RENESAS APPLICATION NOTE

RX Smart Configurator

User’s Guide: IAREW

R20AN0535EJ0110
Rev.1.10
Sep 09, 2019

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.

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

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 three functions to support the embedding of drivers from Renesas in your systems: importing middleware in the form of Firmware Integration Technology (FIT) modules, generating driver code and making pin settings.

1.3 Software Components

The Smart Configurator supports two types of software components: Code Generator (CG) and Firmware Integration Technology (FIT). Drivers and middleware supported by each software type are as follows.

- Basic drivers:
  - CG drivers (CMT, A/D Converter, SCI, etc.)
  - FIT modules (CMT, DTC, DMAC, RSPI, SCIFA, etc.)
- Middleware:
  - FIT modules (USB, Ethernet, Flash Memory (programming the on-chip flash memory), etc.)

The basic driver is a control program for peripheral functions of microcomputer such as CMT, A/D converter, SCI, etc.
It is convenient to embed a software component (CG driver or FIT module) using code generation function.
In addition, FIT modules can be embedded for using middleware such as USB, Ethernet, and Flash memory (programming the on-chip flash memory) as software components.

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.

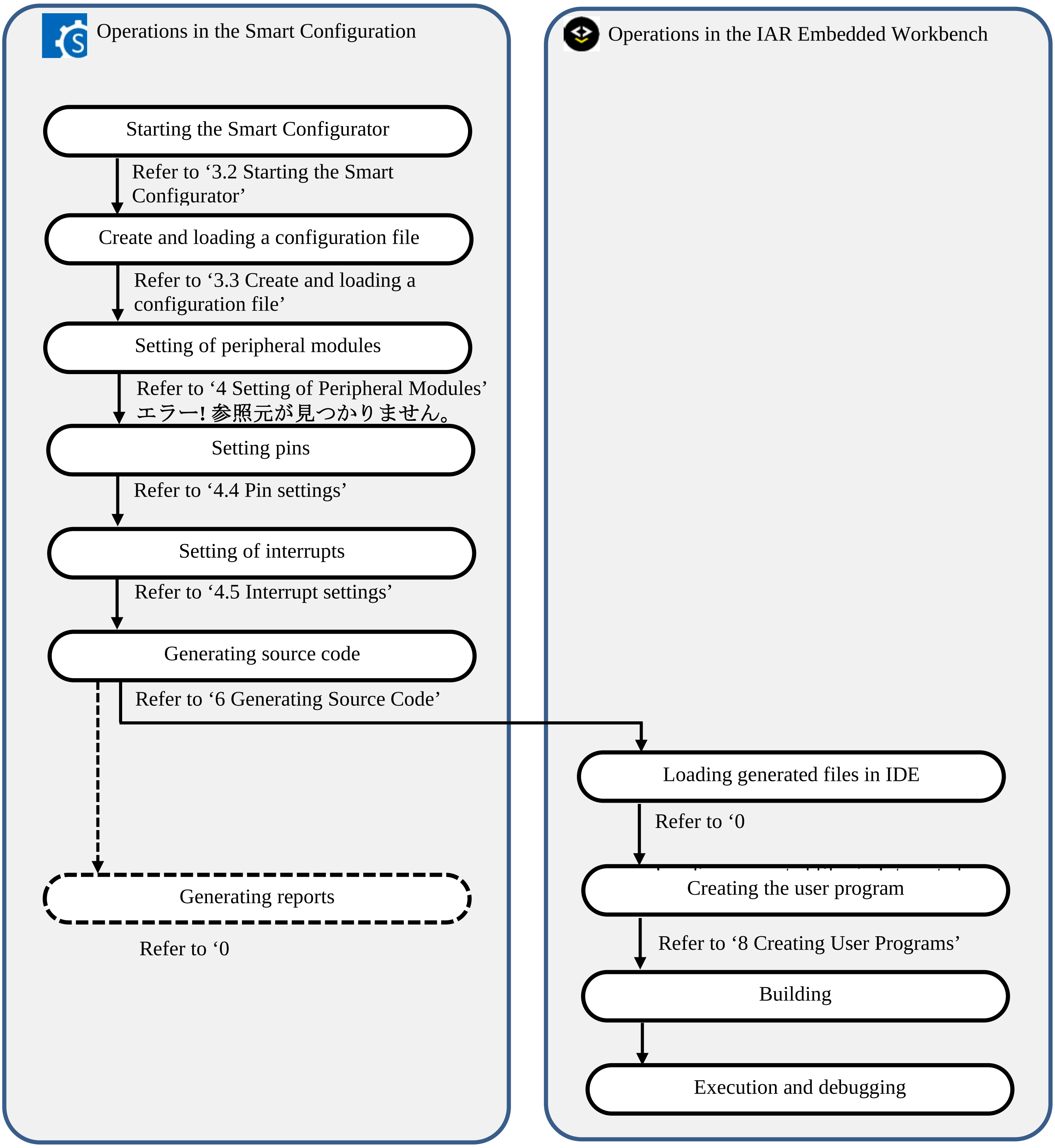
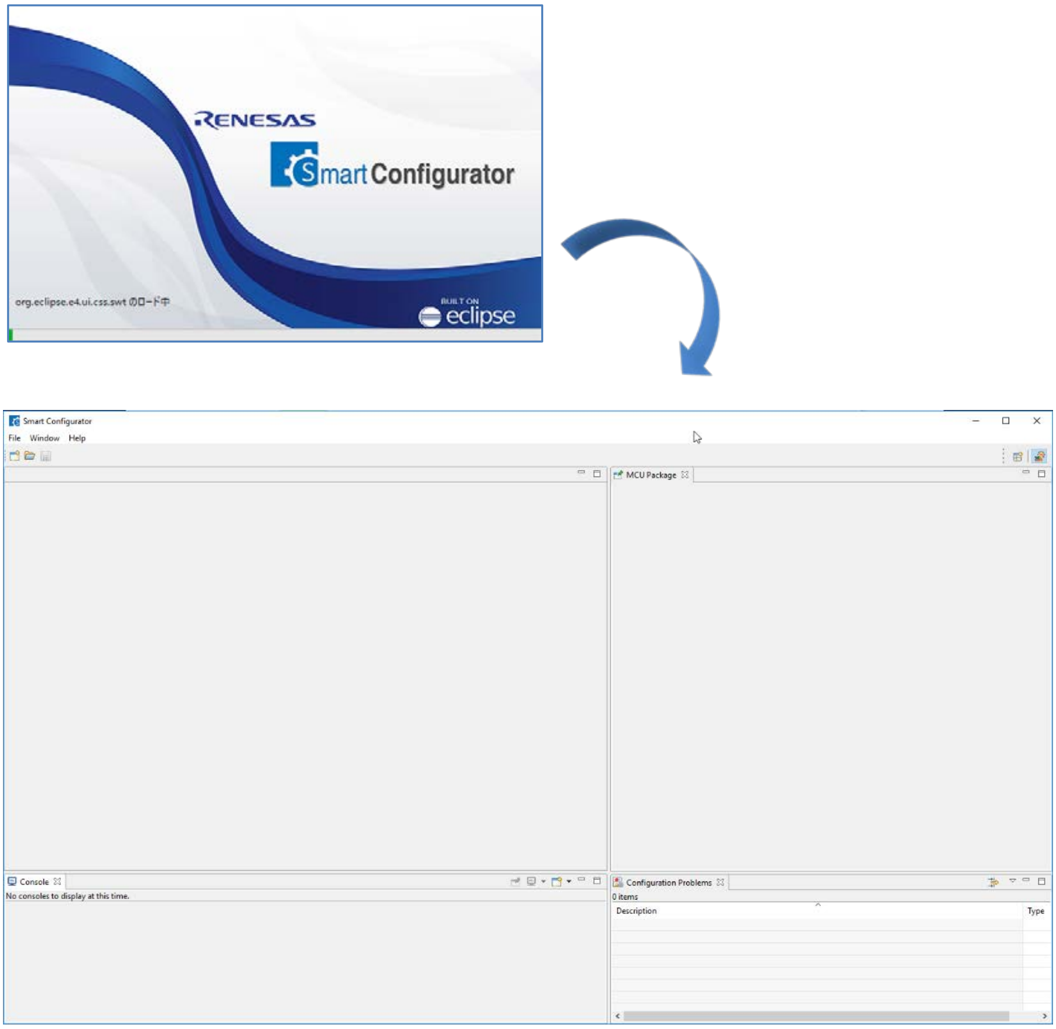

Operations in the Smart Configuration
Starting the Smart Configurator
Refer to ‘3.2 Starting the Smart Configurator’
Create and loading a configuration file
Refer to ‘3.3 Create and loading a configuration file’
Setting of peripheral modules
Refer to ‘4 Setting of Peripheral Modules’ エラー! 参照元が見つかりません。
Setting pins
Refer to ‘4.4 Pin settings’
Setting of interrupts
Refer to ‘4.5 Interrupt settings’
Generating source code
Refer to ‘6 Generating Source Code’
Generating reports
Refer to ‘0
Operations in the IAR Embedded Workbench
Loading generated files in IDE
Refer to ‘0
Creating the user program
Refer to ‘8 Creating User Programs’
Building
Execution and debugging

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.



org.eclipse.e4.ui.css.swt のロード中

No consoles to display at this time.

0 items

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 Downloading FIT modules

The FIT drivers or middlewares are available from the web page of Renesas Electronics.

Download the files from the following address and unzip them.

https://www.renesas.com/fit

3.3.2 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].

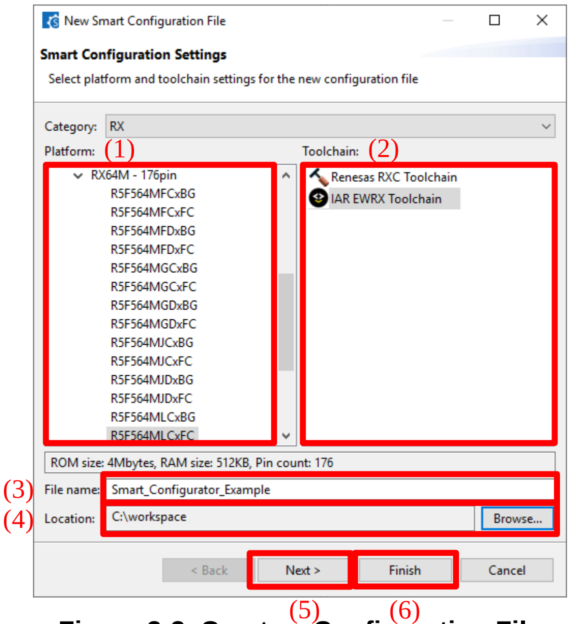
(3) In [File name:], enter the file name.

(4) Confirm [Location:]. If you want to change it, click [Browse] and select the save destination.

(5) If you want to use FIT modules or middleware, click [Next].

Then click [Browse] and set the path of "FITModules" directory which has been unzipped in 3.3.1.

(6) Click [Finish] to create the configuration file.



New Smart Configuration File

Smart Configuration Settings

Select platform and toolchain settings for the new configuration file

Category: RX

Platform:

RX64M - 176pin
R5F564MFCxBG
R5F564MFCxFC
R5F564MFDxBG
R5F564MFDxFC
R5F564MGCxBG
R5F564MGCxFC
R5F564MGDxBG
R5F564MGDxFC
R5F564MJCxBG
R5F564MJCxFC
R5F564MJDxBG
R5F564MJDxFC
R5F564MLCxBG
R5F564MLCxFC

Toolchain:

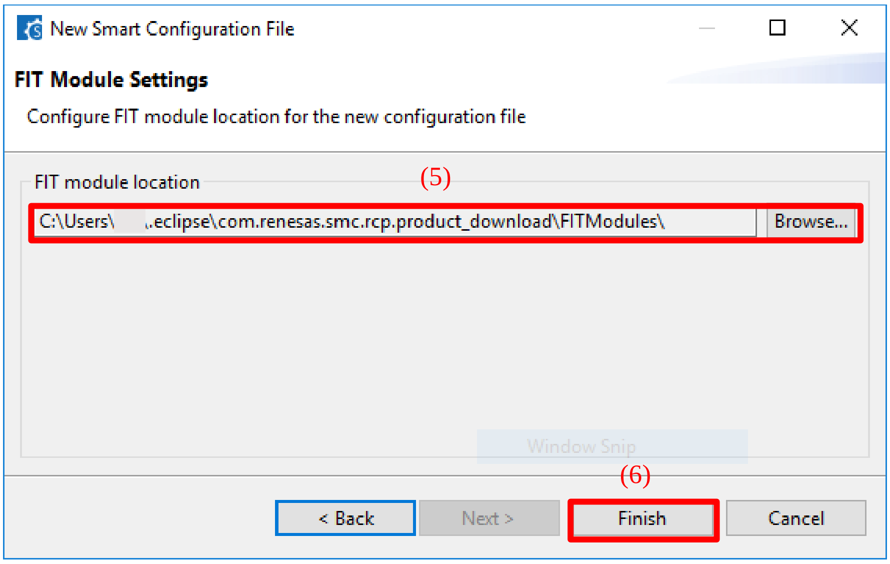
Renesas RXC Toolchain
IAR EWRX Toolchain

ROM size: 4Mbytes, RAM size: 512KB, Pin count: 176

File name: Smart_Configurator_Example

Location: C:\workspace

Figure 3-3 Create a Configuration File



New Smart Configuration File

FIT Module Settings

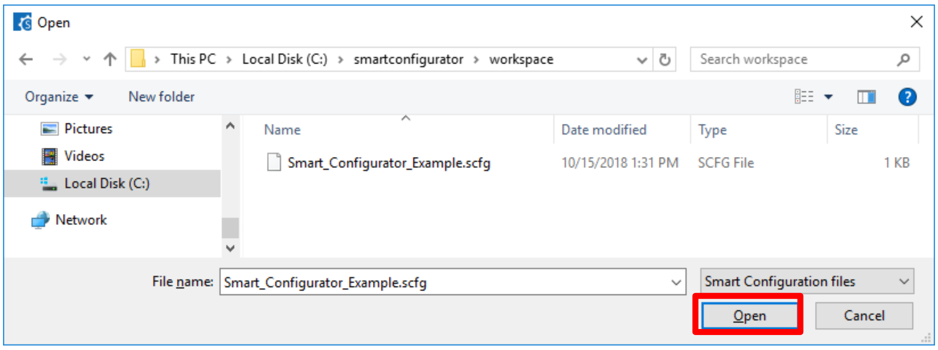
Configure FIT module location for the new configuration file

FIT module location: C:\Users\ .eclipse\com.renesas.smc.rcp.product_download\FITModules\

Figure 3-4 Set the path of FIT module

3.3.3 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].



Open

This PC > Local Disk (C:) > smartconfigurator > workspace

Name	Date modified	Type	Size
Smart_Configurator_Example.scfg	10/15/2018 1:31 PM	SCFG File	1 KB

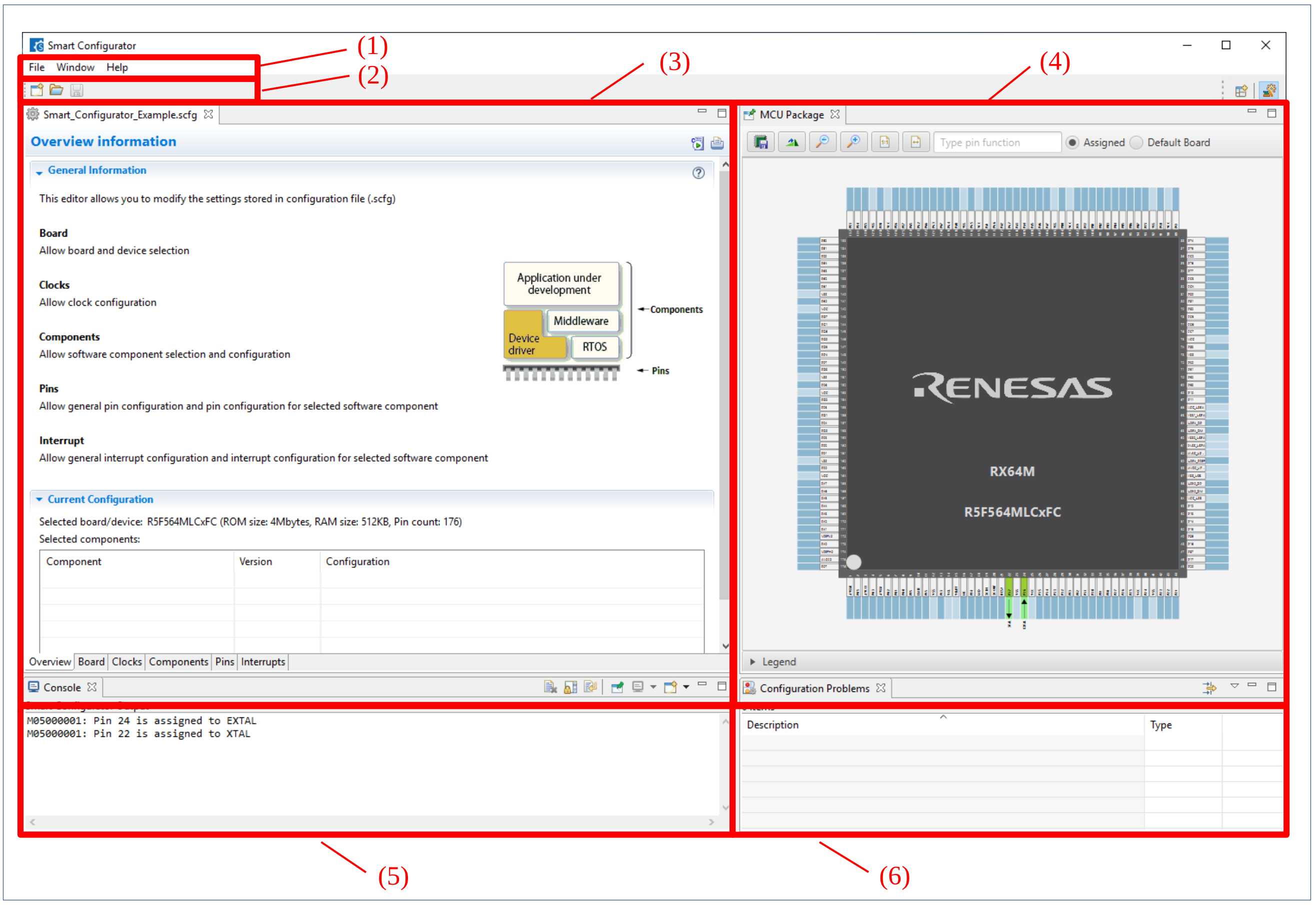
File name: Smart_Configurator_Example.scfg

Smart Configuration files

Figure 3-5 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 3-6, Main Window.



