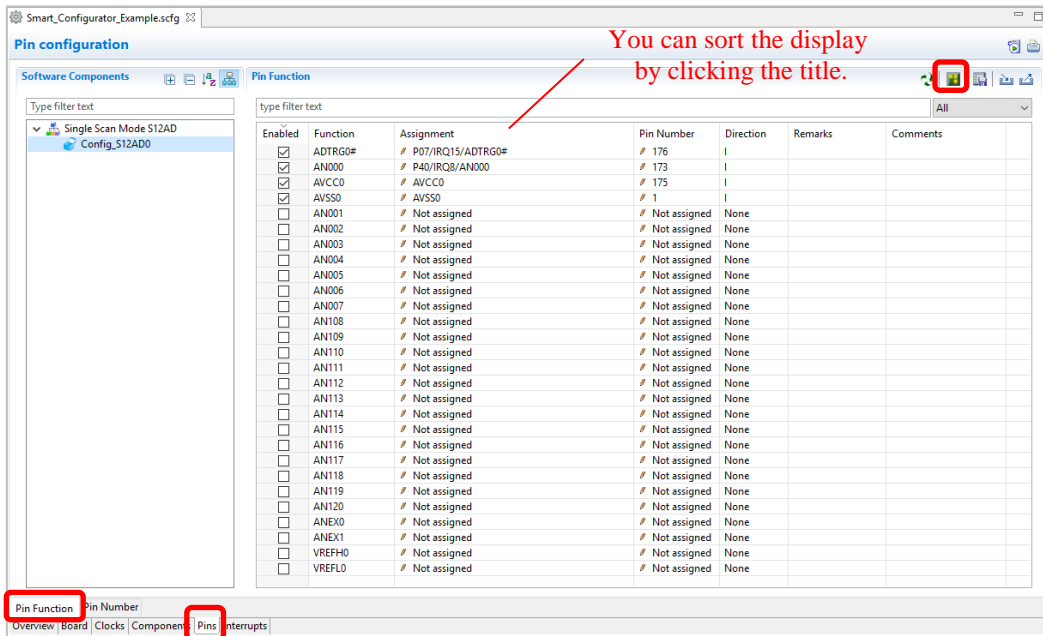


Figure 4-22 [Pins] Page ([Pin Function])

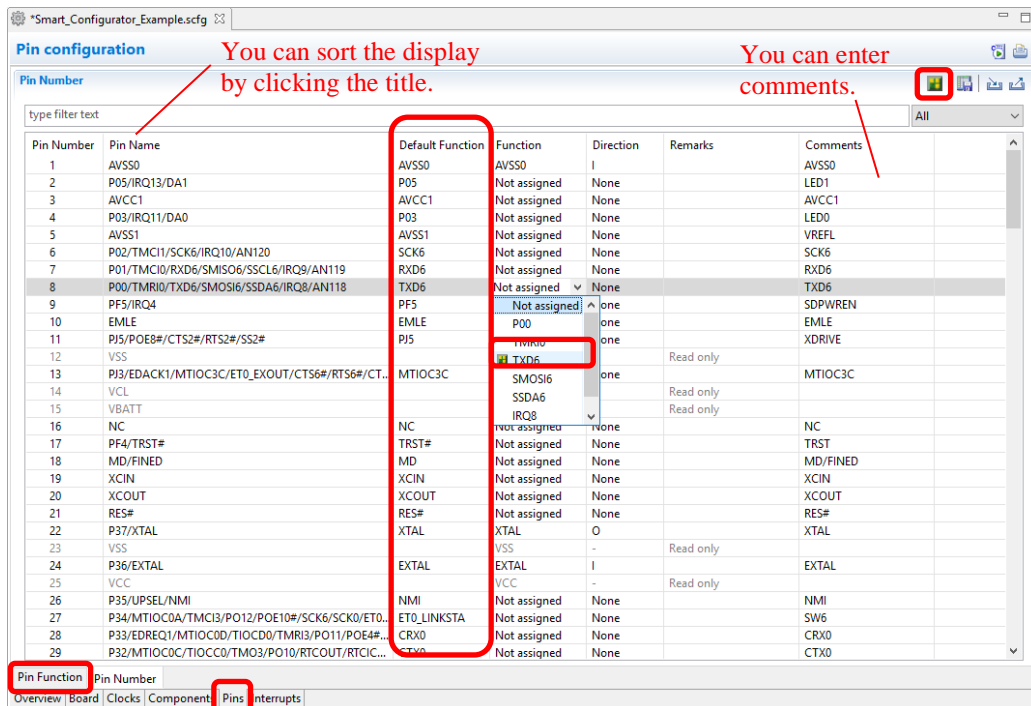





Figure 4-23 [Pins] Page ([Pin Number])

When you select a board on the [Board] page, the initial pin setting information of the board is displayed in [Default Function]. In addition, the [] icon displayed in the [Function] selection list indicates the initial pin function of the board.

4.4.1 Assign pins to resources

In the Pins page, assign pin to the resource used by the component. Pin assignment can be done in either [Pin Function] list or [Pin Number] list.

The procedure for pin assignment in the [Pin Function] list is described below.

- (1) Click on  (Show by Hardware Resource or Software Components) to switch to the software component view.
- (2) Select the target software component (e.g. Config_S12AD1).
- (3) Click the [Enabled] header to sort by pins used.
- (4) Pin assignment is performed with the [Assignment], [Pin Number] column, or  (Next group of pins for the selected resource) button.
 - (a) Click [Assignment] or [Pin Number] and assign a terminal from the list (e.g. change from P17 to P13).
 - (b) Click the  (Next group of pins for the selected resource) button and change the pin assignment. Each time you click, the pin with the function switches.

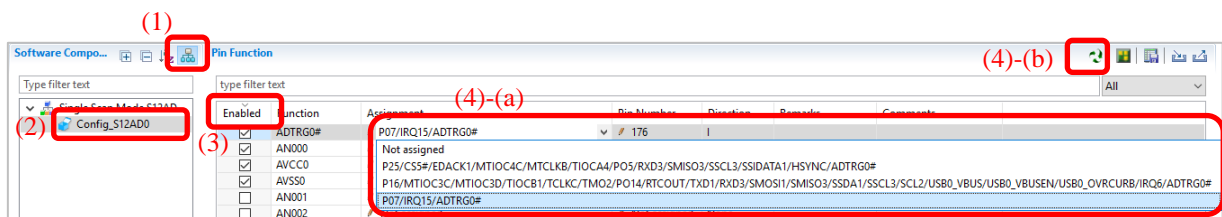


Figure 4-24 Pin Assignments in the [Pin function] List

When the component is set, the check box in the [Enabled] column is checked. Pin assignment is possible even when the component is not set. If pin assignment is done without component being set, we will display "No component is using this pin" in the [Remarks] column.

4.4.2 Pin setting using MCU package

Follow the procedure below to assign pins in the MCU Package view.

- (1) Zoom in to the view by clicking the [🔍 (Zoom in)] button or scrolling the view with the mouse wheel.
- (2) Right-click on the target pin.
- (3) Select the signal to be assigned to the pin.

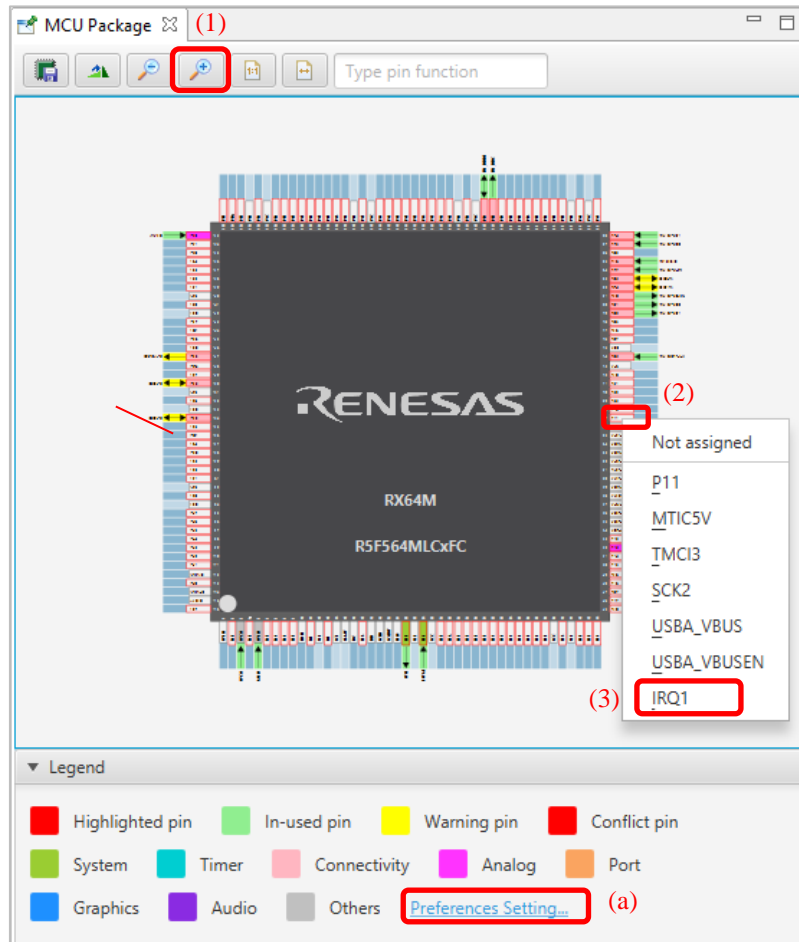



Figure 4-25 Assigning Pins Using the MCU Package View

- (a) The color of the pins can be customized through [Preference Setting...].

4.4.3 Export pin settings

You can export pin assignment settings in XML format. Exported files can be imported into projects of the same device family. Follow the procedure below to export the pin settings.

- (1) Click on the  (Export board setting) button on the [Pins] page.
- (2) In the [Export] dialog, enter the file name to export.

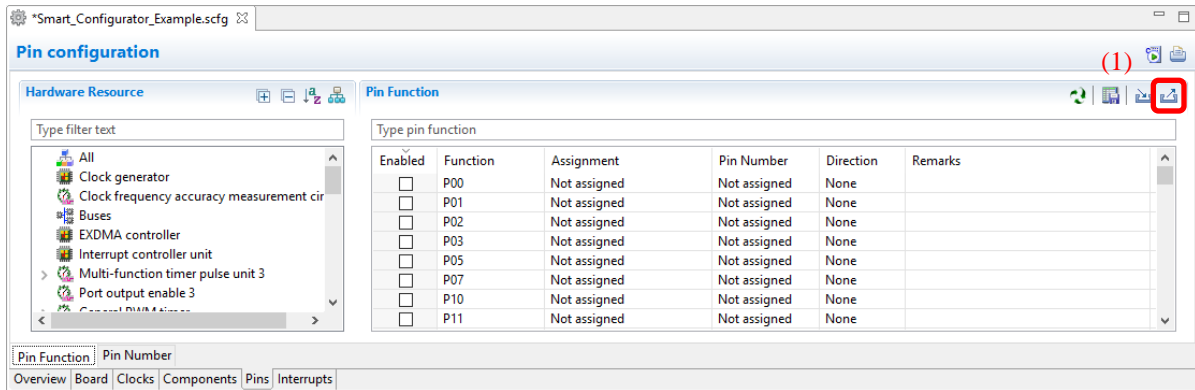




Figure 4-26 Export Pin Settings (XML format)

The Smart Configurator can also export the pin settings to a CSV file. Click on the  (Save the list to .csv file) button on the [Pins] page.

4.4.4 Import pin settings

You can import XML format files including pin assignment settings. When you import a file, the terminal assignment is reflected. Follow the procedure below to import the pin settings.

- (1) Click on the  (Import board setting) button on the [Pins] page.
- (2) In the [Import] dialog, enter the file name to import.

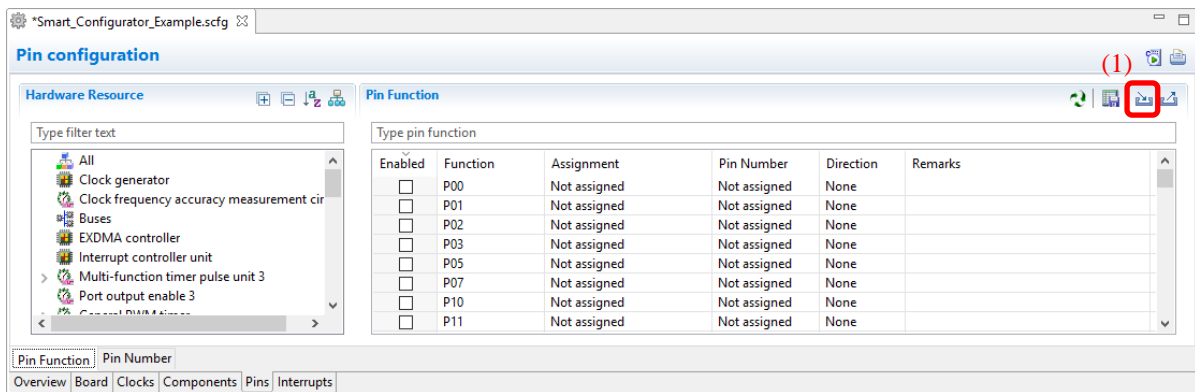


Figure 4-27 Import Pin Settings (XML format)

4.4.5 Pin setting using board pin configuration information

You can set the initial pin configuration of the board at once. The following describes the procedure for collective setting of pins.

- (1) Select [Default Board] in the MCU Package. (The initial pin configuration of the board can be referred.)
- (2) Open the [Pin Configuration] page and click the [Assign default board pins] button.
- (3) When [Assign default board pins] dialog opens, click [Select all].
- (4) Click [OK].

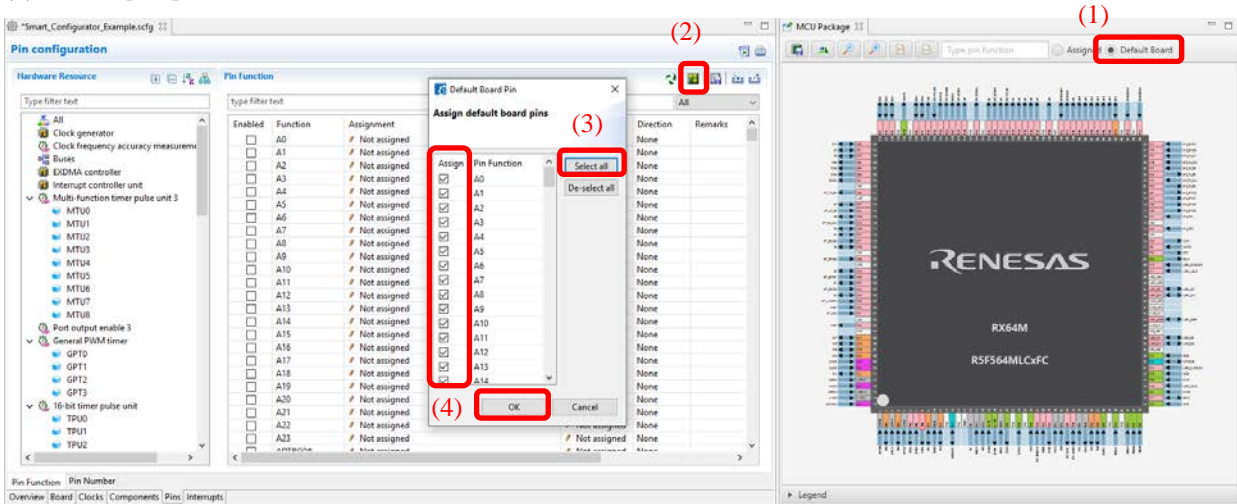


Figure 4-28 Setting for Initial Pin Configuration

If you do not set pin settings all at once, specify them individually in procedure (3).