Overview information

General Information

This editor allows you to modify the settings stored in configuration file (.scfg)

Board
Allow board and device selection

Clocks
Allow clock configuration

Components
Allow software component selection and configuration

Pins
Allow general pin configuration and pin configuration for selected software component

Interrupt
Allow general interrupt configuration and interrupt configuration for selected software component

Current Configuration

Selected board/device: R5F564MLCxFC (ROM size: 4Mbytes, RAM size: 512KB, Pin count: 176)
Selected components:

Component	Version	Configuration

```
M05000001: Pin 24 is assigned to EXTAL
M05000001: Pin 22 is assigned to XTAL
```

Figure 3-6 Main Window

(1) Menu bar

(2) Main tool bar

(3) Smart Configurator view

(4) MCU Package view

(5) Console view

(6) Configuration Problems view

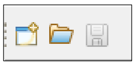
3.4.1 Main menu

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.
	Home Page	Open the home page of the Smart Configurator on the Renesas Electronics website.
	Release Notes	Open the release note of the Smart Configurator on the Renesas Electronics website.
	Tool News	Open the tool news of the Smart Configurator on the Renesas Electronics website.
	API Manual	Open the API manual of the Smart Configurator on the Renesas Electronics website.
	About	The version information is displayed.

3.4.2 Toolbar



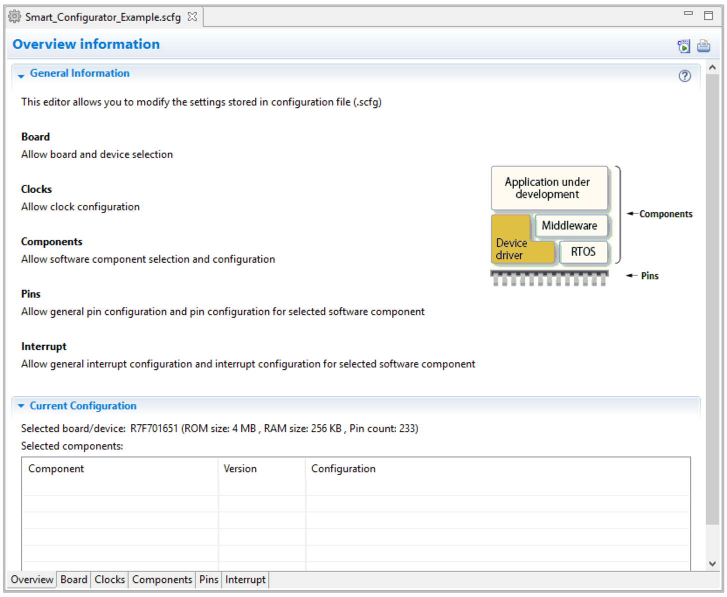
Some functions of the main menu are allocated to the buttons on the toolbar. Table 3-2, Toolbar Buttons and Related Menu Items, shows the description of those tool buttons.

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.



Smart_Configurator_Example.scfg

Overview information

General Information

This editor allows you to modify the settings stored in configuration file (.scfg)

Board
Allow board and device selection

Clocks
Allow clock configuration

Components
Allow software component selection and configuration

Pins
Allow general pin configuration and pin configuration for selected software component

Interrupt
Allow general interrupt configuration and interrupt configuration for selected software component

Current Configuration

Selected board/device: R7F701651 (ROM size: 4 MB , RAM size: 256 KB , Pin count: 233)
Selected components:

Component	Version	Configuration

Figure 3-7 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.

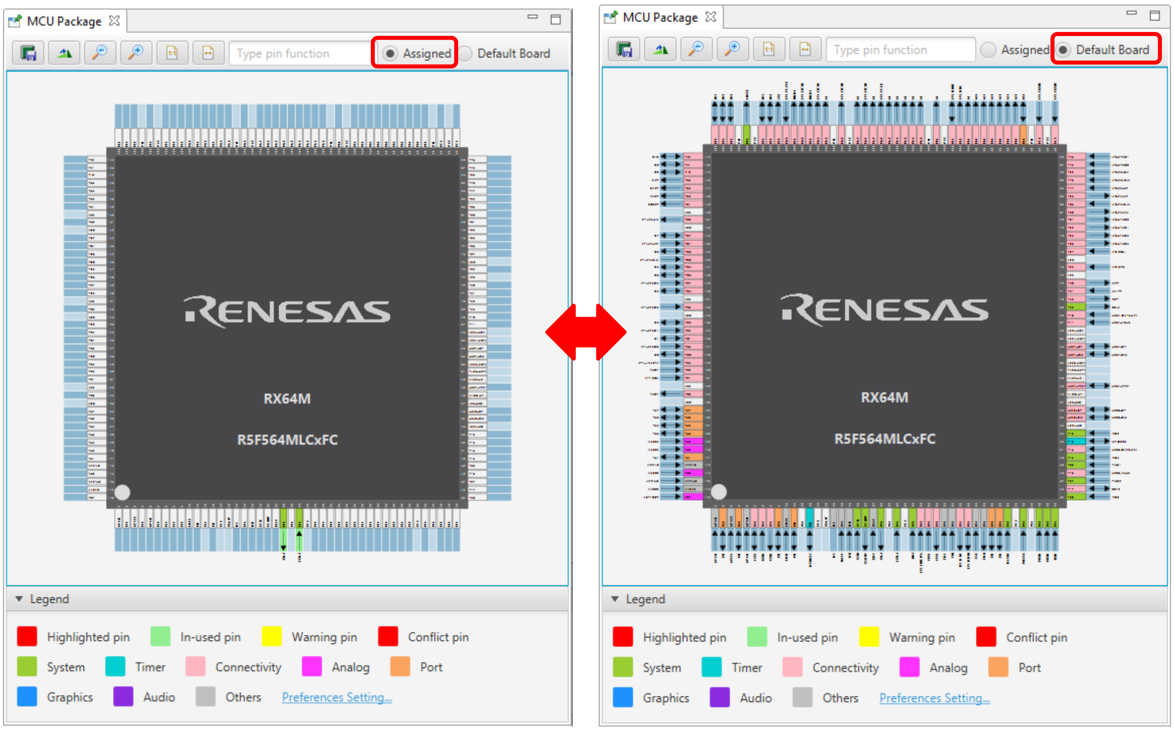
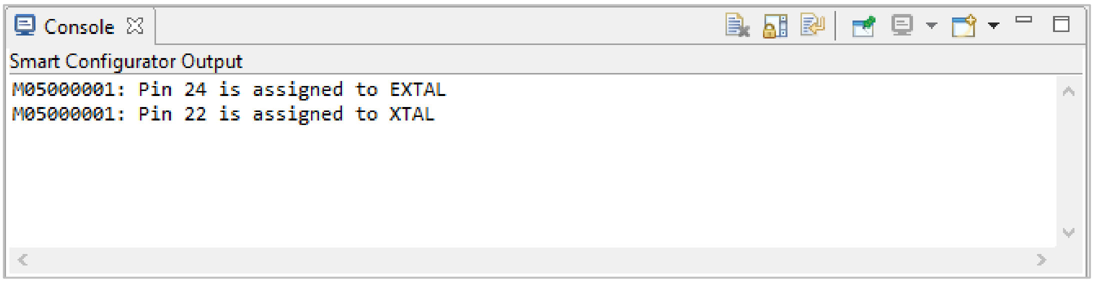


Figure 3-8 MCU Package View

3.4.5 Console view

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



```
M05000001: Pin 24 is assigned to EXTAL
M05000001: Pin 22 is assigned to XTAL
```

Figure 3-9 Console View

3.4.6 Configuration Problems view

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

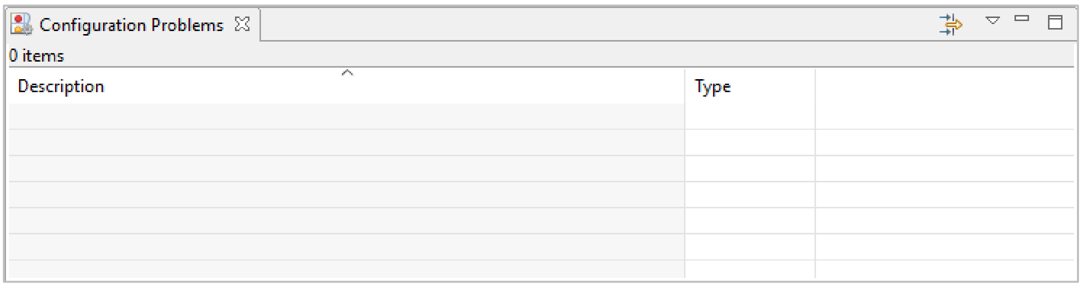


Figure 3-10 Configuration Problems View

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.

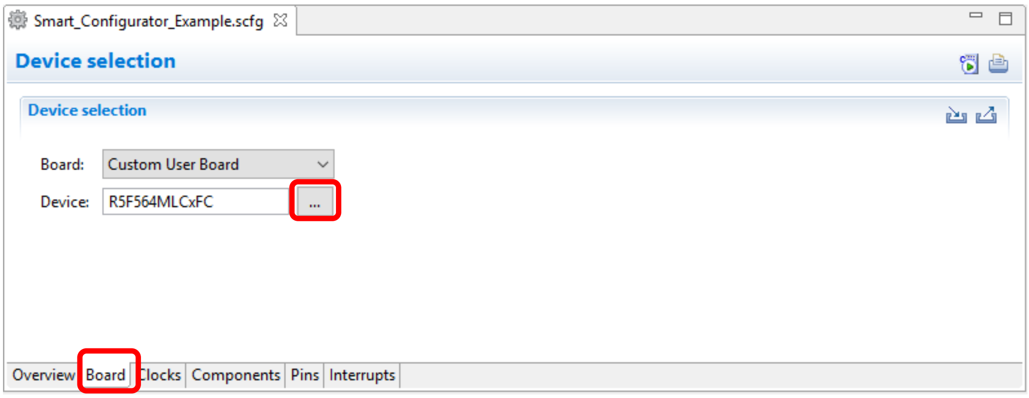
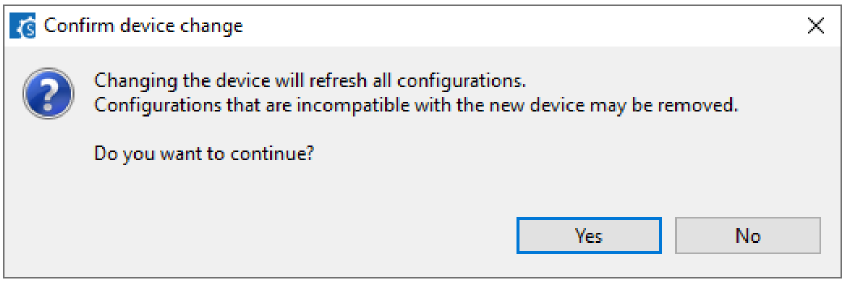


Figure 4-1 Selecting the Device

The following message is displayed when changing the device. For each button operation, refer to "Table 4-1, Device Change Confirmation Operation List".


Confirm device change

Changing the device will refresh all configurations.
Configurations that are incompatible with the new device may be removed.

Do you want to continue?

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.

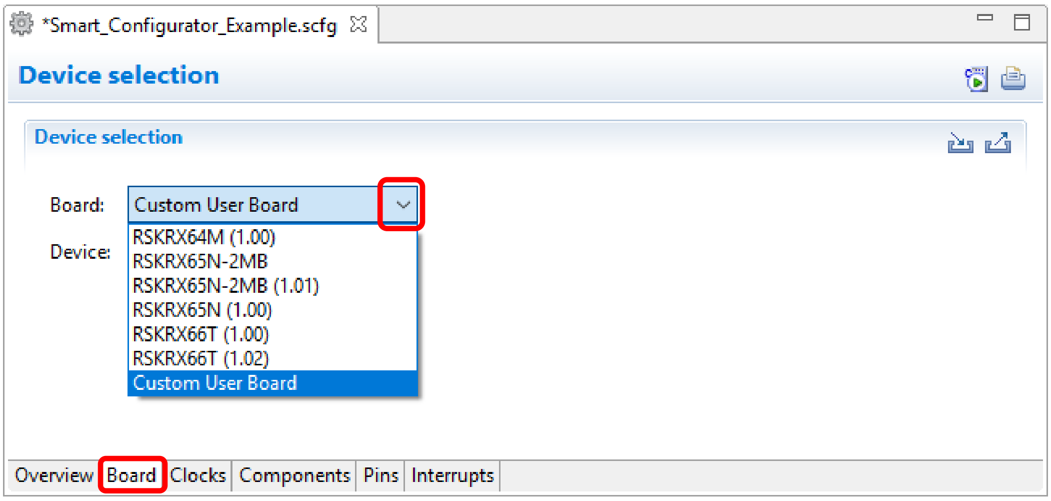
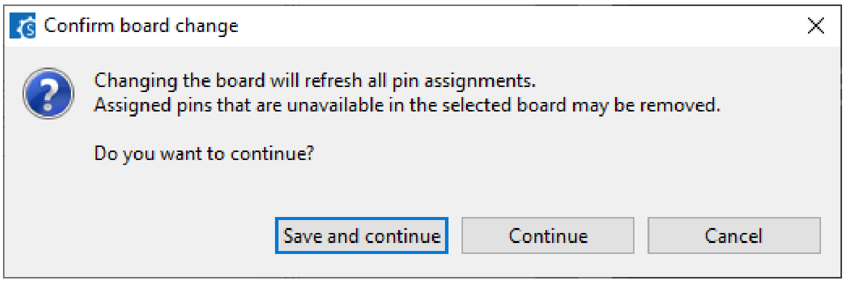


Figure 4-3 Selecting the Board

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".


Confirm board change

Changing the board will refresh all pin assignments.
Assigned pins that are unavailable in the selected board may be removed.

Do you want to continue?

Figure 4-4 Confirm Board Change

Table 4-2 Board Change Confirmation Operation List

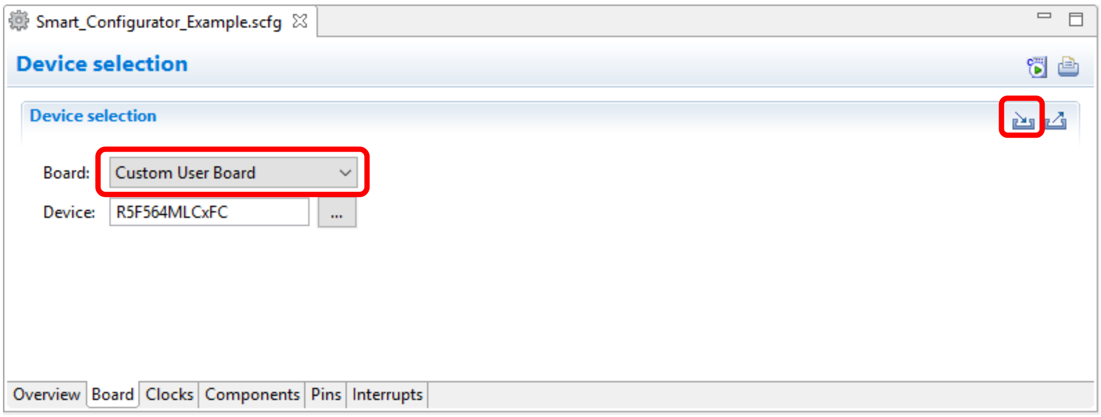
Button	Operation explanation
Save and continue	After saving the current configuration contents to the configuration file, change to the selected device.
Continue	Changes to the selected device without saving the current configuration contents to the configuration file.
Cancel	It does not change the device.

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.



Board: Custom User Board

Device: R5F564MLCxFC

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

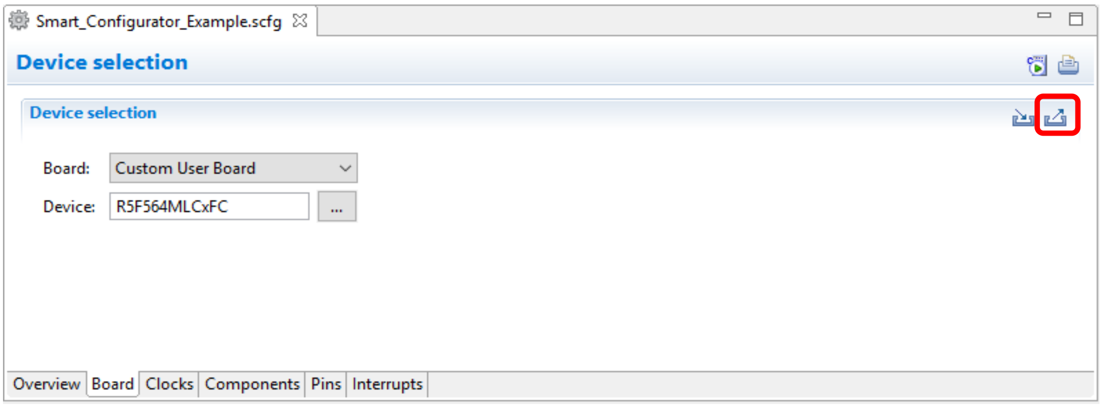
Once a board setting file is imported, the added board is also displayed in the board selection menu of other projects for the same device group.

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.



Board: Custom User Board

Device: R5F564MLCxFC

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.

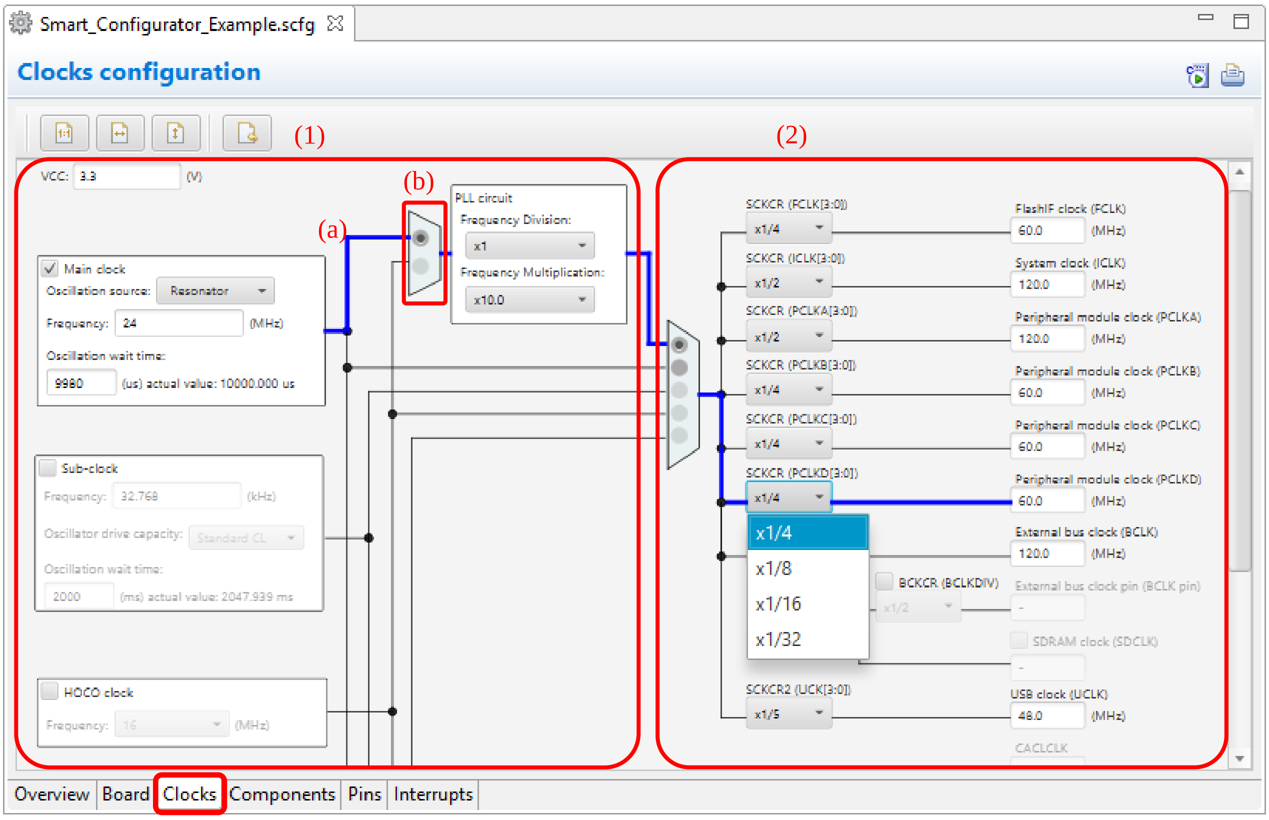
The clocks setting is performed in the following procedure.

(1) Set the clock oscillator circuit.

(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.



VCC: 3.3 (V)

☑ Main clock
Oscillation source: Resonator
Frequency: 24 (MHz)
Oscillation wait time:
9980 (us) actual value: 10000.000 us

□ Sub-clock
Frequency: 32.768 (kHz)
Oscillator drive capacity: Standard CL
Oscillation wait time:
2000 (ms) actual value: 2047.939 ms

□ HOCO clock
Frequency: 16 (MHz)

PLL circuit
Frequency Division: x1
Frequency Multiplication: x10.0

SCKCR (FCLK[3:0]): x1/4 — FlashIF clock (FCLK): 60.0 (MHz)
SCKCR (ICLK[3:0]): x1/2 — System clock (ICLK): 120.0 (MHz)
SCKCR (PCLKA[3:0]): x1/2 — Peripheral module clock (PCLKA): 120.0 (MHz)
SCKCR (PCLKB[3:0]): x1/4 — Peripheral module clock (PCLKB): 60.0 (MHz)
SCKCR (PCLKC[3:0]): x1/4 — Peripheral module clock (PCLKC): 60.0 (MHz)
SCKCR (PCLKD[3:0]): x1/4 — Peripheral module clock (PCLKD): 60.0 (MHz)
External bus clock (BCLK): 120.0 (MHz)
□ BCKCR (BCLKDIV): x1/2 — External bus clock pin (BCLK pin): -
□ SDRAM clock (SDCLK): -
SCKCR2 (UCK[3:0]): x1/5 — USB clock (UCLK): 48.0 (MHz)
CACLCLK

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.

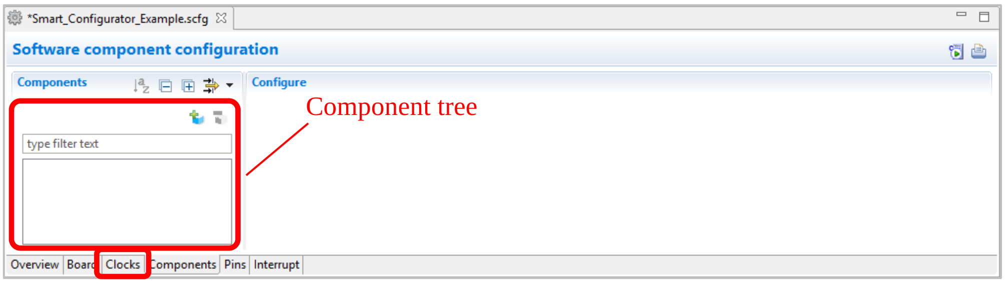


Figure 4-8 Component Page

4.3.1 Adding component

The following describes the procedure for adding a component.

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

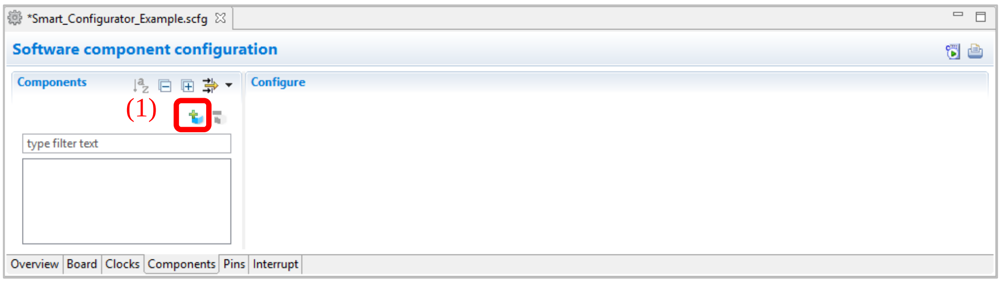
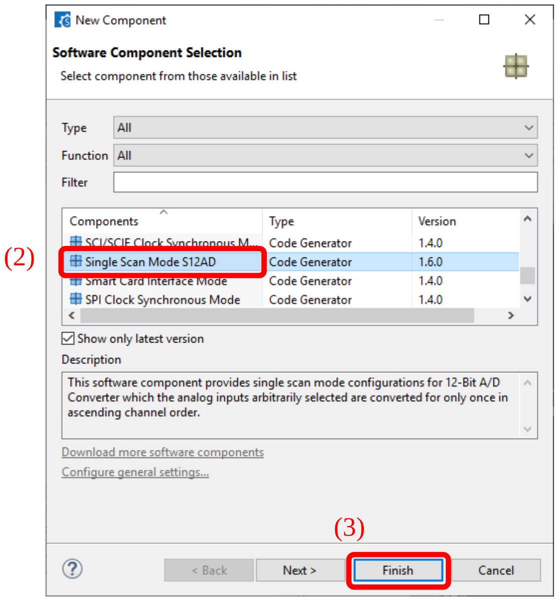


Figure 4-9 Adding Components

(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].


New Component
Software Component Selection
Select component from those available in list
(2)
(3)
Download more software components
Configure general settings...
< Back
Next >
Finish
Cancel

Type: All

Function: All

Filter:

Components	Type	Version
SCI/SCIF Clock Synchronous M...	Code Generator	1.4.0
Single Scan Mode S12AD	Code Generator	1.6.0
Smart Card Interface Mode	Code Generator	1.4.0
SPI Clock Synchronous Mode	Code Generator	1.4.0

☑ Show only latest version

Description

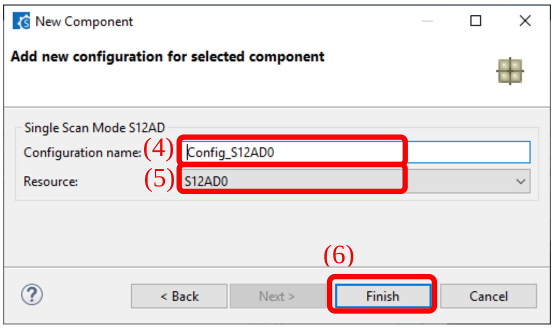
This software component provides single scan mode configurations for 12-Bit A/D Converter which the analog inputs arbitrarily selected are converted for only once in ascending channel order.

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.



New Component

Add new configuration for selected component

Single Scan Mode S12AD

Configuration name: Config_S12AD0

Resource: S12AD0

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

4.3.2 Adding FIT drivers or middleware

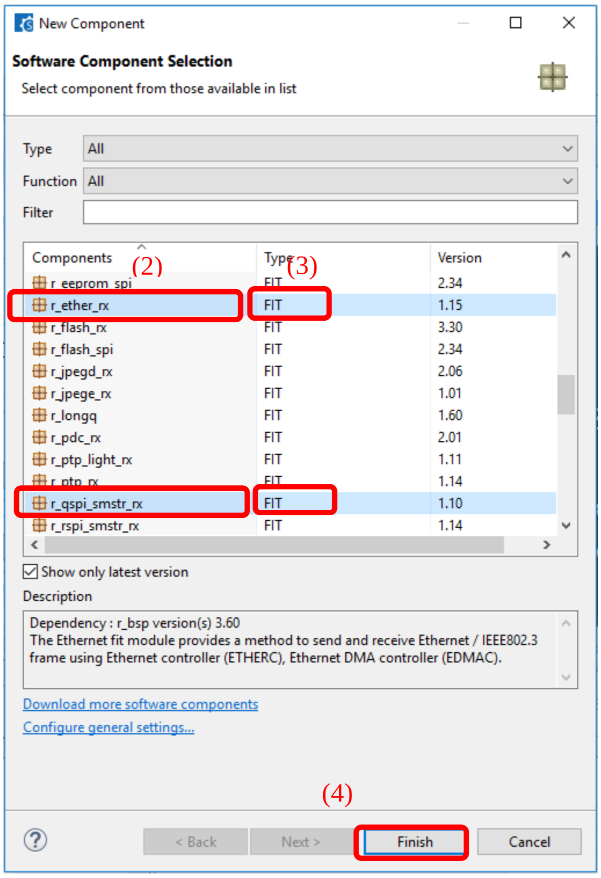
The following describes the procedure for adding a FIT driver or a middleware.

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

(2) Select components from the list in the [Software Component Selection] page of the [New Component] dialog box (e.g. r_ether_rx and r_qspi_smstr_rx). Two or more components can be selected by clicking with the Ctrl key pressed.

(3) Check that [Type] for the selected components is [FIT].

(4) Click on [Finish].



New Component

Software Component Selection

Select component from those available in list

Type: All

Function: All

Filter:

Components	Type	Version
r_eeprom_spi	FIT	2.34
r_ether_rx	FIT	1.15
r_flash_rx	FIT	3.30
r_flash_spi	FIT	2.34
r_jpegd_rx	FIT	2.06
r_jpege_rx	FIT	1.01
r_longq	FIT	1.60
r_pdc_rx	FIT	2.01
r_ptp_light_rx	FIT	1.11
r_ptp_rx	FIT	1.14
r_qspi_smstr_rx	FIT	1.10
r_rspi_smstr_rx	FIT	1.14

☑ Show only latest version

Description

Dependency : r_bsp version(s) 3.60
The Ethernet fit module provides a method to send and receive Ethernet / IEEE802.3 frame using Ethernet controller (ETHERC), Ethernet DMA controller (EDMAC).

Download more software components

Configure general settings...

Figure 4-12 Adding FIT Modules

4.3.3 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.

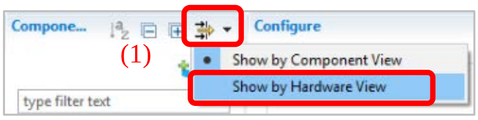
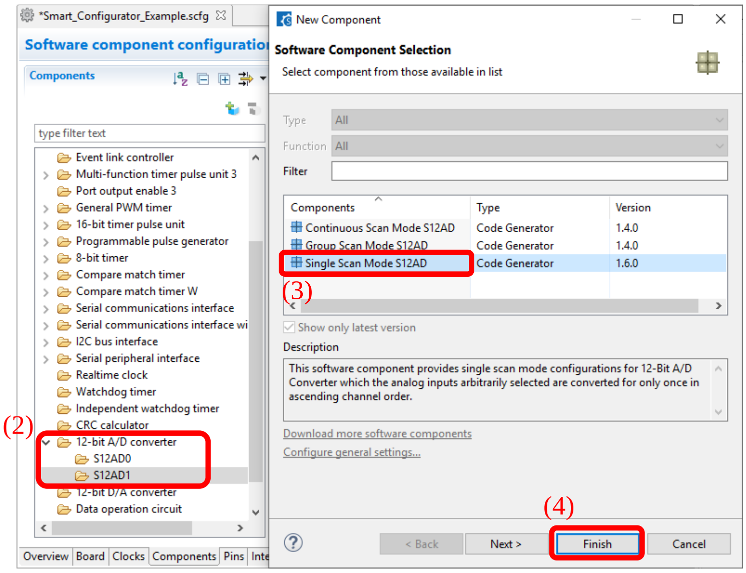


Figure 4-13 Switch to [Show by Hardware View]

(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].



Software Component Selection

Select component from those available in list

Type: All

Function: All

Filter:

Components	Type	Version
Continuous Scan Mode S12AD	Code Generator	1.4.0
Group Scan Mode S12AD	Code Generator	1.4.0
Single Scan Mode S12AD	Code Generator	1.6.0

☑ Show only latest version

Description

This software component provides single scan mode configurations for 12-Bit A/D Converter which the analog inputs arbitrarily selected are converted for only once in ascending channel order.

Download more software components

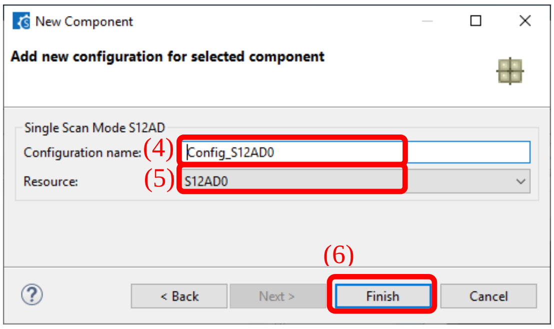
Configure general settings...

Figure 4-14 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.



Add new configuration for selected component

Single Scan Mode S12AD

Configuration name: Config_S12AD0

Resource: S12AD0

Figure 4-15 Add New Configuration for Selected Component (e.g. S12AD1)

4.3.4 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.

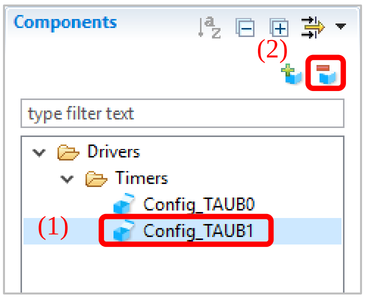


Figure 4-16 Removing a Component

4.3.5 Component configuration settings

Follow the procedure below to setting the component configuration.

(1) Click the component in the component tree. (e.g. Config_S12AD0).