4.4.6 Pin filter feature

By specifying the filter range on the [Pin Function] tab and [Pin Number] tab on the [Pins] page, you can refer to it more easily.

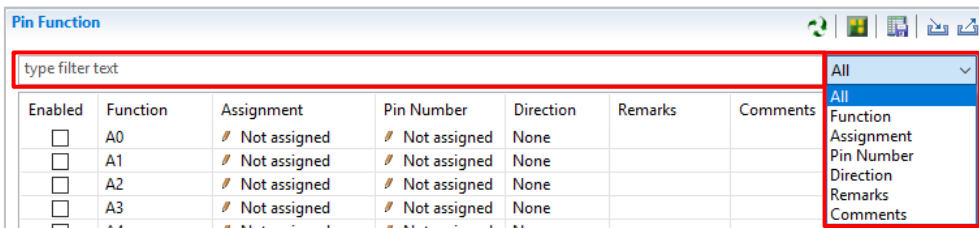


Figure 4-29 Filter for [Pin Function] Tab

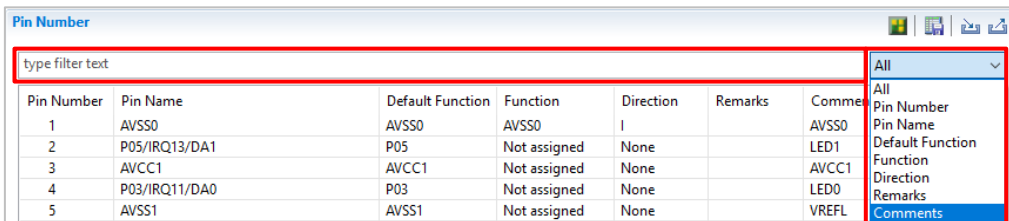


Figure 4-30 Filter for [Pin Number] Tab

4.5 Interrupt settings

Check and set the interrupts of the peripheral modules that have been selected on the [Components] page. The interrupts are displayed for each of the vector numbers. Set the interrupt priority levels, the source of the fast interrupt, or a dynamic interrupt vector number.

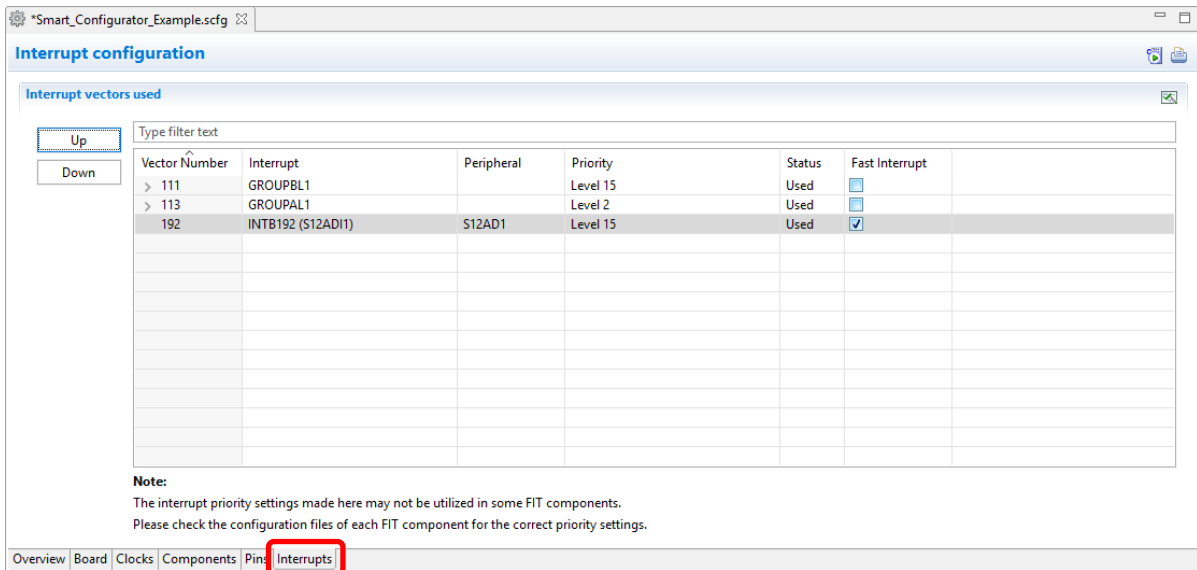


Figure 4-31 [Interrupts] Page

4.5.1 Changing the interrupt priority level and fast interrupt setting

When an interrupt is used in a configuration on the [Components] page, the status of the interrupt will be changed to "Used". To display the used interrupts only, click on the [Show used interrupts] button.

- (1) You can change the interrupt priority level on the [Interrupts] page.
- (2) To use an interrupt as a fast interrupt, tick the checkbox in the [Fast Interrupt] column. Only one interrupt can be specified as a fast interrupt among all interrupts and components used.

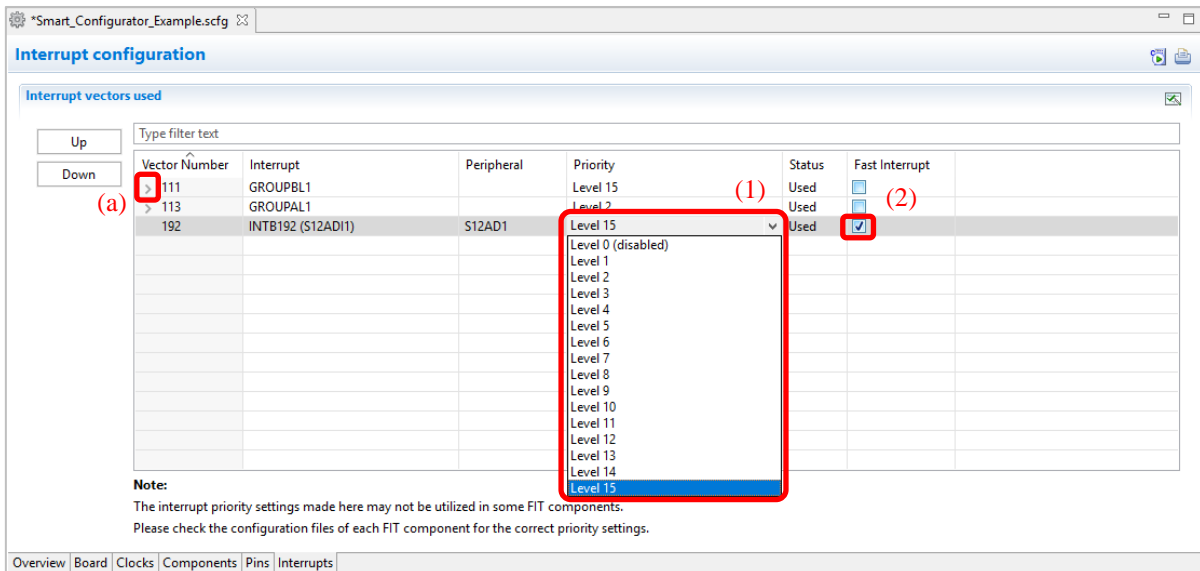


Figure 4-32 Interrupt Settings

- (a) Group interrupts are collapsed in the interrupt table. Click on the [> (Open)] button to expand the view and see the interrupts in the group interrupt list.

4.5.2 Changing the interrupt priority level and fast interrupt setting

The [Interrupt configuration] page enables you to change the vector numbers of software configurable interrupts A and B.

- (1) Select a desired software configurable interrupt.
- (2) The [Up] and [Down] buttons will be enabled. Click on a button to change the vector number.