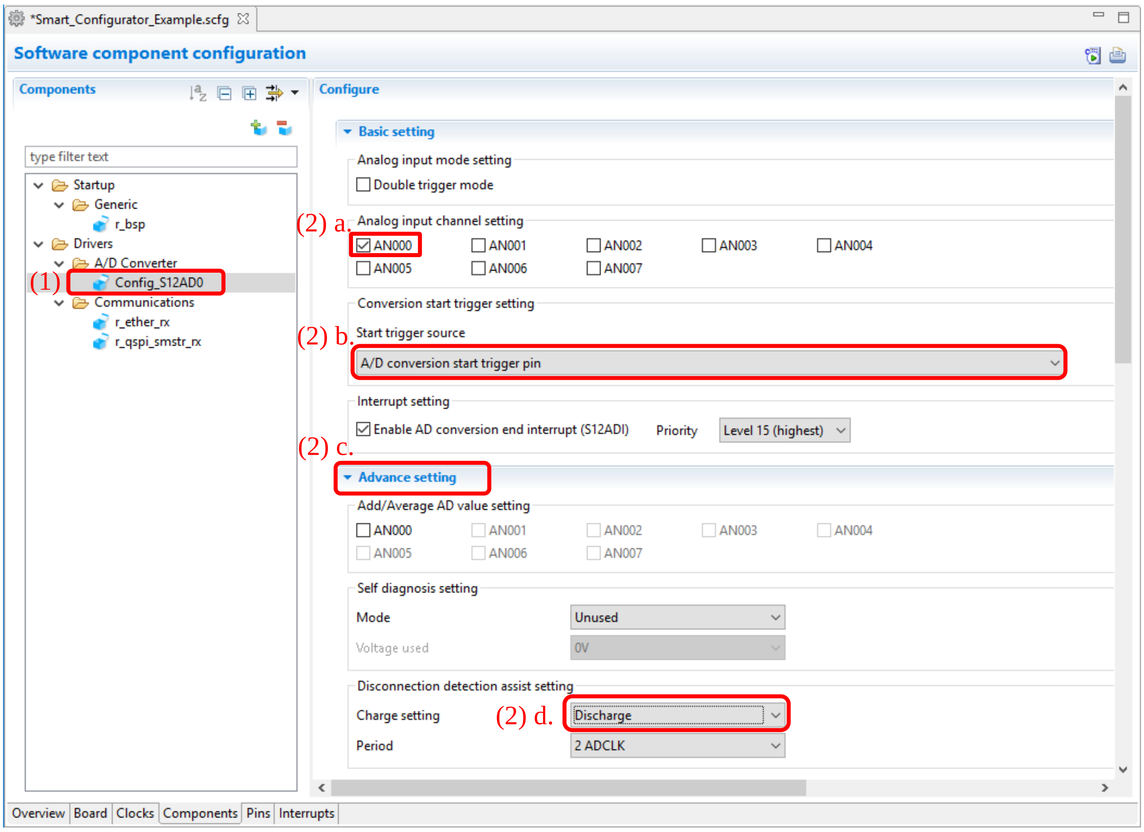
(2) Configure the driver in the [Configure] panel to the right of the Components tree. The Figure 4-17 is an example.

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].



*Smart_Configurator_Example.scfg

Software component configuration

Components

type filter text

- Startup
  - Generic
    - r_bsp
- Drivers
  - A/D Converter
    - Config_S12AD0
  - Communications
    - r_ether_rx
    - r_qspi_smstr_rx

Configure

Basic setting

Analog input mode setting

□ Double trigger mode

Analog input channel setting

☑ AN000 □ AN001 □ AN002 □ AN003 □ AN004

□ AN005 □ AN006 □ AN007

Conversion start trigger setting

Start trigger source: A/D conversion start trigger pin

Interrupt setting

☑ Enable AD conversion end interrupt (S12ADI) Priority: Level 15 (highest)

Advance setting

Add/Average AD value setting

□ AN000 □ AN001 □ AN002 □ AN003 □ AN004

□ AN005 □ AN006 □ AN007

Self diagnosis setting

Mode: Unused

Voltage used: 0V

Disconnection detection assist setting

Charge setting: Discharge

Period: 2 ADCLK

Figure 4-17 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.

4.3.6 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.

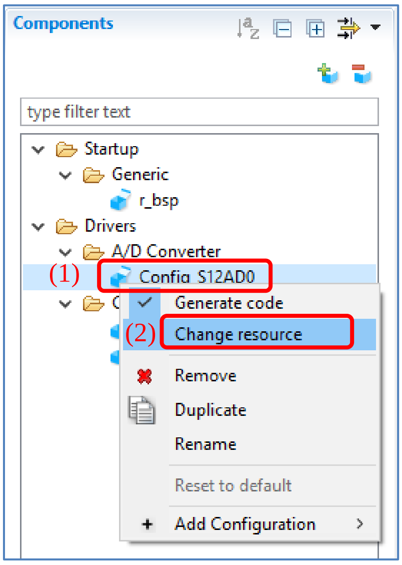
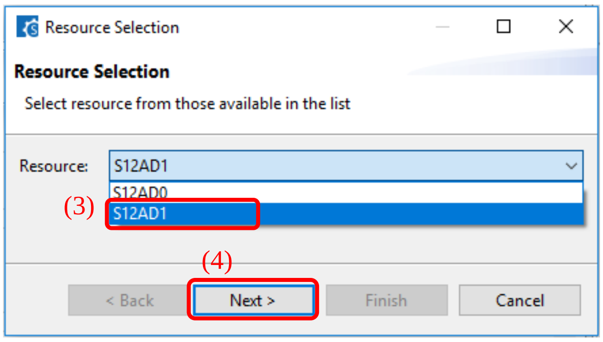


Figure 4-18 Resource Change

(3) Select a new resource in the [Resource Selection] dialog box (e.g. S12AD1).

(4) The [Next] button will be active; click on it.


Resource Selection

Resource Selection

Select resource from those available in the list

Resource: S12AD1

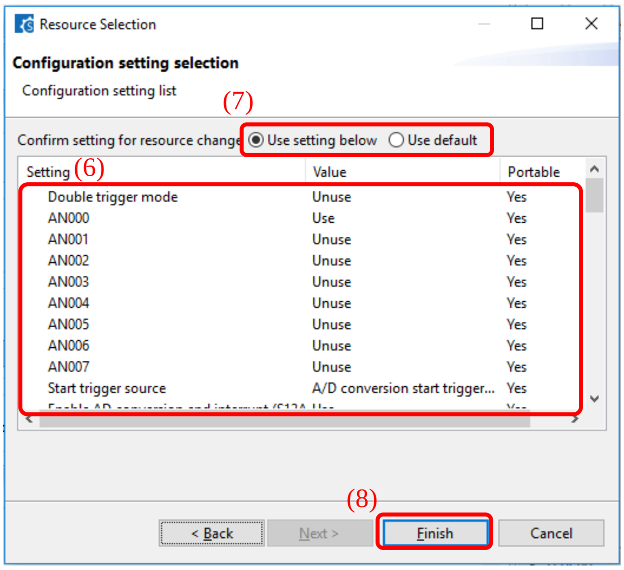
Figure 4-19 Select a New Resource

(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].



Resource Selection

Configuration setting selection

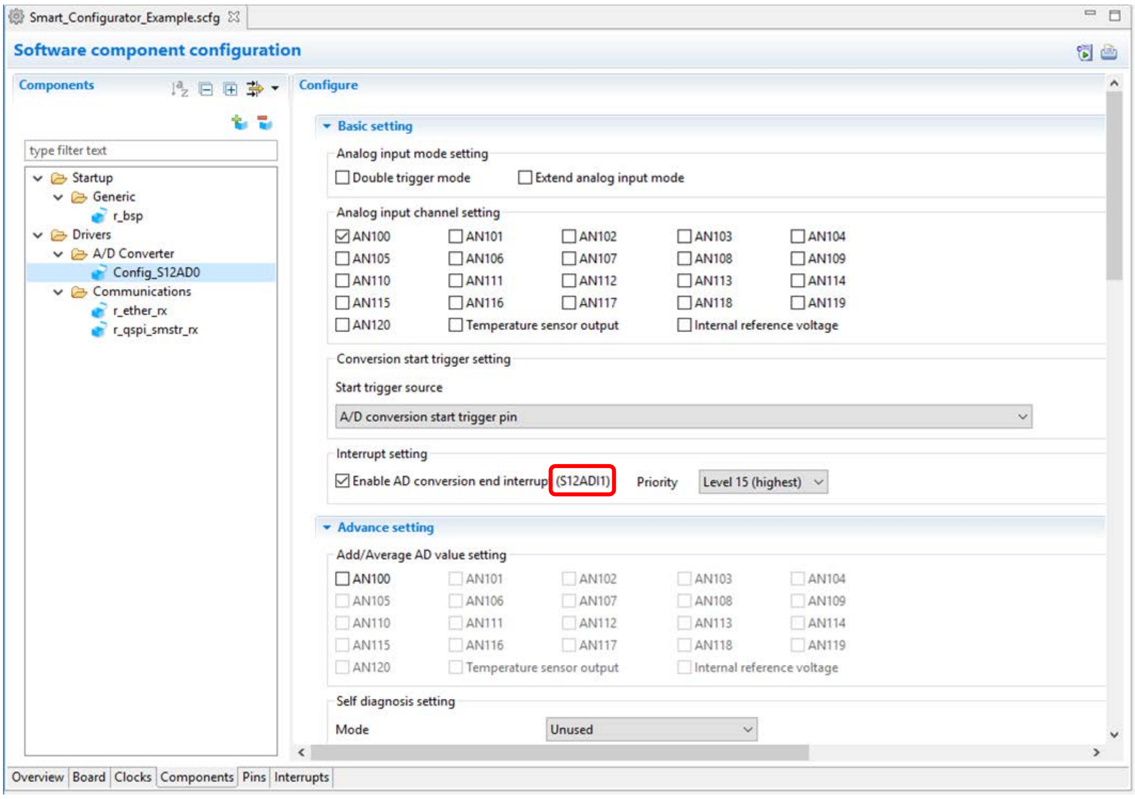
Configuration setting list

Confirm setting for resource change ● Use setting below ○ Use default

Setting	Value	Portable
Double trigger mode	Unuse	Yes
AN000	Use	Yes
AN001	Unuse	Yes
AN002	Unuse	Yes
AN003	Unuse	Yes
AN004	Unuse	Yes
AN005	Unuse	Yes
AN006	Unuse	Yes
AN007	Unuse	Yes
Start trigger source	A/D conversion start trigger...	Yes

Figure 4-20 Confirm New Resource Settings

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



Basic setting

Analog input mode setting

☐ Double trigger mode ☐ Extend analog input mode

Analog input channel setting

☑ AN100 ☐ AN101 ☐ AN102 ☐ AN103 ☐ AN104
☐ AN105 ☐ AN106 ☐ AN107 ☐ AN108 ☐ AN109
☐ AN110 ☐ AN111 ☐ AN112 ☐ AN113 ☐ AN114
☐ AN115 ☐ AN116 ☐ AN117 ☐ AN118 ☐ AN119
☐ AN120 ☐ Temperature sensor output ☐ Internal reference voltage

Conversion start trigger setting

Start trigger source: A/D conversion start trigger pin

Interrupt setting

☑ Enable AD conversion end interrupt (S12ADI1) Priority: Level 15 (highest)

Advance setting

Add/Average AD value setting

☐ AN100 ☐ AN101 ☐ AN102 ☐ AN103 ☐ AN104
☐ AN105 ☐ AN106 ☐ AN107 ☐ AN108 ☐ AN109
☐ AN110 ☐ AN111 ☐ AN112 ☐ AN113 ☐ AN114
☐ AN115 ☐ AN116 ☐ AN117 ☐ AN118 ☐ AN119
☐ AN120 ☐ Temperature sensor output ☐ Internal reference voltage

Self diagnosis setting

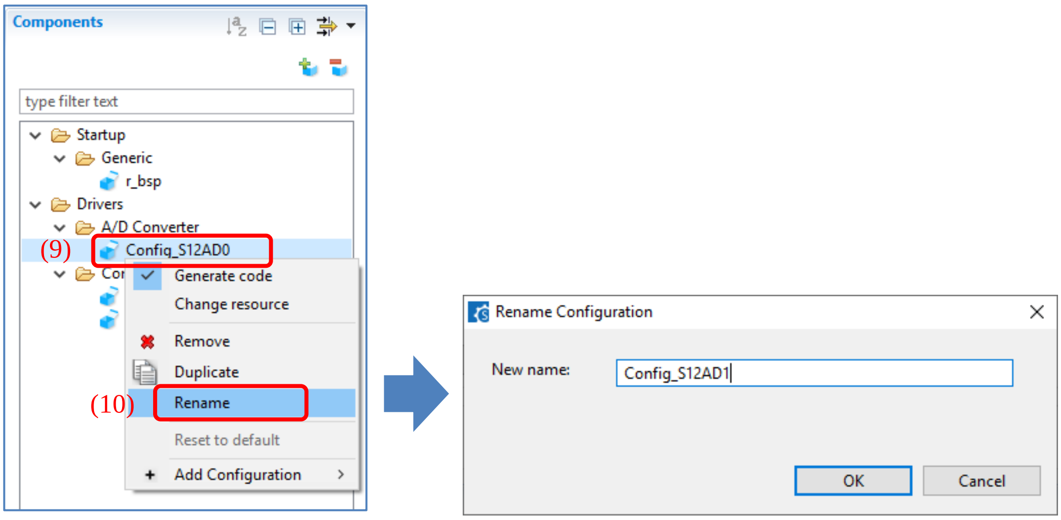
Mode: Unused

Figure 4-21 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).



New name: Config_S12AD1

Figure 4-22 Renaming the Configuration

4.3.7 Setting of the FIT software components

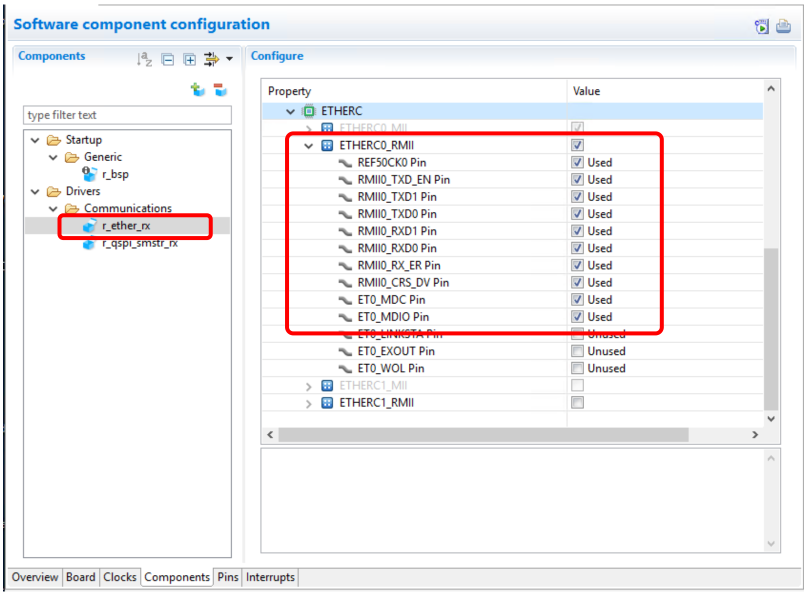
The FIT drivers or middleware are available by setting configuration options.

The way of setting depends on each component.

- Set on the panel
- Set with modifying the configuration file for the FIT module

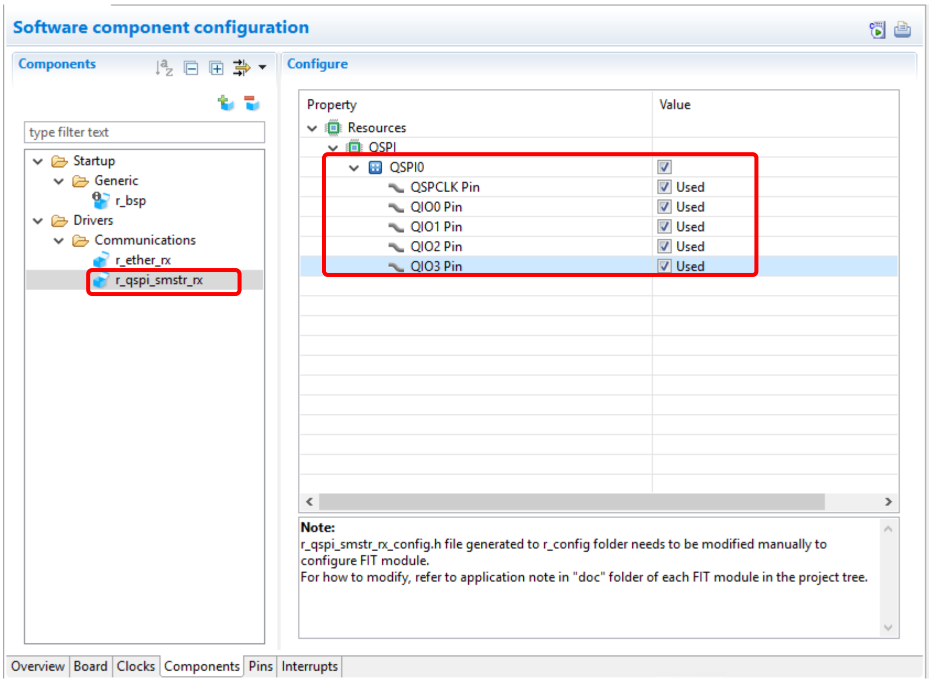
The configuration file is generated in the r_config folder after source code generation.
See the ‘エラー! 参照元が見つかりません。 エラー! 参照元が見つかりません。’ to set the configuration options.

Set the functions of the ports at the panel in each component.
Some examples are following.



Property	Value
ETHERC	
ETHERC0_MII	☑
ETHERC0_RMII	☑
REF50CK0 Pin	☑ Used
RMII0_TXD_EN Pin	☑ Used
RMII0_TXD1 Pin	☑ Used
RMII0_TXD0 Pin	☑ Used
RMII0_RXD1 Pin	☑ Used
RMII0_RXD0 Pin	☑ Used
RMII0_RX_ER Pin	☑ Used
RMII0_CRS_DV Pin	☑ Used
ET0_MDC Pin	☑ Used
ET0_MDIO Pin	☑ Used
ET0_LINKSTA Pin	□ Unused
ET0_EXOUT Pin	□ Unused
ET0_WOL Pin	□ Unused
ETHERC1_MII	□
ETHERC1_RMII	□

Figure 4-23 port setting of `r_ether_rx`



Property	Value
Resources	
QSPI	
QSPI0	☑
QSPCLK Pin	☑ Used
QIO0 Pin	☑ Used
QIO1 Pin	☑ Used
QIO2 Pin	☑ Used
QIO3 Pin	☑ Used

Note:
r_qspi_smstr_rx_config.h file generated to r_config folder needs to be modified manually to configure FIT module.
For how to modify, refer to application note in "doc" folder of each FIT module in the project tree.

Figure 4-24 port setting of `r_qspi_smstr_rx`

4.3.8 Changing the version of the FIT software components

Change the version of the FIT software component as follows.
(1) Right click the FIT software component in the component tree.

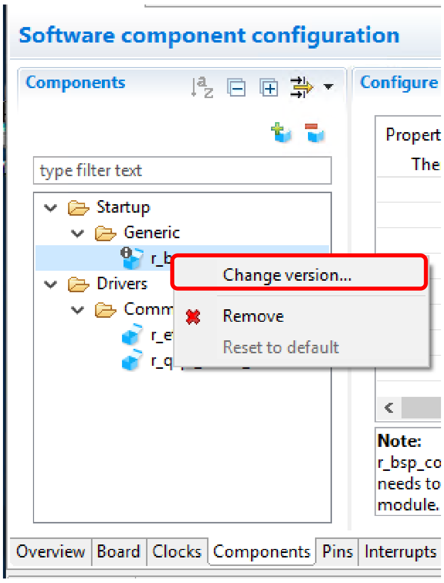
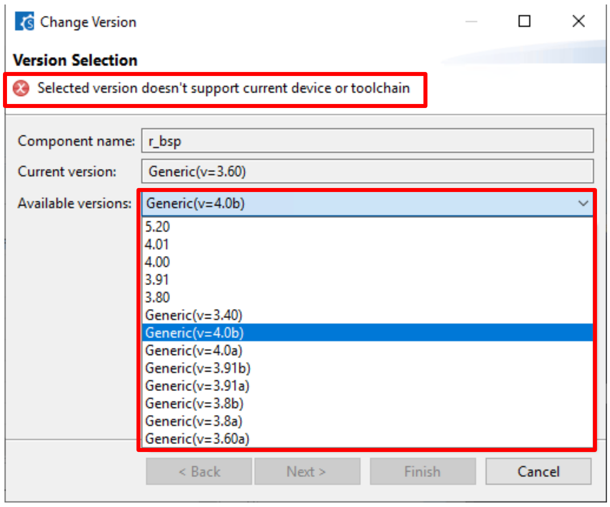


Figure 4-25 Changing the version of FIT software component

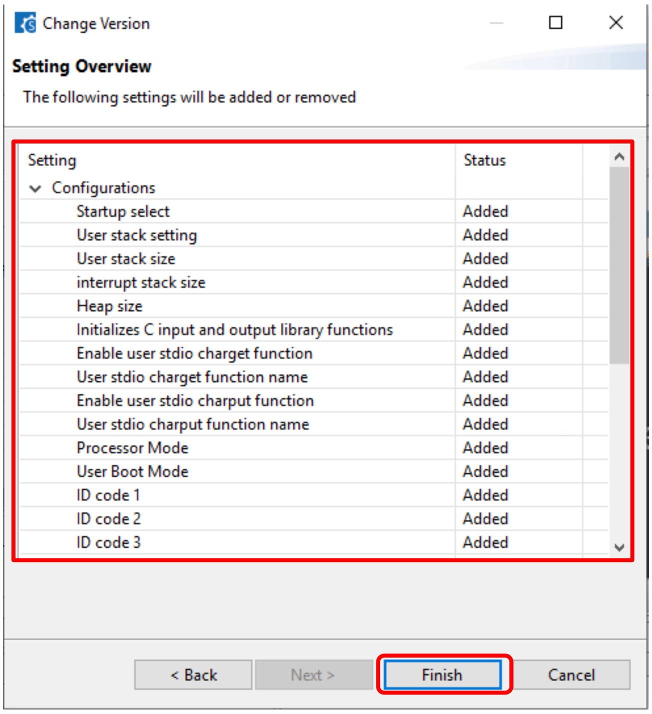
(2) Select the [Change version…] in the context menu.
(3) If supported version is selected, the [Next] button will be clickable.
Otherwise, the message “Selected version doesn't support current device or toolchain” is displayed.



Selected version doesn't support current device or toolchain

Component name: r_bsp
Current version: Generic(v=3.60)
Available versions: Generic(v=4.0b)

(4) Click the [Next] button.
(5) The setting change items are displayed.
Click the [Finish] button if there are no problems with changing the settings.

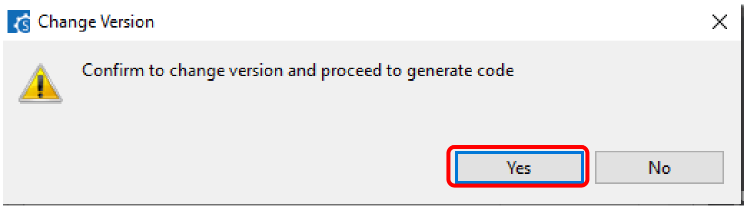


Setting Overview

The following settings will be added or removed

Setting	Status
Configurations	
Startup select	Added
User stack setting	Added
User stack size	Added
interrupt stack size	Added
Heap size	Added
Initializes C input and output library functions	Added
Enable user stdio charget function	Added
User stdio charget function name	Added
Enable user stdio charput function	Added
User stdio charput function name	Added
Processor Mode	Added
User Boot Mode	Added
ID code 1	Added
ID code 2	Added
ID code 3	Added

(6) The message “Confirm to change version and proceed to generate code” is displayed.
Click [Yes] button if there are no problems.



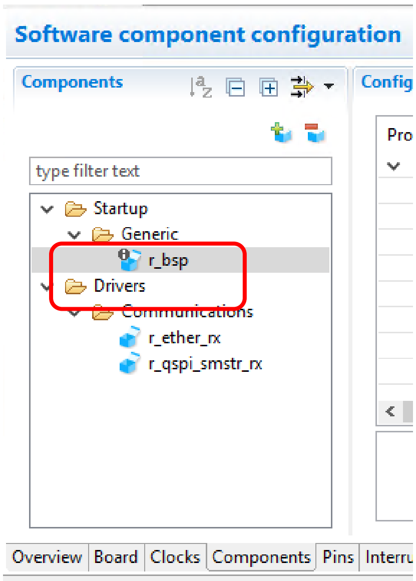
Confirm to change version and proceed to generate code

(7) The version of software component has been changed then the source codes are generated automatically.

4.3.9 Sample program sign of FIT icon

If the icon of FIT driver or middleware is [📦], there are sample programs of it.
Download the sample program files from Renesas Electronics home page.

https://www.renesas.com/fit

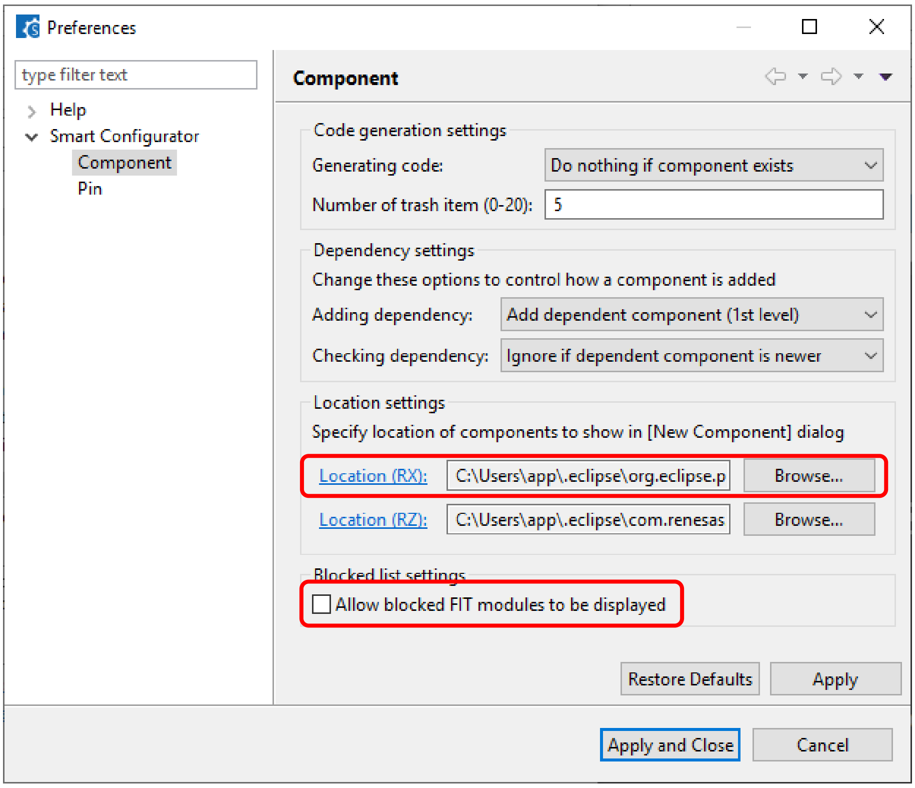


4.3.10 Configure general setting of component

You can change the general setting of the component such as location and dependency. If you want to change it, click the [Configure general settings...] link on the [Software Component Selection] page displayed in the [New Component] dialog (Figure 4-10), and display the [Preferences] dialog.

If you want to change the location of components, click [Browse] link at [Location (RX)] and set the location.

If you want to display the all components in the [Software Component Selection] dialog when you add some components at 4.3.1, you should check the [Allow blocked FIT modules to be displayed].



Code generation settings

Generating code: Do nothing if component exists

Number of trash item (0-20): 5

Dependency settings

Change these options to control how a component is added

Adding dependency: Add dependent component (1st level)

Checking dependency: Ignore if dependent component is newer

Location settings

Specify location of components to show in [New Component] dialog

Location (RX): C:\Users\app\.eclipse\org.eclipse.p Browse...

Location (RZ): C:\Users\app\.eclipse\com.renesas Browse...

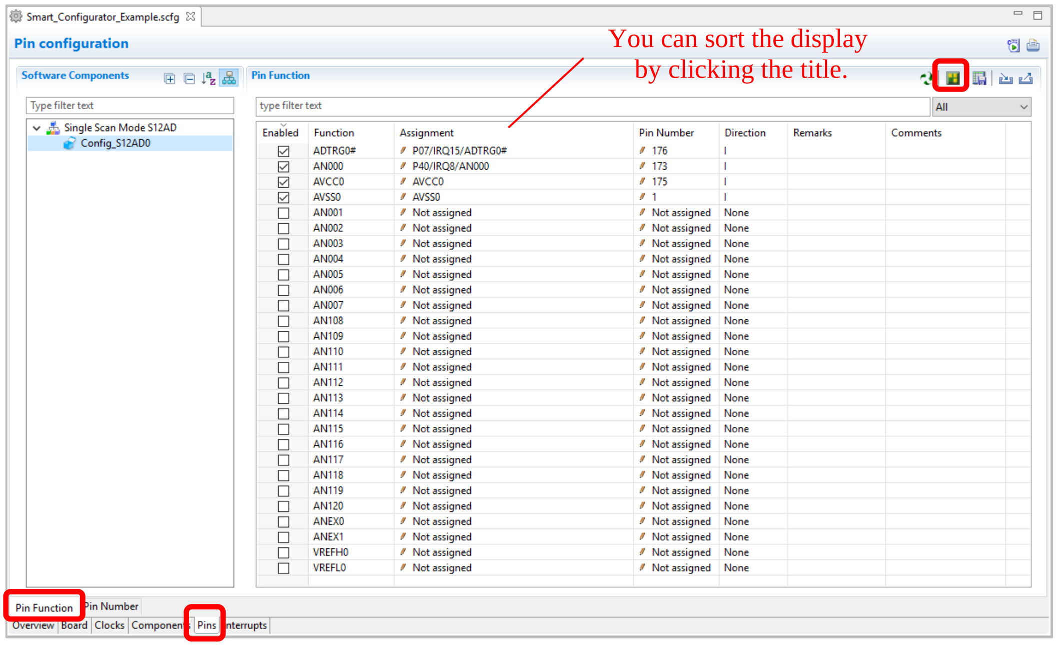
Blocked list settings

☐ Allow blocked FIT modules to be displayed

Figure 4-26 Configure general setting of component

4.4 Pin settings

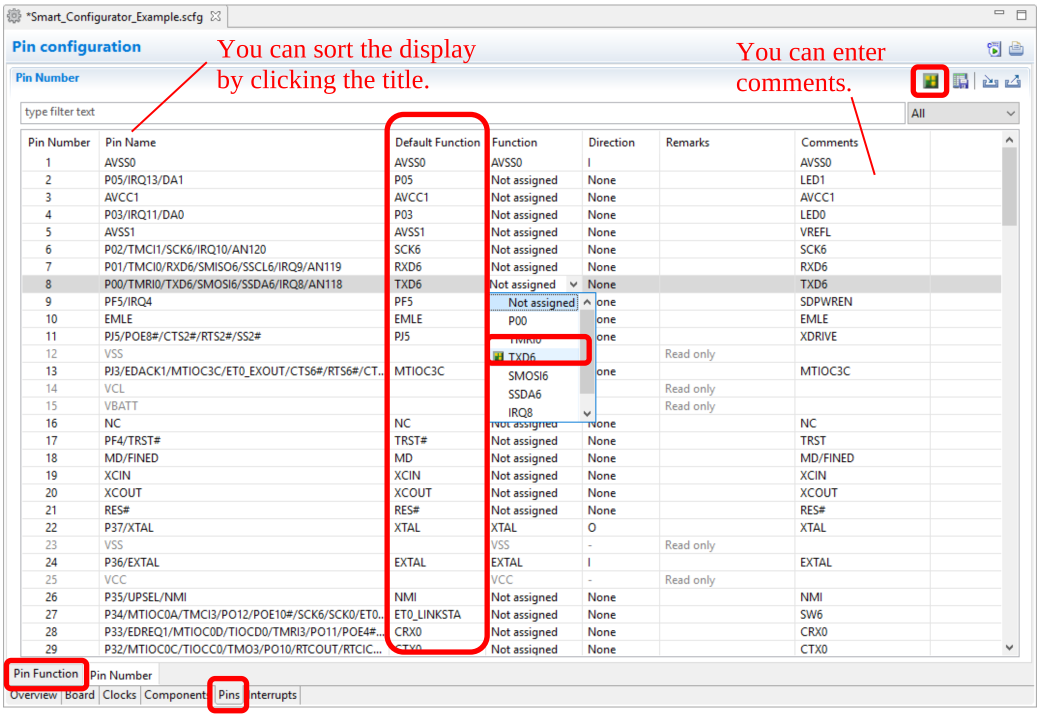
The [Pins] page is used for allocating pin functions. You can switch the display by clicking on the [Pin Function] and [Pin Number] tabs. The [Pin Function] list shows the pin functions for each of the peripheral functions, and the [Pin Number] list shows all pins in order of pin number.



You can sort the display by clicking the title.

Enabled	Function	Assignment	Pin Number	Direction	Remarks	Comments
☑	ADTRG0#	P07/IRQ15/ADTRG0#	176	I		
☑	AN000	P40/IRQ8/AN000	173	I		
☑	AVCC0	AVCC0	175	I		
☑	AVSS0	AVSS0	1	I		
□	AN001	Not assigned	Not assigned	None		
□	AN002	Not assigned	Not assigned	None		
□	AN003	Not assigned	Not assigned	None		
□	AN004	Not assigned	Not assigned	None		
□	AN005	Not assigned	Not assigned	None		
□	AN006	Not assigned	Not assigned	None		
□	AN007	Not assigned	Not assigned	None		
□	AN108	Not assigned	Not assigned	None		
□	AN109	Not assigned	Not assigned	None		
□	AN110	Not assigned	Not assigned	None		
□	AN111	Not assigned	Not assigned	None		
□	AN112	Not assigned	Not assigned	None		
□	AN113	Not assigned	Not assigned	None		
□	AN114	Not assigned	Not assigned	None		
□	AN115	Not assigned	Not assigned	None		
□	AN116	Not assigned	Not assigned	None		
□	AN117	Not assigned	Not assigned	None		
□	AN118	Not assigned	Not assigned	None		
□	AN119	Not assigned	Not assigned	None		
□	AN120	Not assigned	Not assigned	None		
□	ANEX0	Not assigned	Not assigned	None		
□	ANEX1	Not assigned	Not assigned	None		
□	VREFH0	Not assigned	Not assigned	None		
□	VREFL0	Not assigned	Not assigned	None		

Figure 4-27 [Pins] Page ([Pin Function])



You can sort the display by clicking the title.

You can enter comments.

Pin Number	Pin Name	Default Function	Function	Direction	Remarks	Comments
1	AVSS0	AVSS0	AVSS0	I		AVSS0
2	P05/IRQ13/DA1	P05	Not assigned	None		LED1
3	AVCC1	AVCC1	Not assigned	None		AVCC1
4	P03/IRQ11/DA0	P03	Not assigned	None		LED0
5	AVSS1	AVSS1	Not assigned	None		VREFL
6	P02/TMCI1/SCK6/IRQ10/AN120	SCK6	Not assigned	None		SCK6
7	P01/TMCI0/RXD6/SMISO6/SSCL6/IRQ9/AN119	RXD6	Not assigned	None		RXD6
8	P00/TMRI0/TXD6/SMOSI6/SSDA6/IRQ8/AN118	TXD6	Not assigned	None		TXD6
9	PF5/IRQ4	PF5		one		SDPWREN
10	EMLE	EMLE		one		EMLE
11	PJ5/POE8#/CTS2#/RTS2#/SS2#	PJ5		one		XDRIVE
12	VSS				Read only	
13	PJ3/EDACK1/MTIOC3C/ET0_EXOUT/CTS6#/RTS6#/CT...	MTIOC3C		one		MTIOC3C
14	VCL				Read only	
15	VBATT				Read only	
16	NC	NC	Not assigned	None		NC
17	PF4/TRST#	TRST#	Not assigned	None		TRST
18	MD/FINED	MD	Not assigned	None		MD/FINED
19	XCIN	XCIN	Not assigned	None		XCIN
20	XCOUT	XCOUT	Not assigned	None		XCOUT
21	RES#	RES#	Not assigned	None		RES#
22	P37/XTAL	XTAL	XTAL	O		XTAL
23	VSS		VSS	-	Read only	
24	P36/EXTAL	EXTAL	EXTAL	I		EXTAL
25	VCC		VCC	-	Read only	
26	P35/UPSEL/NMI	NMI	Not assigned	None		NMI
27	P34/MTIOC0A/TMCI3/PO12/POE10#/SCK6/SCK0/ET0...	ET0_LINKSTA	Not assigned	None		SW6
28	P33/EDREQ1/MTIOC0D/TIOCD0/TMRI3/PO11/POE4#...	CRX0	Not assigned	None		CRX0
29	P32/MTIOC0C/TIOCC0/TMO3/PO10/RTCOUT/RTCIC...	CTX0	Not assigned	None		CTX0

Figure 4-28 [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.