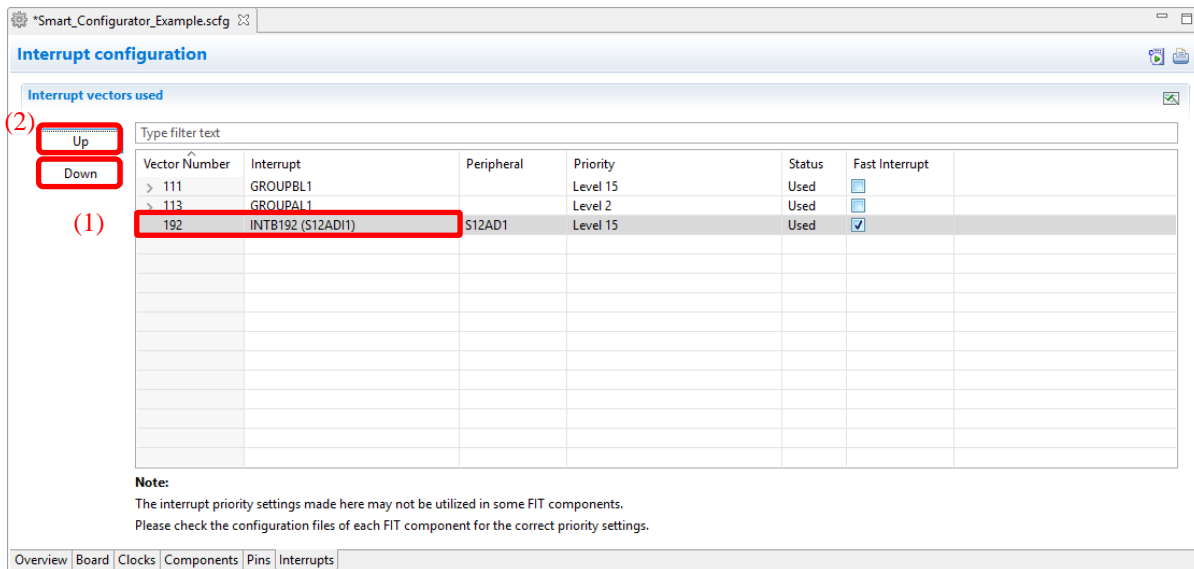


Figure 4-33 Changing the Vector Number of Software Configurable Interrupt A or B

5. Managing Conflicts

Adding components, setting pins and interrupts may cause problems related to resource mismatch. This information will be displayed in the **Configuration Problems** view. User can refer to the information displayed to fix the conflict issues.

5.1 Resource conflicts

When two software components are configured to use the same resource (e.g. S12AD1), an error mark (❌) will be displayed in the [Components tree].

The [Configuration Problems view] will display messages on peripheral conflicts to inform the user in which software configurations peripheral conflicts have been detected.

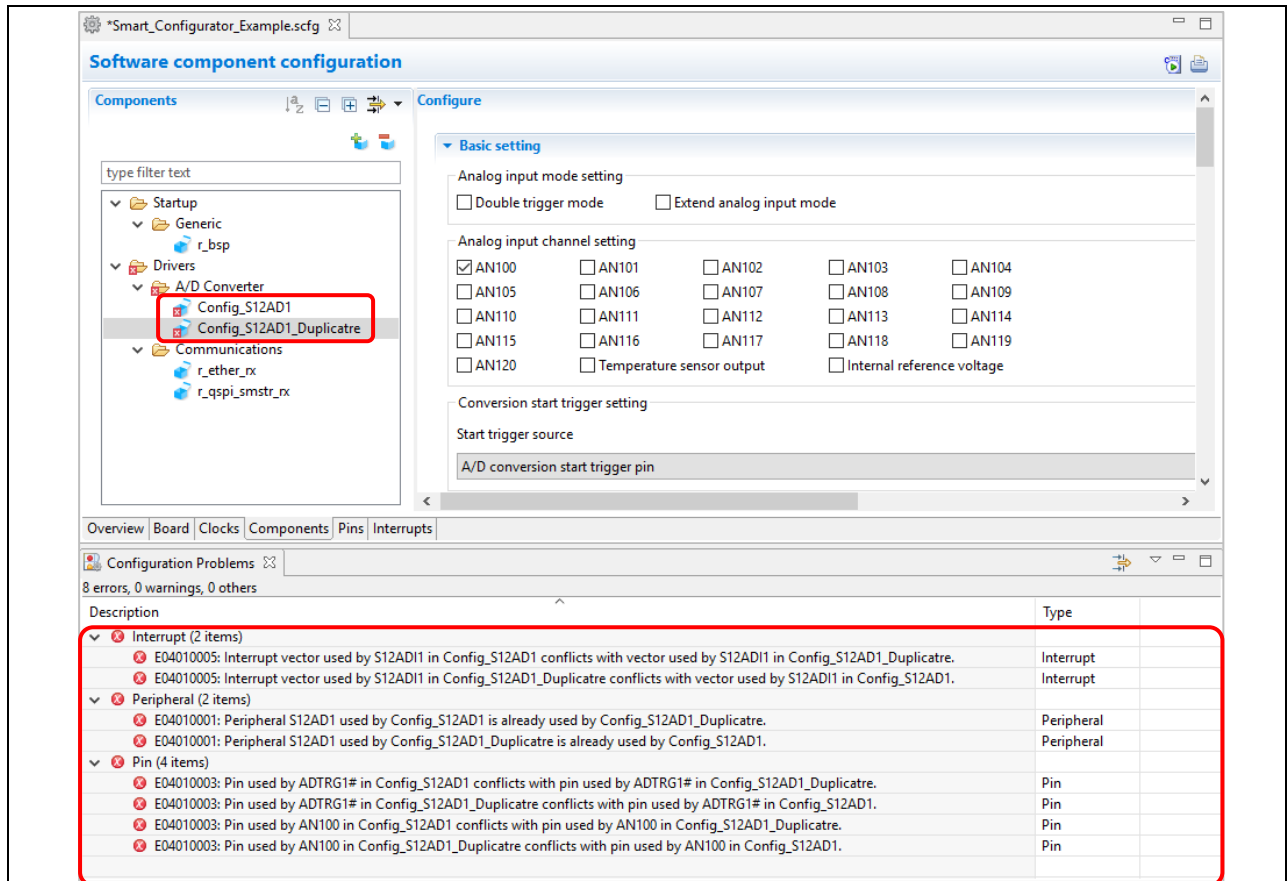



Figure 5-1 Resource Conflicts

5.2 Resolving pin conflicts

When multiple pin functions are assigned to the same pin, an error mark  is displayed in the tree and [Pin Function] list on the [Pins] page.

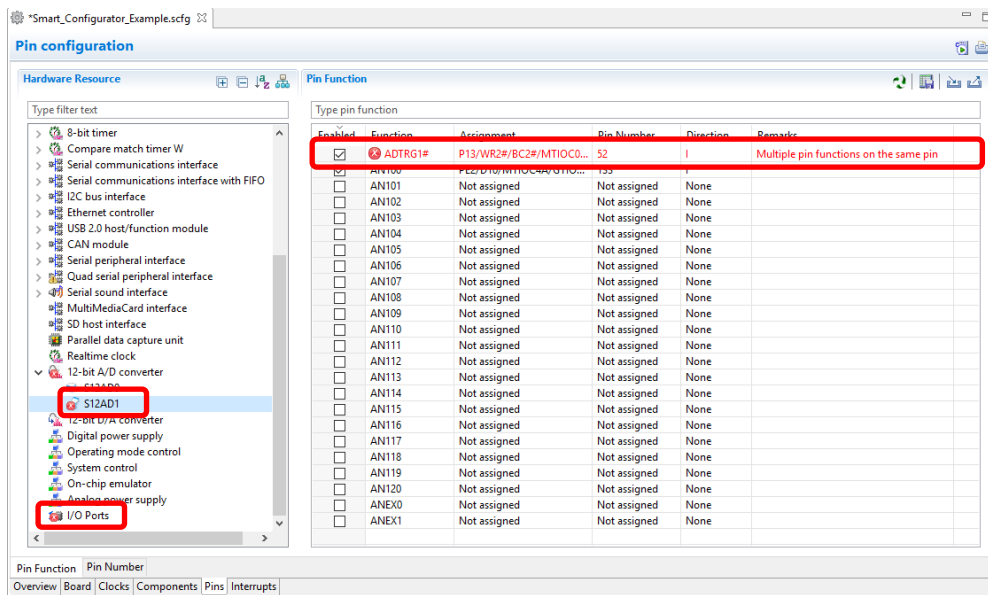


Figure 5-2 Pin Conflicts

The detailed information regarding conflicts is displayed in the [Configuration Problems view].

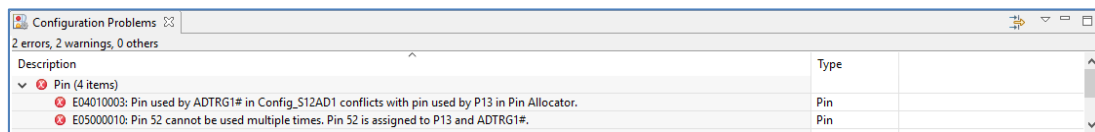


Figure 5-3 Pin Conflict Message

To resolve a conflict, right-click on the node with an error mark on the tree and select [Resolve conflict].

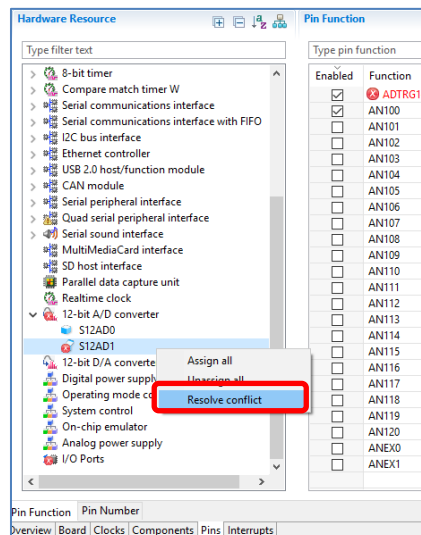



Figure 5-4 Resolving Pin Conflicts

The pins of the selected node will be re-assigned to other pins.

6. Generating Source Code

6.1 Generating Source Code File

Output a source file for the configured details by clicking on the  (Generate Code) button in the Smart Configurator view.

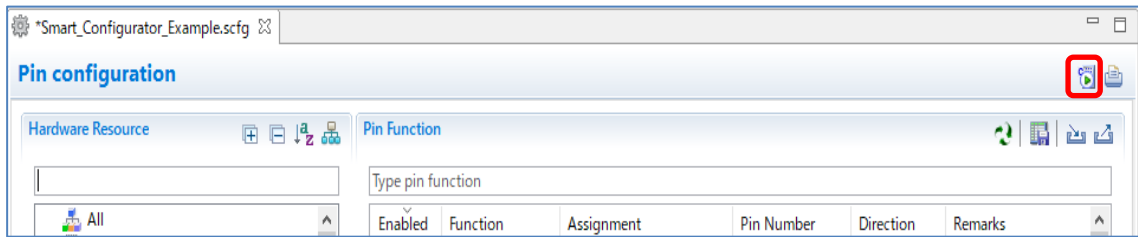


Figure 6-1 Generating a Source File

The Smart Configurator generates a source file in <ConfigurationFileDir>\src\smc_gen. If your Smart Configurator has already generated a file, a backup copy of that file is also generated (refer to the section 6.6, Backing up Generated Source Code).

6.2 Configuration of Generated Files and File Names

Figure 6-2, Configuration of Generated Files and File Names, shows the folders and files output by the Smart Configurator. "ConfigName" indicates the configuration name set in the component.