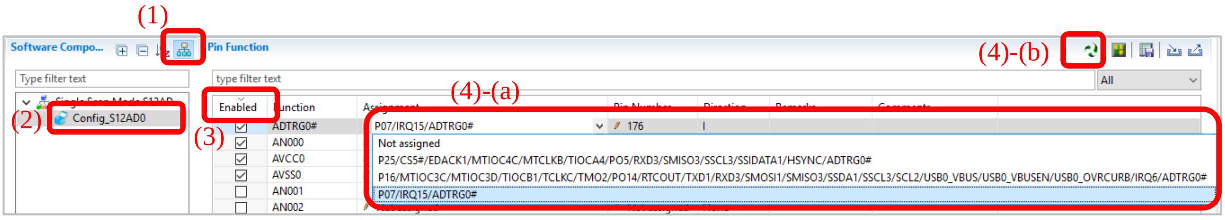
(2) Select the target software component (e.g. Config_S12AD1).

(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.



Enabled	Function	Assignment	Pin Number	Direction	Remarks	Comments
☑	ADTRG0#	P07/IRQ15/ADTRG0#	176	I		
☑	AN000					
☑	AVCC0					
☑	AVSS0					
☐	AN001					
☐	AN002					

Not assigned
P25/CS5#/EDACK1/MTIOC4C/MTCLKB/TIOCA4/PO5/RXD3/SMISO3/SSCL3/SSIDATA1/HSYNC/ADTRG0#
P16/MTIOC3C/MTIOC3D/TIOCB1/TCLKC/TMO2/PO14/RTCOUT/TXD1/RXD3/SMOSI1/SMISO3/SSDA1/SSCL3/SCL2/USB0_VBUS/USB0_VBUSEN/USB0_OVRCURB/IRQ6/ADTRG0#
P07/IRQ15/ADTRG0#

Figure 4-29 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.

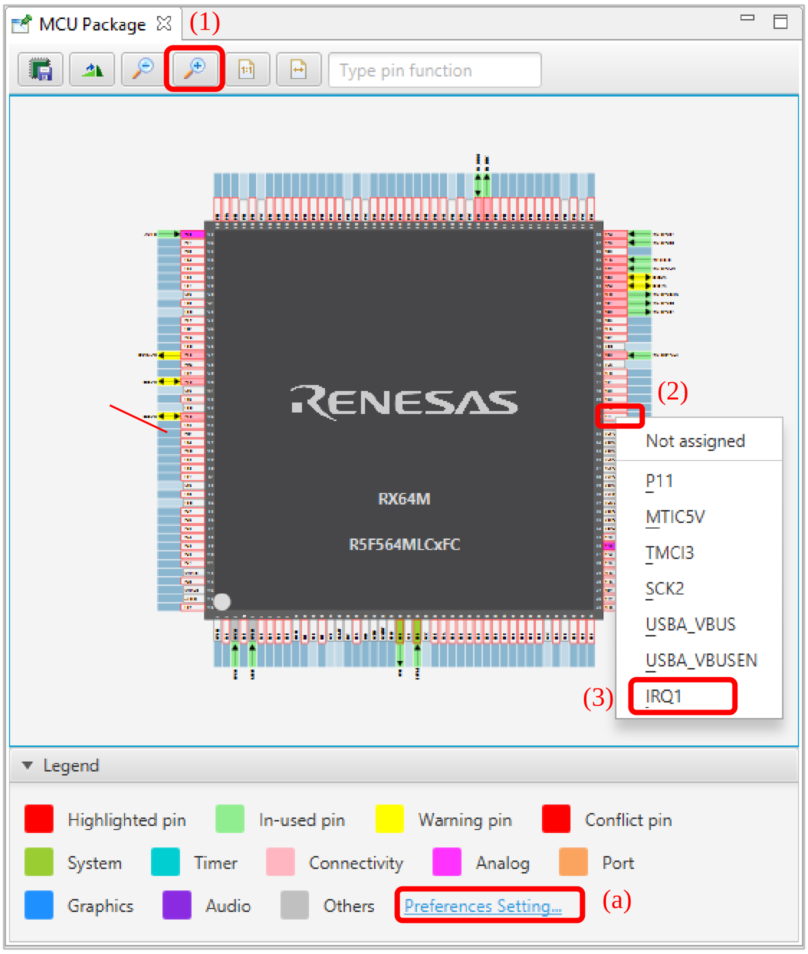


Figure 4-30 Assigning Pins Using the MCU Package View

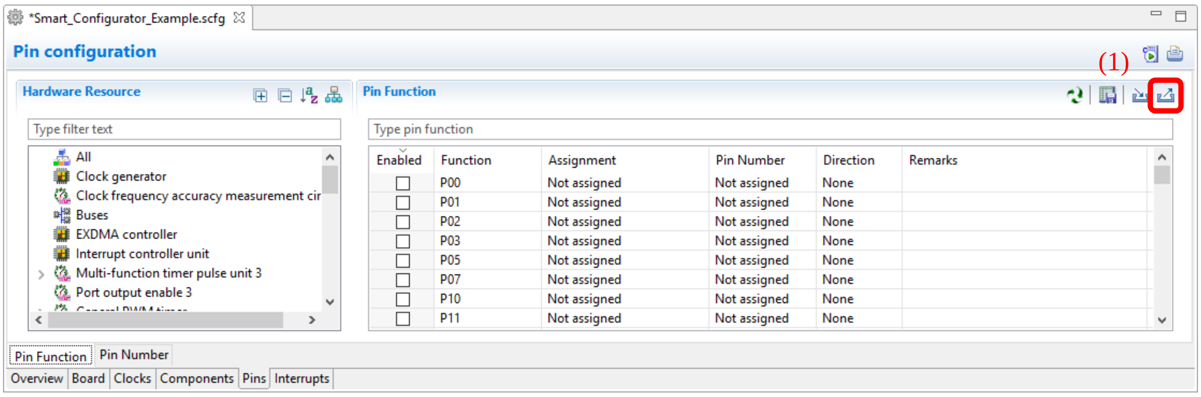
(a) The color of the pins can be customized through [Preference Setting...].

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.



Enabled	Function	Assignment	Pin Number	Direction	Remarks
□	P00	Not assigned	Not assigned	None	
□	P01	Not assigned	Not assigned	None	
□	P02	Not assigned	Not assigned	None	
□	P03	Not assigned	Not assigned	None	
□	P05	Not assigned	Not assigned	None	
□	P07	Not assigned	Not assigned	None	
□	P10	Not assigned	Not assigned	None	
□	P11	Not assigned	Not assigned	None	

Figure 4-31 Export Pin Settings (XML format)

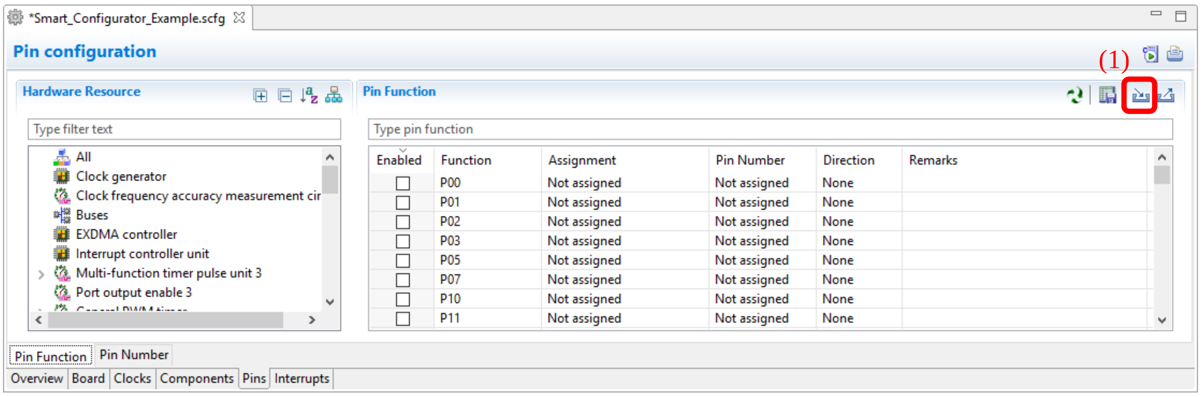
The Smart Configurator can also export the pin settings to a CSV file. Click on the [(Save the list to .csv file)] button on the [Pins] page.

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.



Enabled	Function	Assignment	Pin Number	Direction	Remarks
□	P00	Not assigned	Not assigned	None	
□	P01	Not assigned	Not assigned	None	
□	P02	Not assigned	Not assigned	None	
□	P03	Not assigned	Not assigned	None	
□	P05	Not assigned	Not assigned	None	
□	P07	Not assigned	Not assigned	None	
□	P10	Not assigned	Not assigned	None	
□	P11	Not assigned	Not assigned	None	

Figure 4-32 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].

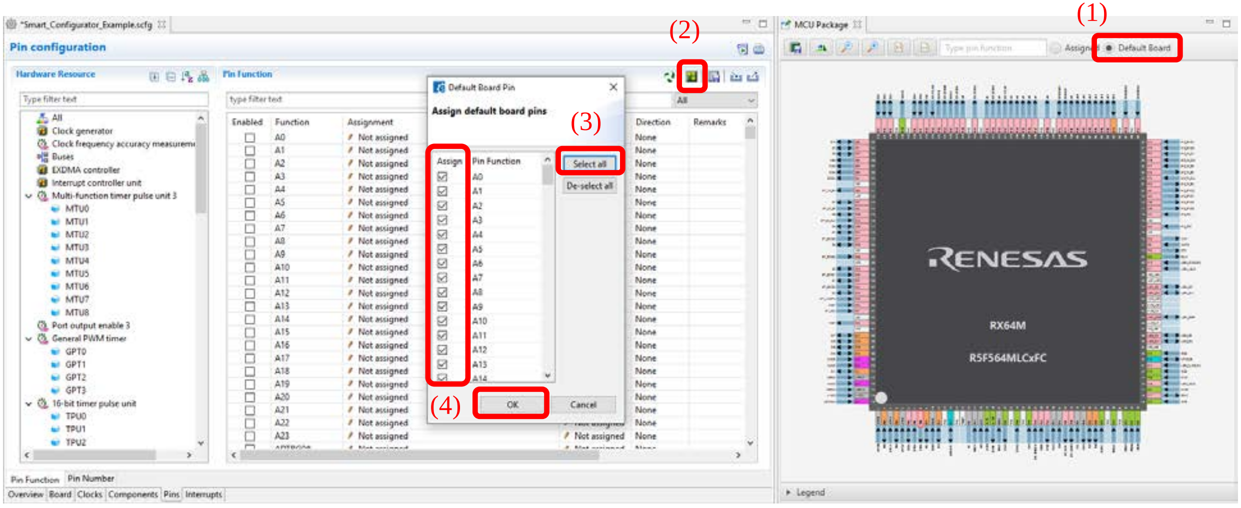
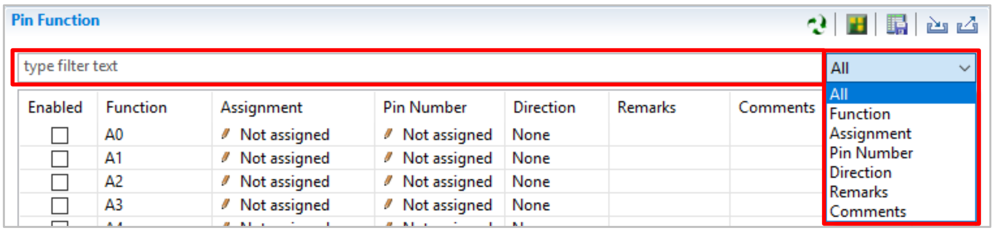


Figure 4-33 Setting for Initial Pin Configuration

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.



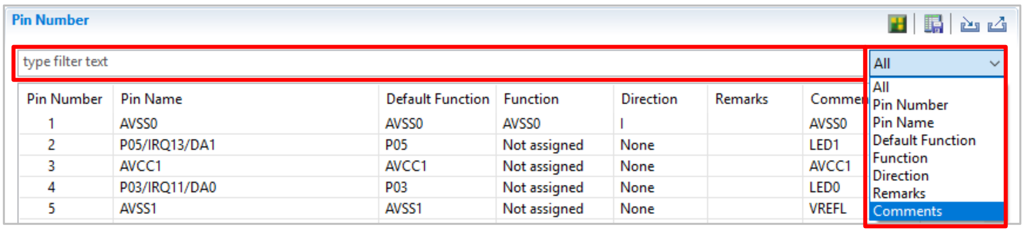
Pin Function

type filter text

All
- All
- Function
- Assignment
- Pin Number
- Direction
- Remarks
- Comments

Enabled	Function	Assignment	Pin Number	Direction	Remarks	Comments
□	A0	Not assigned	Not assigned	None		
□	A1	Not assigned	Not assigned	None		
□	A2	Not assigned	Not assigned	None		
□	A3	Not assigned	Not assigned	None		

Figure 4-34 Filter for [Pin Function] Tab



Pin Number

type filter text

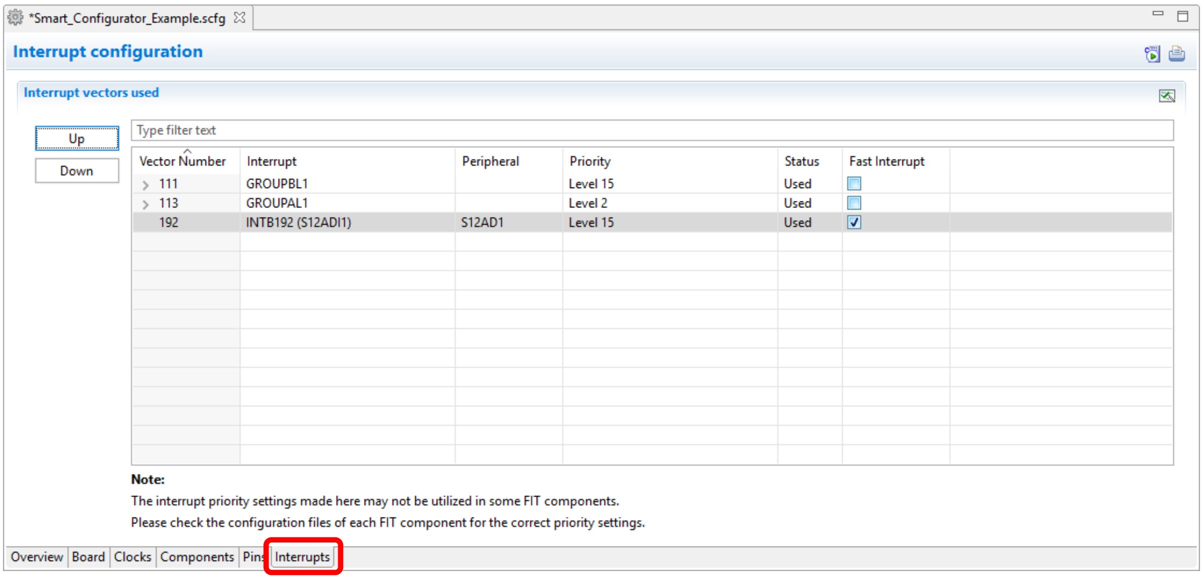
All
- All
- Pin Number
- Pin Name
- Default Function
- Function
- Direction
- Remarks
- Comments

Pin Number	Pin Name	Default Function	Function	Direction	Remarks	Comme
1	AVSS0	AVSS0	AVSS0	I		AVSS0
2	P05/IRQ13/DA1	P05	Not assigned	None		LED1
3	AVCC1	AVCC1	Not assigned	None		AVCC1
4	P03/IRQ11/DA0	P03	Not assigned	None		LED0
5	AVSS1	AVSS1	Not assigned	None		VREFL

Figure 4-35 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.



Vector Number	Interrupt	Peripheral	Priority	Status	Fast Interrupt
› 111	GROUPBL1		Level 15	Used	□
› 113	GROUPAL1		Level 2	Used	□
192	INTB192 (S12ADI1)	S12AD1	Level 15	Used	☑

Note:
The interrupt priority settings made here may not be utilized in some FIT components.
Please check the configuration files of each FIT component for the correct priority settings.