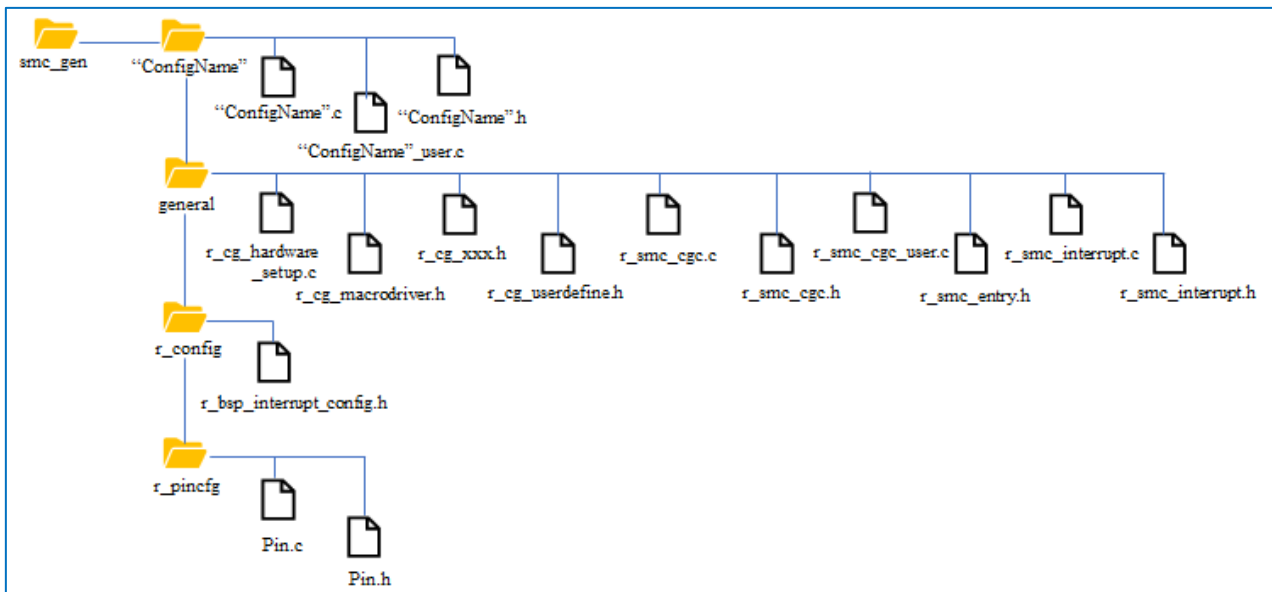


Figure 6-2 Configuration of Generated Files and File Names

Table 6-1 Description of Generated File

Folder	File	Description
{ConfigName}	-	This folder is generated for the added component. API functions in this folder are named after the <i>ConfigName</i> (configuration name).
	<i>{ConfigName}.c</i>	This file contains functions to initialize driver (<i>R_ConfigName_Create</i>) and perform operations that are driver-specific, e.g. start (<i>R_ConfigName_Start</i>) and stop (<i>R_ConfigName_Stop</i>).
	<i>{ConfigName}_user.c</i>	This file contains interrupt service routines and functions for user to add code after the driver initialization (<i>R_ConfigName_Create</i>). User can add codes and functions in the dedicated user code areas.
	<i>{ConfigName}.h</i>	This is header file for <i>{ConfigName}.c</i> and <i>{ConfigName}_user.c</i> .
general	-	This folder is always generated. It contains header files and source files commonly used by drivers of the same peripheral function.
	<i>r_cg_hardware_setup.c</i>	This file is always generated. It contains <i>R_Systeminit</i> that calls all driver initialization functions with the name <i>R_ConfigName_Create</i> . <i>R_Systeminit</i> also calls the functions for initializing clocks other than the clock source, fast interrupt, and group interrupts.
	<i>r_cg_macrodriver.h</i>	This file is always generated. This header file contains common macro definitions used in drivers.
	<i>r_cg_xxx.h</i> ^(Note*1)	These files are always generated. The files contain macro definitions for setting SFR registers.
	<i>r_cg_userdefine.h</i>	This file is always generated. User can add macro definitions in the dedicated user code areas.
	<i>r_smc_cgc.c</i>	This file is always generated. It contains the initialization of clock sources other than the clock source selected in the [Clocks] page.
	<i>r_smc_cgc.h</i>	This file is always generated. This header file contains macro definitions to initialize clocks other than the selected clock source.
	<i>r_smc_cgc_user.c</i>	This file contains functions to be added to <i>R_CGC_Create</i> after the CGC initialization. User can add codes and functions in the dedicated user code areas.
	<i>r_smc_entry.h</i>	This file is always generated. This file includes the header files of CG drivers that are added to the project. When using functions of CG drivers in source files added by user, including this file is necessary.
	<i>r_smc_interrupt.c</i>	This file is always generated. It contains fast interrupt and group interrupt initialization (depending on hardware specification).
	<i>r_smc_interrupt.h</i>	This file is always generated. It contains macro definitions for fast interrupt and group interrupt initialization. It also contains the priority level of all interrupts that are configured in the [Interrupts] tabbed page. User can use these macro definitions in application codes.
r_config	<i>r_bsp_interrupt_config.h</i>	This file is always generated. It contains mapping of the software configurable interrupts A and B (depending on hardware specification).
r_pincfg	<i>Pin.c</i>	This file is always generated. It is a reference of pin function initialization for all peripherals configured in the [Pins] tabbed page (except I/O Ports).
	<i>Pin.h</i>	This file is always generated. It contains the function prototypes of pin settings in <i>Pin.c</i>

Note *1: xxx is the name of a component.

6.3 Initializing Clocks

Configurations of clock source in [Clocks] page are generated in \src\smc_gen\r_config folder.

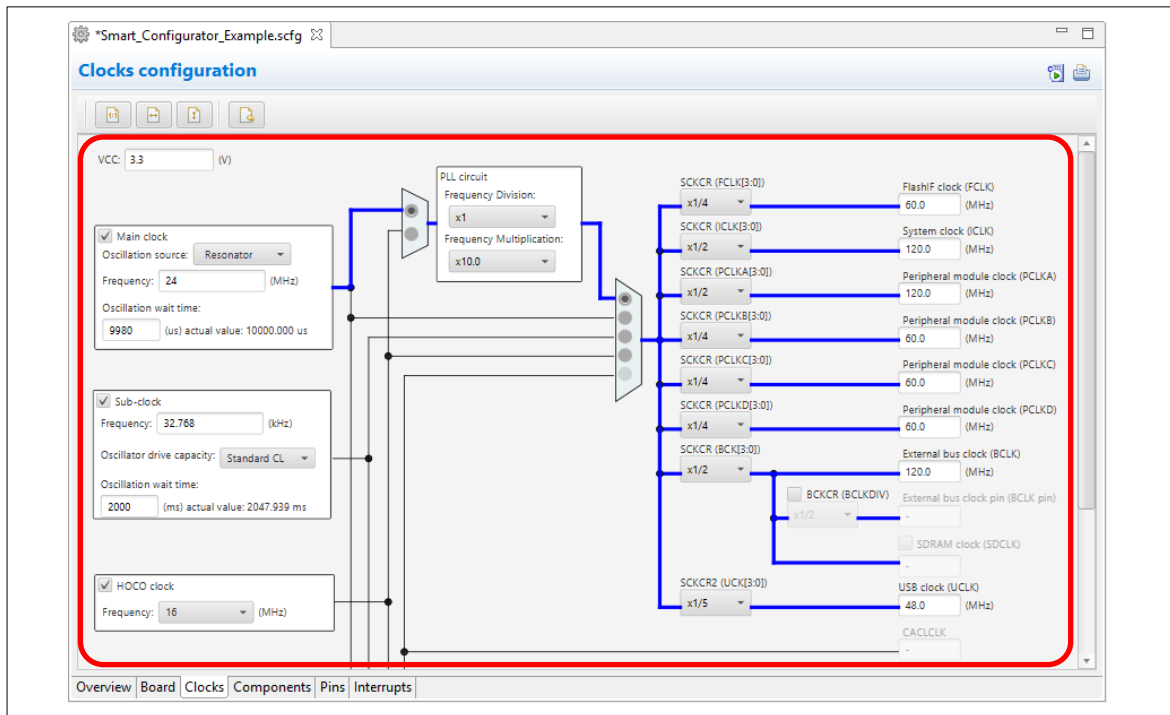


Figure 6-3 Clocks Source Configuration

Table 6-2 Clock Source File Description

Folder	File	Macros/Functions	Description
general	<i>r_cg_cgc.c</i>	<i>R_CGC_Create</i>	This API function initializes clocks other than the selected clock source. <i>R_Systeminit</i> in <i>r_cg_hardware_setup.c</i> will call this function before entering <i>main()</i> function.
	<i>r_cg_cgc.h</i>	Macros related to clocks	These macros are for clock initialization in <i>R_CGC_Create</i> .
	<i>r_cg_cgc_user.c</i>	<i>R_CGC_Create_UserInit</i>	This API function is used to add code to <i>R_CGC_Create</i> after the CGC initialization.

6.4 Initializing Pins

Pin configuration settings are generated by the component into source files as shown in (1) and (2) below.

- (1) Pins initialization for drivers with { *ConfigName* }

The pin function is initialized with *R_ConfigName_Create* of `\src\smc_gen\{ConfigName}\{ConfigName}.c`.

Table 6-3 File to Initialize Pins

Folder	File	Function	Description
{ ConfigName }	<i>{ConfigName}.c</i>	<i>R_ConfigName_Create</i>	This API function initializes pins used by this component. <i>R_Systeminit</i> in <i>r_cg_hardware_setup.c</i> will call this function before entering <i>main()</i> function.

- (2) Reference pins initialization codes

Refer to *Pin.c* in the `\src\smc_gen\r_pincfg` folder for the initialization code of all pin functions set on the [Pins] page (except I/O ports).

Table 6-4 Reference File for Initialization of All Pins

Folder	File	Function	Description
r_pincfg	<i>Pin.c</i>	<i>R_Pins_Create</i>	This function contains the initialization codes of all pins function configured at [Pins] page except I/O ports.

6.5 Initializing Interrupts

Configurations in [Interrupt] page are generated in few source files.

Vector Number	Interrupt	Peripheral	Priority	Status	Fast Interrupt
111	GROUPBL1		Level 15 (1)	Used	<input type="checkbox"/>
21	S12CMPH1	S12AD1		Used	
113	GROUPAL1		Level 2	Used	<input type="checkbox"/>
192 (3)	INTB192 (S12AD11)	S12AD1 (2)	Level 15	Used	<input checked="" type="checkbox"/> (4)

Figure 6-4 Interrupt Configuration

Table 6-5 Interrupt Generation File Description


No	Item	Folder	File	Description
(1)	Priority	general	<i>r_smc_interrupt.c</i>	This interrupt priority level setting is for group interrupts (Note2). It is initialized in <i>R_Interrupt_Create</i> of this file. <i>R_Systeminit</i> in <i>r_cg_hardware_setup.c</i> will call this function before entering <i>main()</i> function.
(2)	Priority	{ConfigName}	{ConfigName}.c	This interrupt priority level setting is for normal interrupts and software configurable interrupts A and B (Note2). It is initialized in <i>R_ConfigName_Create</i> of this file. <i>R_Systeminit</i> in <i>r_cg_hardware_setup.c</i> will call this function before entering <i>main()</i> function.
(1) (2)	Priority	general	<i>r_smc_interrupt.h</i>	Priority level of all interrupts configured in the [Interrupts] tabbed page is defined in this file. User can use these macro definitions in the application codes.
(3)	Vector Number	r_config	<i>r_bsp_interrupt_config.h</i>	Vector number of software configurable interrupts A and B (Note2) in the [Interrupts] tabbed page will be mapped in this file and handled by <i>r_bsp</i> .
(4)	Fast Interrupt	general	<i>r_smc_interrupt.c</i>	Fast interrupt setting will be initialized in <i>R_Interrupt_Create</i> of this file. <i>R_Systeminit</i> in <i>r_cg_hardware_setup.c</i> will call this function before entering <i>main()</i> function.
			<i>r_smc_interrupt.h</i>	Priority level of all interrupts configured in the [Interrupts] tabbed page is defined in this file. User can use these macro definitions in the application codes.

Note *2: The type of interrupt depends on hardware specifications.

6.6 Backing up Generated Source Code

The smart configurator has a source code backup function.

<ConfigurationFileDir>\trash\<Date-and-Time>

The Smart Configurator generates a backup folder for the previously generated source code when new code is generated by clicking on [ <Date-and-Time> indicates the date and time when the backup folder is created after code generation.

7. Loading generated files in Integrated development environment

Load source code outputted by Smart Configurator on Integrated Development Environment Platform.

7.1 Loading in IAR Embedded Workbench

When IAR environment is selected for the compiler to be used, Smart Configurator also outputs the project connection file (.ipcf) together with the source file. The project connection file contains source file registration information. It is not necessary for the user to add or delete source files after configuration change in the Smart Configurator.

The usage procedure is as follows.

- (1) Select [Add Project Connection..] from the [Project] menu of IAR Embedded Workbench.
- (2) The [Add Project Connection] dialog is displayed. Select [IAR Project Connection] and click [OK].
- (3) In the [Select IAR Project Connection File] dialog box, browse to the folder where the configuration file is saved, select the project connection file (.ipcf), and click the [Open] button.
- (4) The source file output by the Smart Configurator is added to the workspace.

8. Creating User Programs

Create a user program in the IDE. This chapter describes how to add custom code to the source file generated by the SC.

8.1 Adding Custom Code in the Case of Code Generator

When [Code Generator] is selected as the component type, if files which have the same name already exist, new code will be merged only with the existing code that is between the comments below.

```
/* Start user code for xxxx. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
```

In the case of [Code Generator], three files are generated for each of the specified peripheral functions. The file names are "Config_XXX.h", "Config_XXX.c", and "Config_XXX_user.c" as the default, with "XXX" representing the name of the peripheral module. For example, "XXX" will be "CMT3" for the compare-match timer (resource CMT3). The comments to indicate where to add custom code are at the start and end of each of the three files. Comments to indicate where to add user code are also added to the interrupt function for the peripheral module corresponding to Config.XXX_user.c. The following examples are for CMT3 (Config_CMT3_user.c).

```

/*****
Pragma directive
*****/
/* Start user code for pragma. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */

/*****
Includes
*****/
#include "r_cg_macrodriver.h"
#include "r_cg_userdefine.h"
#include "Config_CMT3.h"
/* Start user code for include. 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_CMT3_Create_UserInit
* Description : This function adds user code after initializing the CMT3 channel
* Arguments : None
* Return Value : None
*****/

void R_Config_CMT3_Create_UserInit(void)
{
    /* Start user code for user init. Do not edit comment generated here */
    /* End user code. Do not edit comment generated here */
}

```

```
/******  
* Function Name: r_Config_CMT3_cmi3_interrupt  
* Description : This function is CMI3 interrupt service routine  
* Arguments : None  
* Return Value : None  
*****/  
  
#if FAST_INTERRUPT_VECTOR == VECT_PERIB_INTB129  
#pragma interrupt r_Config_CMT3_cmi3_interrupt(vect=VECT(PERIB,INTB129),fint)  
#else  
#pragma interrupt r_Config_CMT3_cmi3_interrupt(vect=VECT(PERIB,INTB129))  
#endif  
static void r_Config_CMT3_cmi3_interrupt(void)  
{  
    /* Start user code for r_Config_CMT3_cmi3_interrupt. Do not edit comment generated here */  
    /* End user code. Do not edit comment generated here */  
}  
  
/* Start user code for adding. Do not edit comment generated here */  
/* End user code. Do not edit comment generated here */
```


9. Generating Reports

The Smart Configurator can output the configuration information of the project to the report. Follow the procedure below to generate a report.

9.1 Report on Configuration

A report is output in response to clicking on the [📄 (Generate Report)] button in the Smart Configurator view.

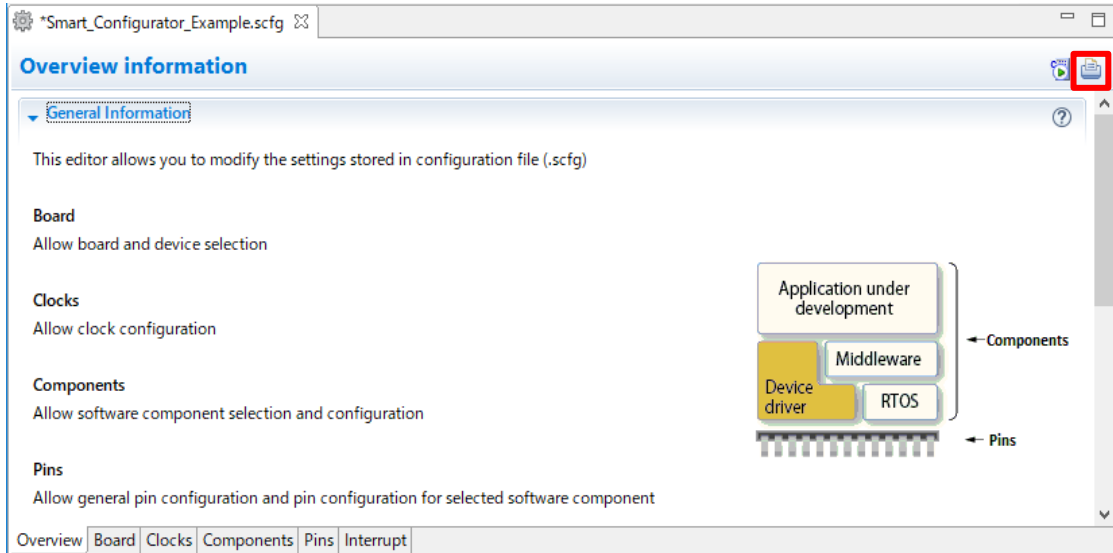


Figure 9-1 Output of a Report on the Configuration (as a Text File)

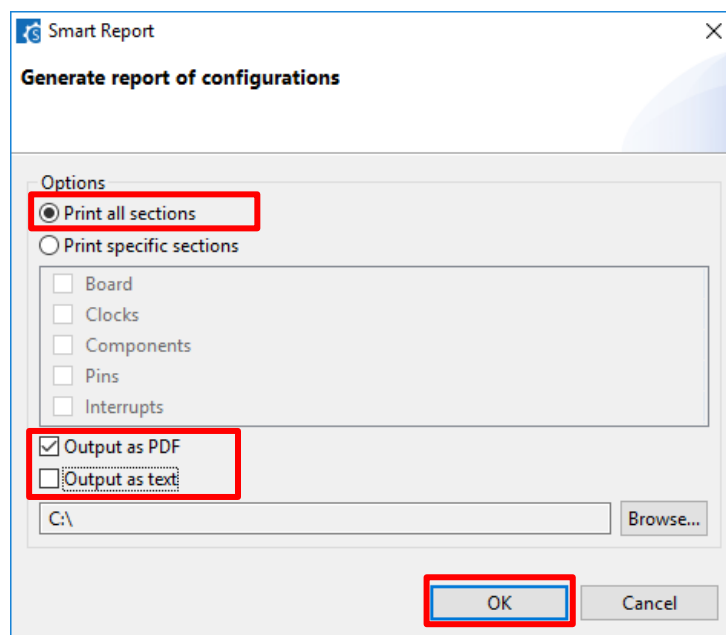


Figure 9-2 Dialog Box for Output of a Report

9.2 Configuration of Pin Function List and Pin Number List (in csv Format)

A list of the configuration of pin functions and pin numbers (whichever is selected at the time) is output in response to clicking on [Save] (Save the list to .csv file) on the [Pins] page of the Smart Configurator view.