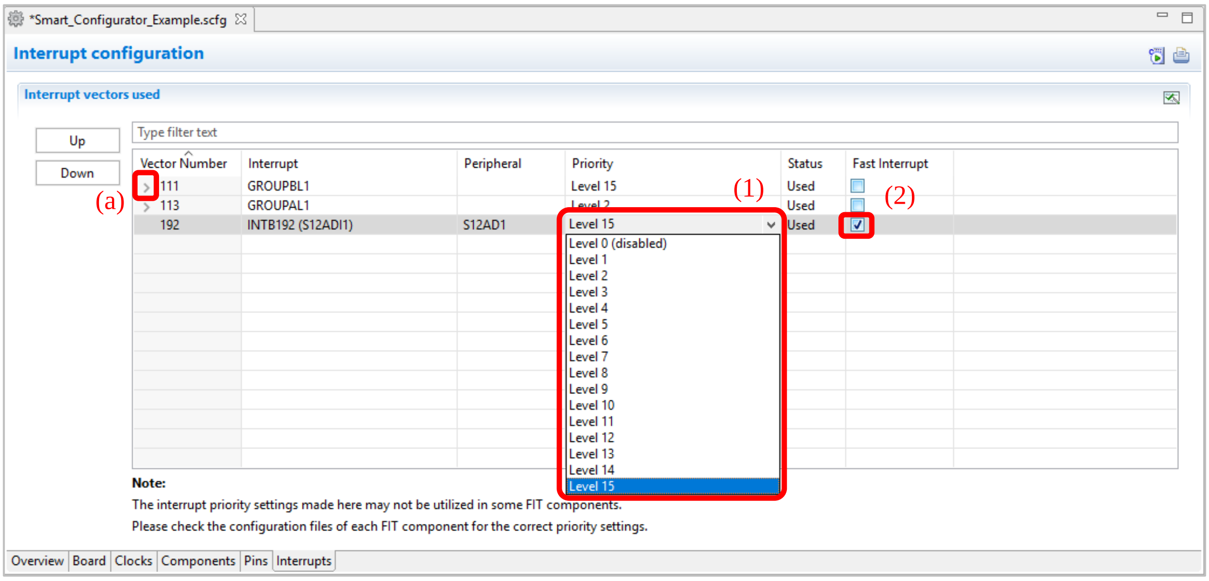
Figure 4-36 [Interrupts] Page

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.



Vector Number	Interrupt	Peripheral	Priority	Status	Fast Interrupt
› 111	GROUPBL1		Level 15	Used	□
› 113	GROUPAL1		Level 2	Used	□
192	INTB192 (S12ADI1)	S12AD1	Level 15	Used	☑

Note:
The interrupt priority settings made here may not be utilized in some FIT components.
Please check the configuration files of each FIT component for the correct priority settings.

Figure 4-37 Interrupt Settings

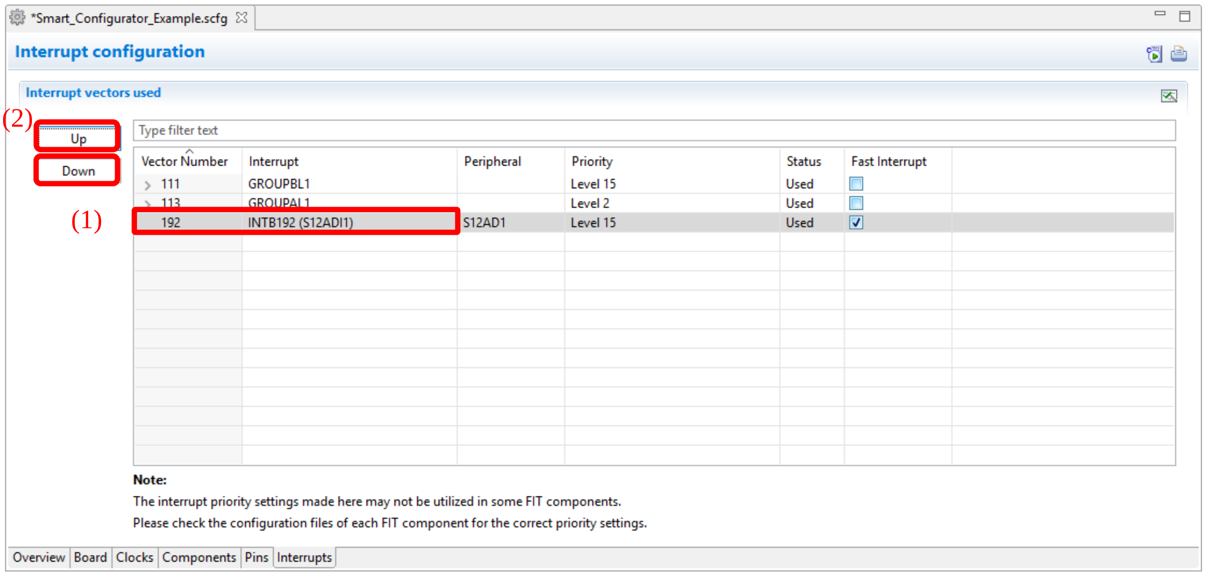
(a) Group interrupts are collapsed in the interrupt table. Click on the [› (Open)] button to expand the view and see the interrupts in the group interrupt list.

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.



Vector Number	Interrupt	Peripheral	Priority	Status	Fast Interrupt
111	GROUPBL1		Level 15	Used	□
113	GROUPAL1		Level 2	Used	□
192	INTB192 (S12ADI1)	S12AD1	Level 15	Used	☑

Note:
The interrupt priority settings made here may not be utilized in some FIT components.
Please check the configuration files of each FIT component for the correct priority settings.

Figure 4-38 Changing the Vector Number of Software Configurable Interrupt A or B

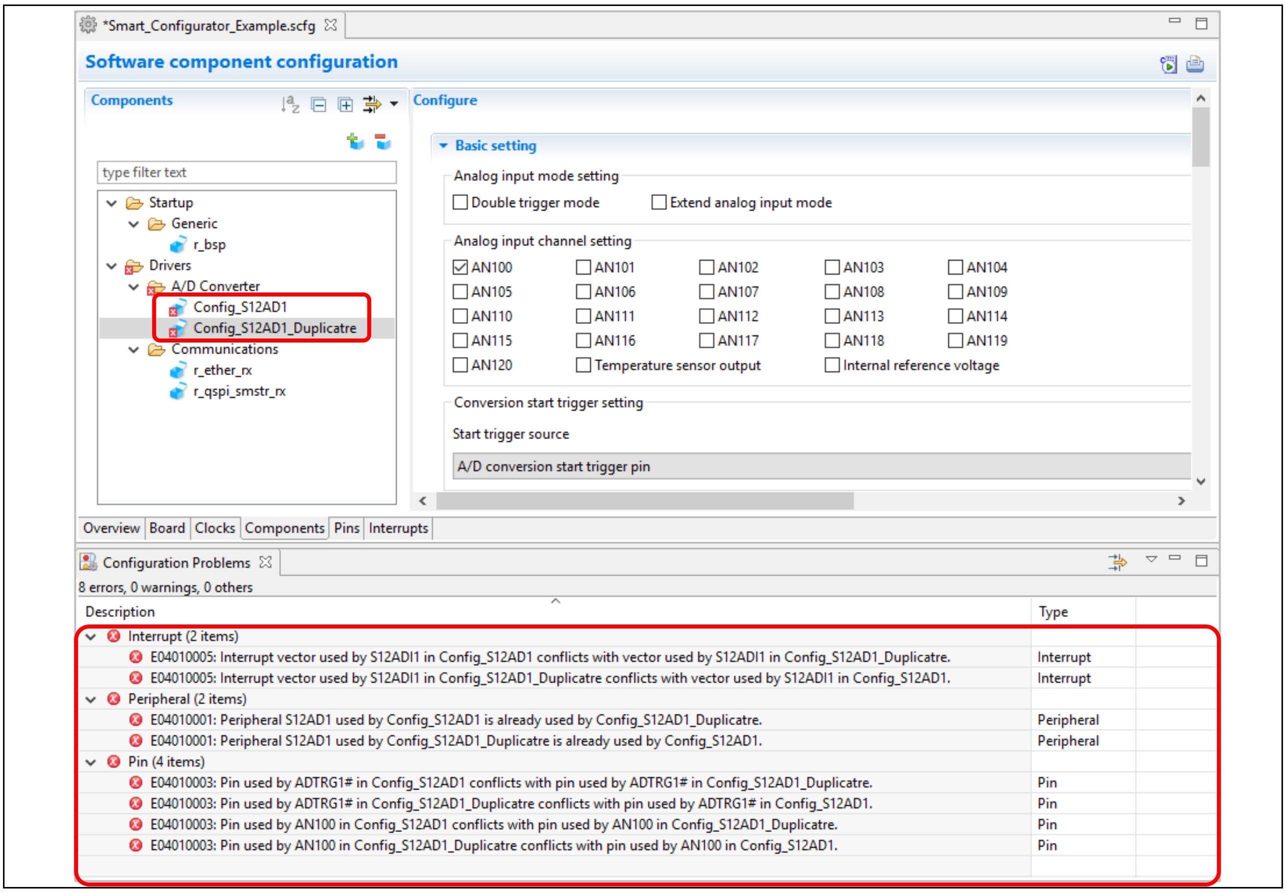
5. Managing Conflicts

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.



Analog input mode setting
□ Double trigger mode □ Extend analog input mode

Analog input channel setting

☑ AN100	□ AN101	□ AN102	□ AN103	□ AN104
□ AN105	□ AN106	□ AN107	□ AN108	□ AN109
□ AN110	□ AN111	□ AN112	□ AN113	□ AN114
□ AN115	□ AN116	□ AN117	□ AN118	□ AN119
□ AN120	□ Temperature sensor output		□ Internal reference voltage	

Conversion start trigger setting
Start trigger source
A/D conversion start trigger pin

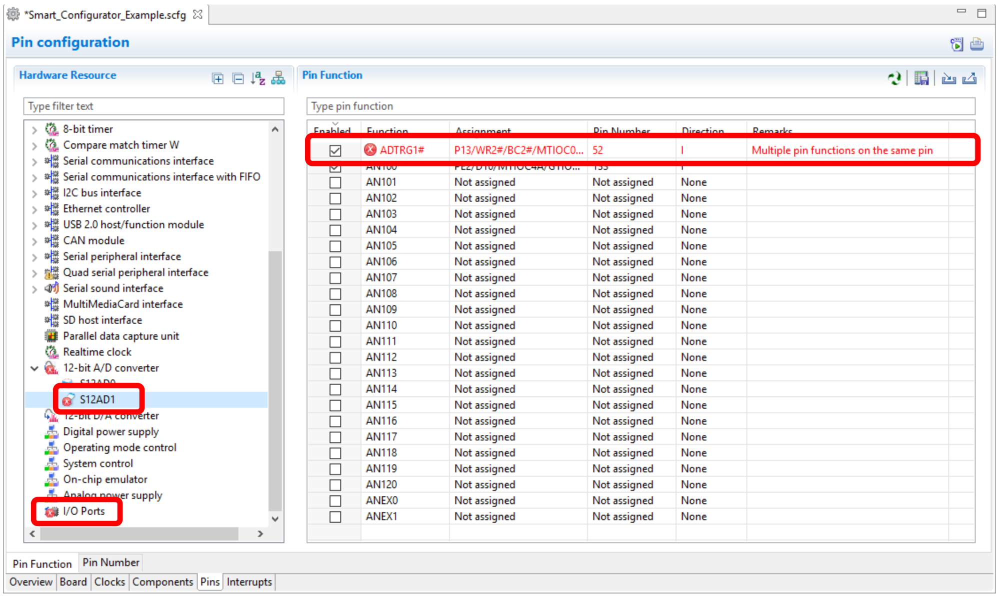
8 errors, 0 warnings, 0 others

Description	Type
Interrupt (2 items)	
E04010005: Interrupt vector used by S12ADI1 in Config_S12AD1 conflicts with vector used by S12ADI1 in Config_S12AD1_Duplicatre.	Interrupt
E04010005: Interrupt vector used by S12ADI1 in Config_S12AD1_Duplicatre conflicts with vector used by S12ADI1 in Config_S12AD1.	Interrupt
Peripheral (2 items)	
E04010001: Peripheral S12AD1 used by Config_S12AD1 is already used by Config_S12AD1_Duplicatre.	Peripheral
E04010001: Peripheral S12AD1 used by Config_S12AD1_Duplicatre is already used by Config_S12AD1.	Peripheral
Pin (4 items)	
E04010003: Pin used by ADTRG1# in Config_S12AD1 conflicts with pin used by ADTRG1# in Config_S12AD1_Duplicatre.	Pin
E04010003: Pin used by ADTRG1# in Config_S12AD1_Duplicatre conflicts with pin used by ADTRG1# in Config_S12AD1.	Pin
E04010003: Pin used by AN100 in Config_S12AD1 conflicts with pin used by AN100 in Config_S12AD1_Duplicatre.	Pin
E04010003: Pin used by AN100 in Config_S12AD1_Duplicatre conflicts with pin used by AN100 in Config_S12AD1.	Pin

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.

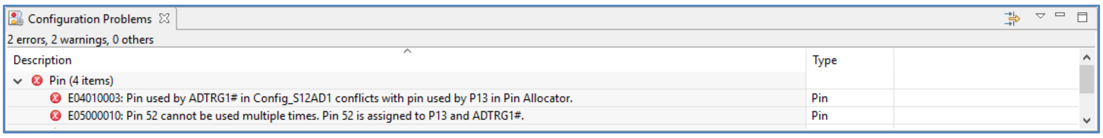


- 8-bit timer
- Compare match timer W
- Serial communications interface
- Serial communications interface with FIFO
- I2C bus interface
- Ethernet controller
- USB 2.0 host/function module
- CAN module
- Serial peripheral interface
- Quad serial peripheral interface
- Serial sound interface
- MultiMediaCard interface
- SD host interface
- Parallel data capture unit
- Realtime clock
- 12-bit A/D converter
  - S12AD0
  - S12AD1
- 12-bit D/A converter
- Digital power supply
- Operating mode control
- System control
- On-chip emulator
- Analog power supply
- I/O Ports

Enabled	Function	Assignment	Pin Number	Direction	Remarks
☑	ADTRG1#	P13/WR2#/BC2#/MTIOC0...	52	I	Multiple pin functions on the same pin
☑	AN100	PE2/D10/MTIOC4A/GTIO...	133	I	
□	AN101	Not assigned	Not assigned	None	
□	AN102	Not assigned	Not assigned	None	
□	AN103	Not assigned	Not assigned	None	
□	AN104	Not assigned	Not assigned	None	
□	AN105	Not assigned	Not assigned	None	
□	AN106	Not assigned	Not assigned	None	
□	AN107	Not assigned	Not assigned	None	
□	AN108	Not assigned	Not assigned	None	
□	AN109	Not assigned	Not assigned	None	
□	AN110	Not assigned	Not assigned	None	
□	AN111	Not assigned	Not assigned	None	
□	AN112	Not assigned	Not assigned	None	
□	AN113	Not assigned	Not assigned	None	
□	AN114	Not assigned	Not assigned	None	
□	AN115	Not assigned	Not assigned	None	
□	AN116	Not assigned	Not assigned	None	
□	AN117	Not assigned	Not assigned	None	
□	AN118	Not assigned	Not assigned	None	
□	AN119	Not assigned	Not assigned	None	
□	AN120	Not assigned	Not assigned	None	
□	ANEX0	Not assigned	Not assigned	None	
□	ANEX1	Not assigned	Not assigned	None	

Figure 5-2 Pin Conflicts

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

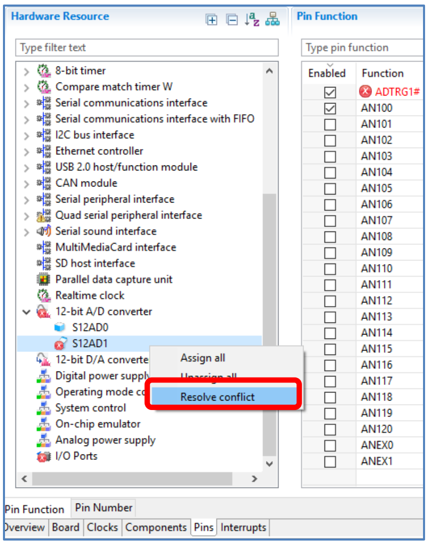


2 errors, 2 warnings, 0 others

Description	Type
Pin (4 items)	
E04010003: Pin used by ADTRG1# in Config_S12AD1 conflicts with pin used by P13 in Pin Allocator.	Pin
E05000010: Pin 52 cannot be used multiple times. Pin 52 is assigned to P13 and ADTRG1#.	Pin

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].



- 8-bit timer
- Compare match timer W
- Serial communications interface
- Serial communications interface with FIFO
- I2C bus interface
- Ethernet controller
- USB 2.0 host/function module
- CAN module
- Serial peripheral interface
- Quad serial peripheral interface
- Serial sound interface
- MultiMediaCard interface
- SD host interface
- Parallel data capture unit
- Realtime clock
- 12-bit A/D converter
  - S12AD0
  - S12AD1
- 12-bit D/A converte
- Digital power suppl
- Operating mode co
- System control
- On-chip emulator
- Analog power supply
- I/O Ports

Enabled	Function
☑	ADTRG1#
☑	AN100
□	AN101
□	AN102
□	AN103
□	AN104
□	AN105
□	AN106
□	AN107
□	AN108
□	AN109
□	AN110
□	AN111
□	AN112
□	AN113
□	AN114
□	AN115
□	AN116
□	AN117
□	AN118
□	AN119
□	AN120
□	ANEX0
□	ANEX1

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.

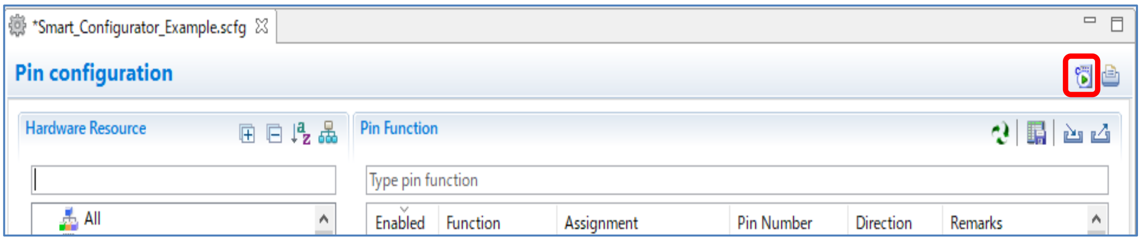


Figure 6-1 Generating a Source File

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.