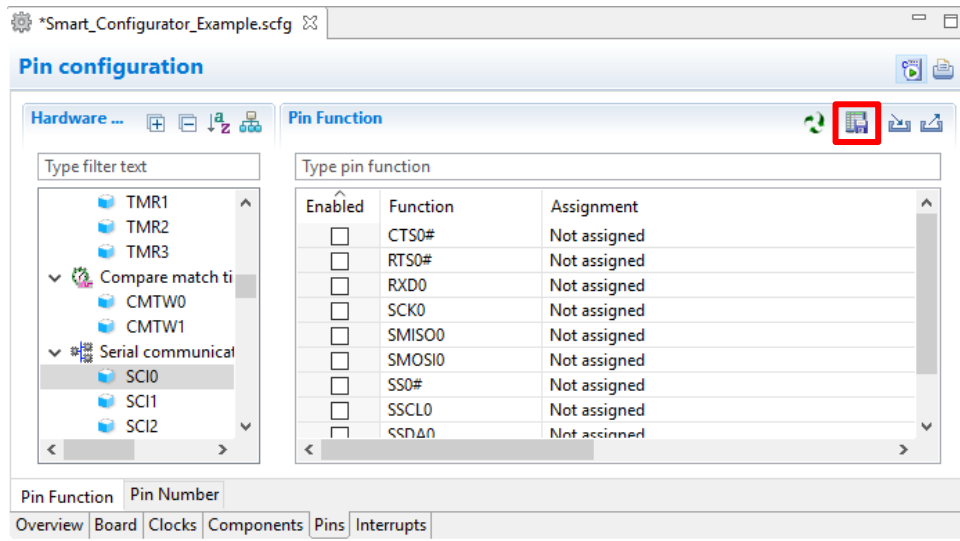


Figure 9-3 Output of a List of Pin Functions or Numbers (in csv Format)

9.3 Image of MCU Package (in png Format)

An image of the MCU package is output in response to clicking on the [Save] (Save Package View to external image file) button of the [MCU Package] view.

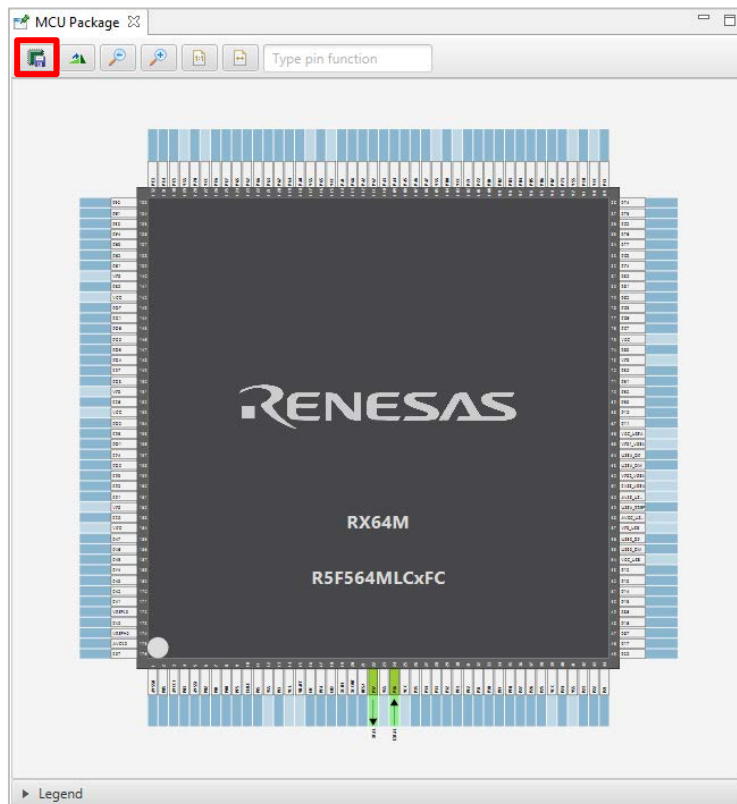


Figure 9-4 Outputting a Figure of MCU Package (in png Format)

10. Help

10.1 Help

Refer to the help system for detailed information on the Smart Configurator.

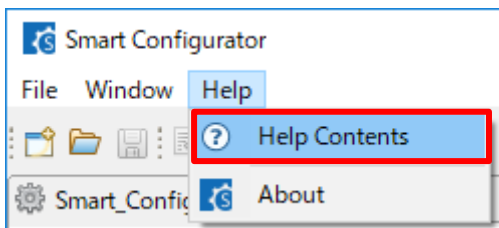


Figure 10-1 Help Menu

The help system can also be activated from the [Overview] page.

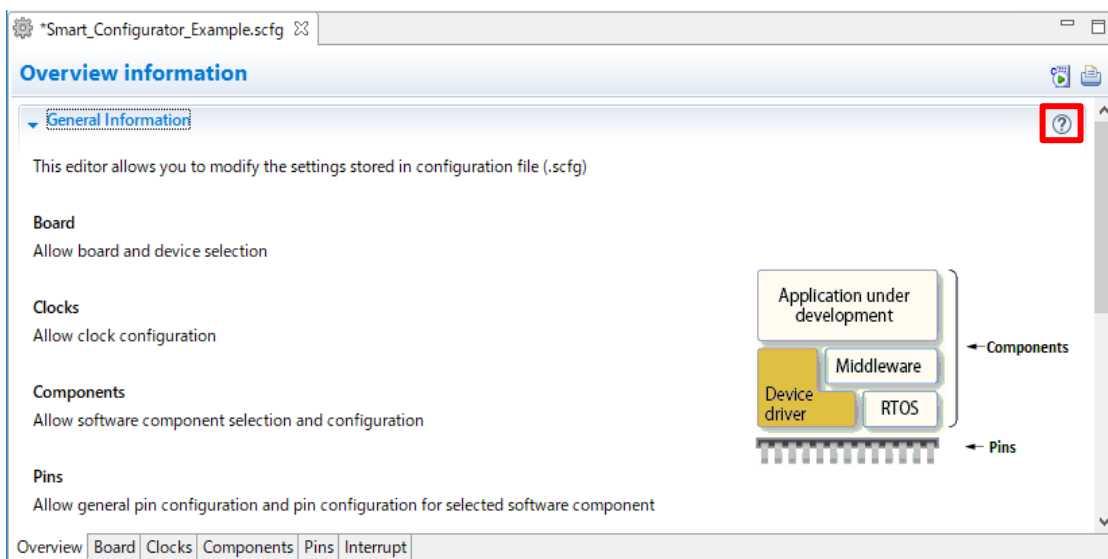


Figure 10-2 Quick Start

11. Documents for Reference

User's Manual: Hardware

Obtain the latest version of the manual from the web site of Renesas Electronics.

Technical Update/Technical News

Obtain the latest information from the web site of Renesas Electronics.

User's Manual: Development Environment

Obtain the latest version of the manual from each company web site.

Website and Support

Renesas Electronics Website

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

Inquiries

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

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

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jan 25, 2019	-	First edition issued

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

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

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity.

Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

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

3. Input of signal during power-off state

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

4. Handling of unused pins

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

5. Clock signals

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

6. Voltage application waveform at input pin

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

7. Prohibition of access to reserved addresses

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

8. Differences between products

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

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall 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.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; 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.
6. 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.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. 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.
8. 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.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall 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.
10. 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.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.
(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

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

Renesas Electronics Corporation
TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

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

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700

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

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

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

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

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

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

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

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338