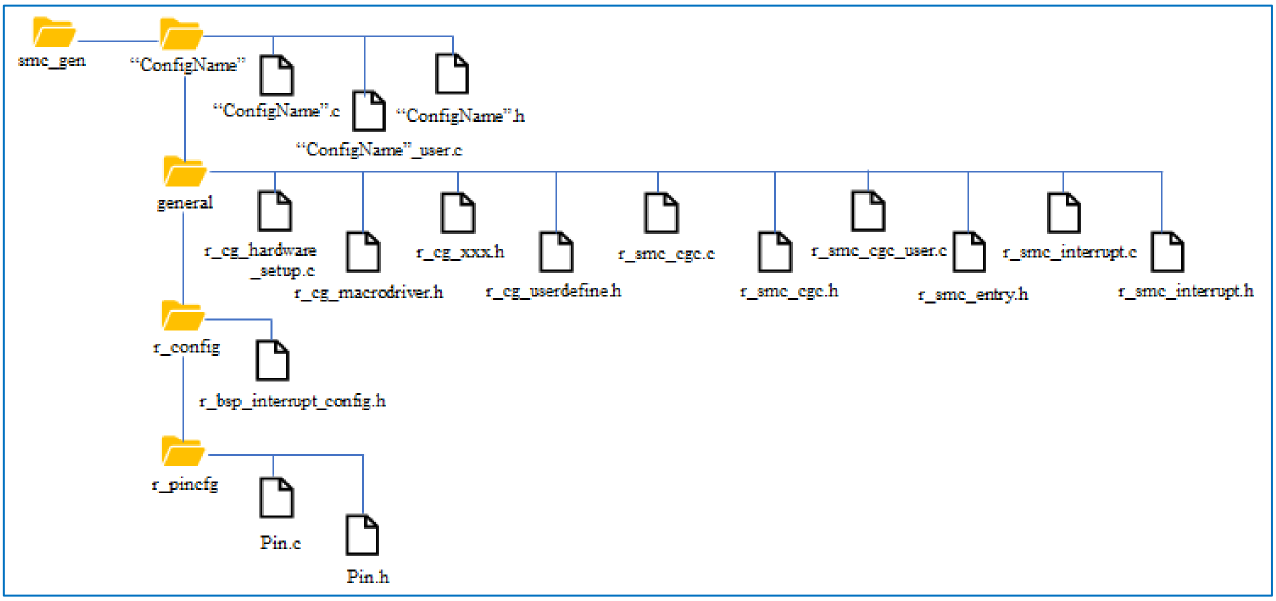


Figure 6-2 Configuration of Generated Files and File Names

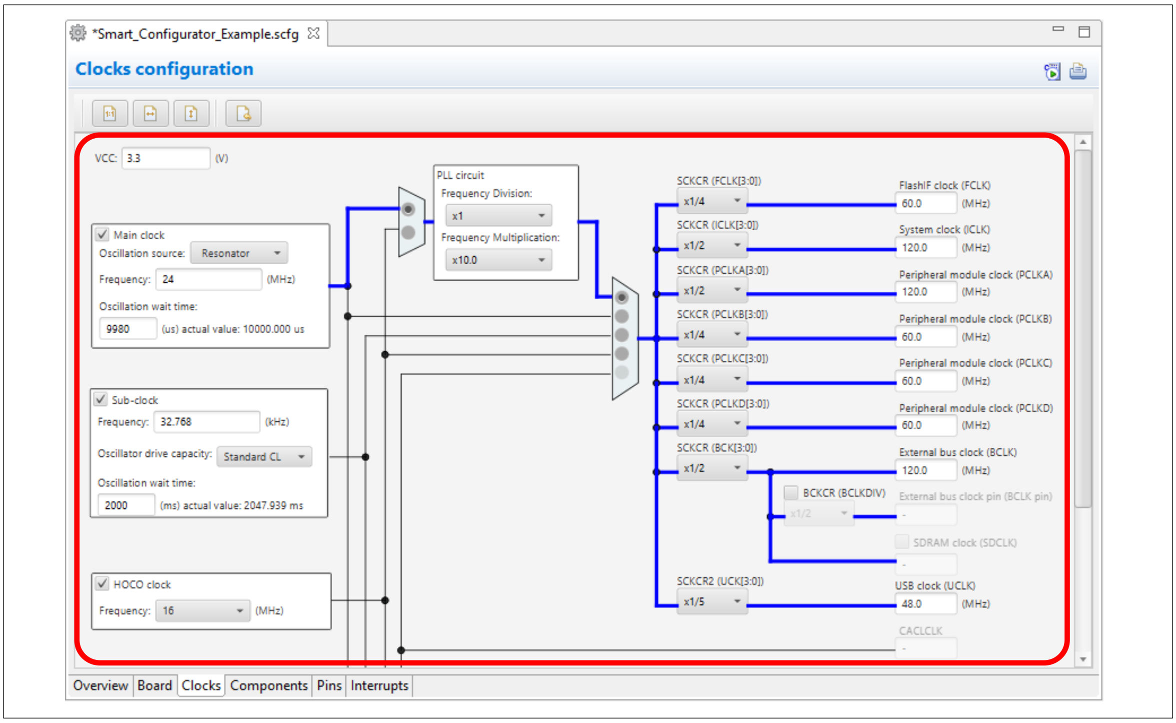
Table 6-1 Description of Generated File

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

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.



VCC: 3.3 (V)

☑ Main clock
Oscillation source: Resonator
Frequency: 24 (MHz)
Oscillation wait time:
9980 (us) actual value: 10000.000 us

☑ Sub-clock
Frequency: 32.768 (kHz)
Oscillator drive capacity: Standard CL
Oscillation wait time:
2000 (ms) actual value: 2047.939 ms

☑ HOCO clock
Frequency: 16 (MHz)

PLL circuit
Frequency Division: x1
Frequency Multiplication: x10.0

SCKCR (FCLK[3:0]) x1/4 → FlashIF clock (FCLK) 60.0 (MHz)
SCKCR (ICLK[3:0]) x1/2 → System clock (ICLK) 120.0 (MHz)
SCKCR (PCLKA[3:0]) x1/2 → Peripheral module clock (PCLKA) 120.0 (MHz)
SCKCR (PCLKB[3:0]) x1/4 → Peripheral module clock (PCLKB) 60.0 (MHz)
SCKCR (PCLKC[3:0]) x1/4 → Peripheral module clock (PCLKC) 60.0 (MHz)
SCKCR (PCLKD[3:0]) x1/4 → Peripheral module clock (PCLKD) 60.0 (MHz)
SCKCR (BCK[3:0]) x1/2 → External bus clock (BCLK) 120.0 (MHz)
□ BCKCR (BCLKDIV) x1/2 → External bus clock pin (BCLK pin) -
□ SDRAM clock (SDCLK) -
SCKCR2 (UCK[3:0]) x1/5 → USB clock (UCLK) 48.0 (MHz)
CACLCLK -

Figure 6-3 Clocks Source Configuration

Table 6-2 Clock Source File Description

Folder	File	Macros/Functions	Description
general	*r_cg_cgc.c*	*R_CGC_Create*	This API function initializes clocks other than the selected clock source. *R_Systeminit* in *r_cg_hardware_setup.c* will call this function before entering *main()* function.
	r_cg_cgc.h	Macros related to clocks	These macros are for clock initialization in *R_CGC_Create*.
	r_cg_cgc_user.c	*R_CGC_Create_UserInit*	This API function is used to add code to *R_CGC_Create* 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}	*{ConfigName}.c*	*R_ConfigName_Create*	This API function initializes pins used by this component. *R_Systeminit* in *r_cg_hardware_setup.c* will call this function before entering *main()* function.

(2) Reference pins initialization codes

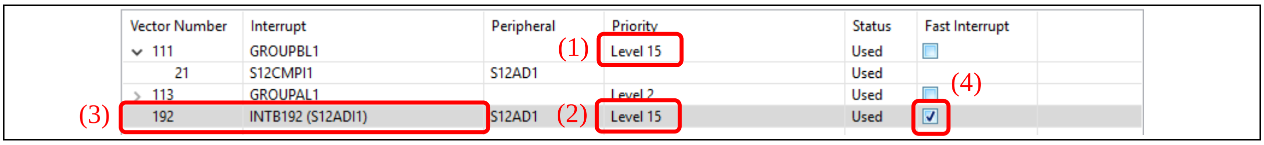
Refer to *Pin.c* in the *\src\smc_gen\r_pincfg* folder for the initialization code of all pin functions set on the [Pins] page (except I/O ports).

Table 6-4 Reference File for Initialization of All Pins

Folder	File	Function	Description
r_pincfg	*Pin.c*	*R_Pins_Create*	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	Used	□
21	S12CMPI1	S12AD1		Used	
113	GROUPAL1		Level 2	Used	□
192	INTB192 (S12ADI1)	S12AD1	Level 15	Used	☑

Figure 6-4 Interrupt Configuration

Table 6-5 Interrupt Generation File Description

No	Item	Folder	File	Description
(1)	Priority	**general**	*r_smc_interrupt.c*	This interrupt priority level setting is for group interrupts (Note2). It is initialized in *R_Interrupt_Create* of this file. *R_Systeminit* in *r_cg_hardware_setup.c* will call this function before entering *main()* 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 *R_ConfigName_Create* of this file. *R_Systeminit* in *r_cg_hardware_setup.c* will call this function before entering *main()* function.
(1) (2)	Priority	**general**	*r_smc_interrupt.h*	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**	*r_bsp_interrupt_config.h*	Vector number of software configurable interrupts A and B (Note2) in the [Interrupts] tabbed page will be mapped in this file and handled by *r_bsp*.
(4)	Fast Interrupt	**general**	*r_smc_interrupt.c*	Fast interrupt setting will be initialized in *R_Interrupt_Create* of this file. *R_Systeminit* in *r_cg_hardware_setup.c* will call this function before entering *main()* function.
			r_smc_interrupt.h	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 Adding Custom Code of FIT

When [FIT] is selected as the component type, the configuration options are set in r_xxx_config.h in the folder r_config. For the settings of the configuration options, refer to the application note (in the doc folder) on the FIT module (r_xxx) which you have added to the project tree.

If the target file already exists, the existing contents of the file are protected when source code is output.

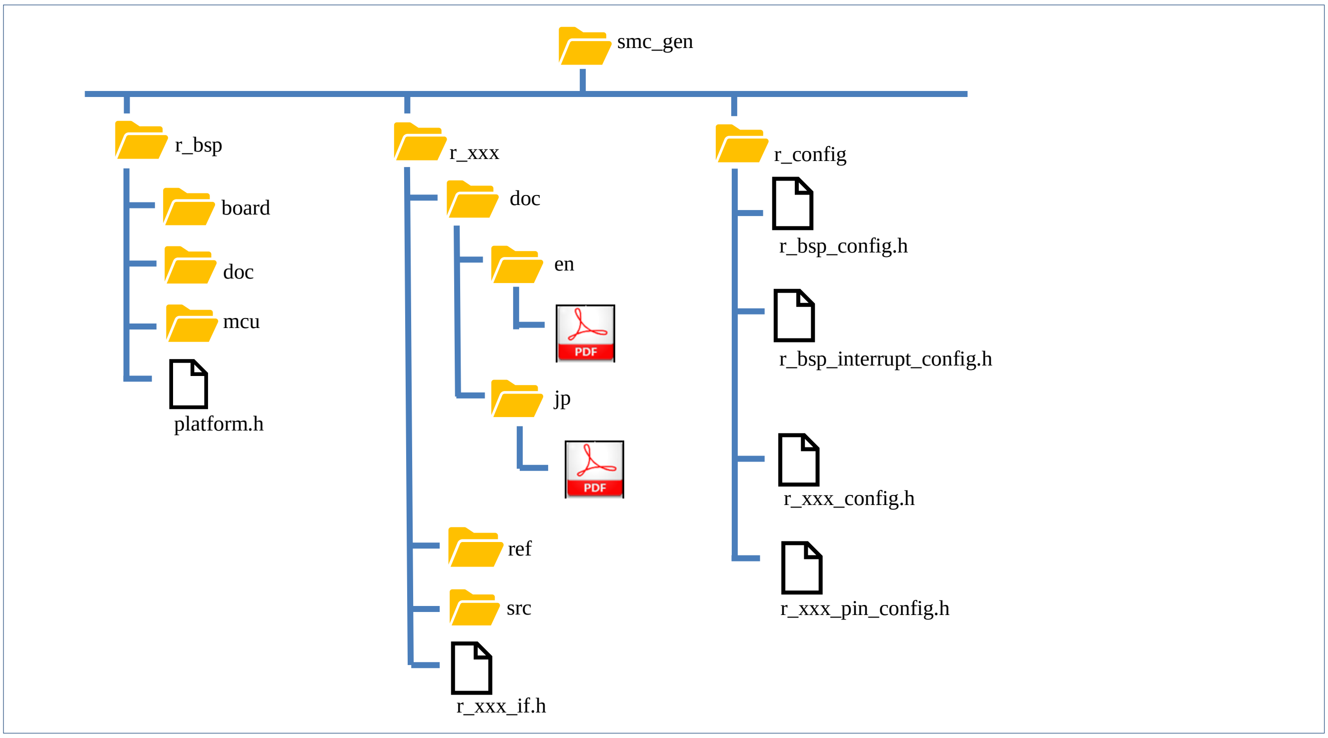


Figure 7-1 Tree Structure of Directories and Files for a FIT Module

7.2 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 */
}
```

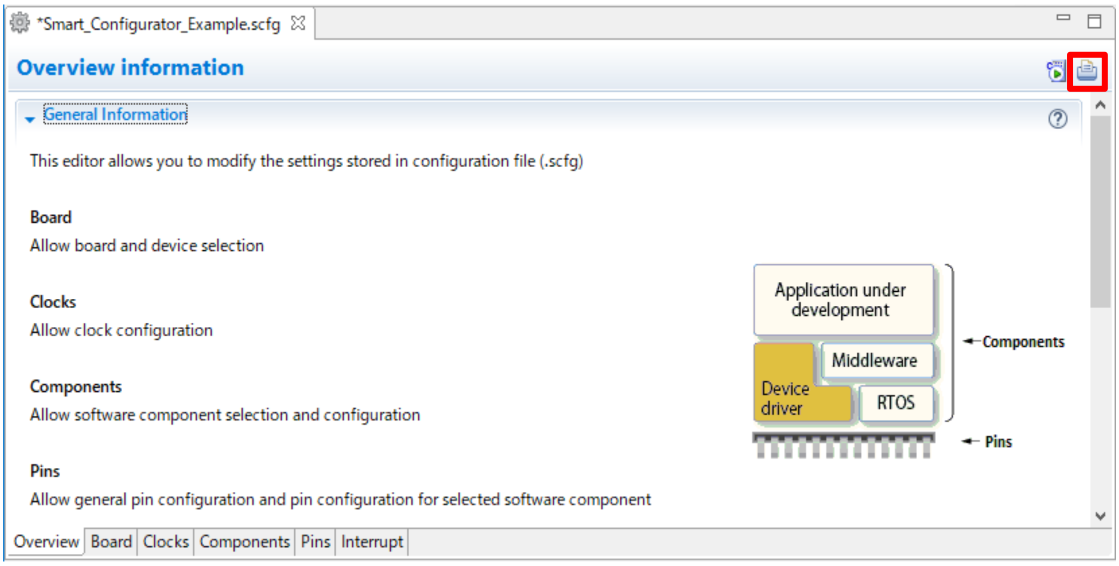
```
/******************************************************************************
* Function Name: r_Config_CMT3_cmi3_interrupt
* Description  : This function is CMI3 interrupt service routine
* Arguments    : None
* Return Value : None
******************************************************************************/

#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.



This editor allows you to modify the settings stored in configuration file (.scfg)

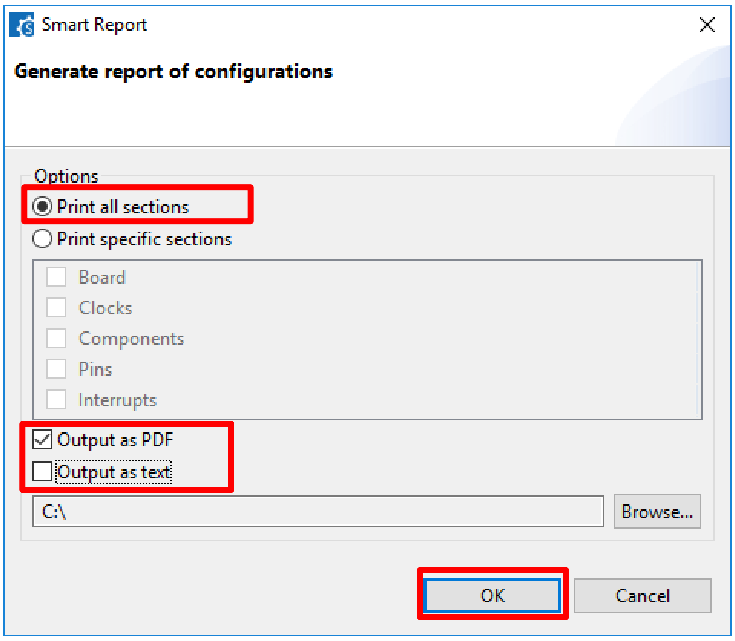
Board
Allow board and device selection

Clocks
Allow clock configuration

Components
Allow software component selection and configuration

Pins
Allow general pin configuration and pin configuration for selected software component

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



Generate report of configurations

Options

● Print all sections

○ Print specific sections

□ Board
□ Clocks
□ Components
□ Pins
□ Interrupts

☑ Output as PDF

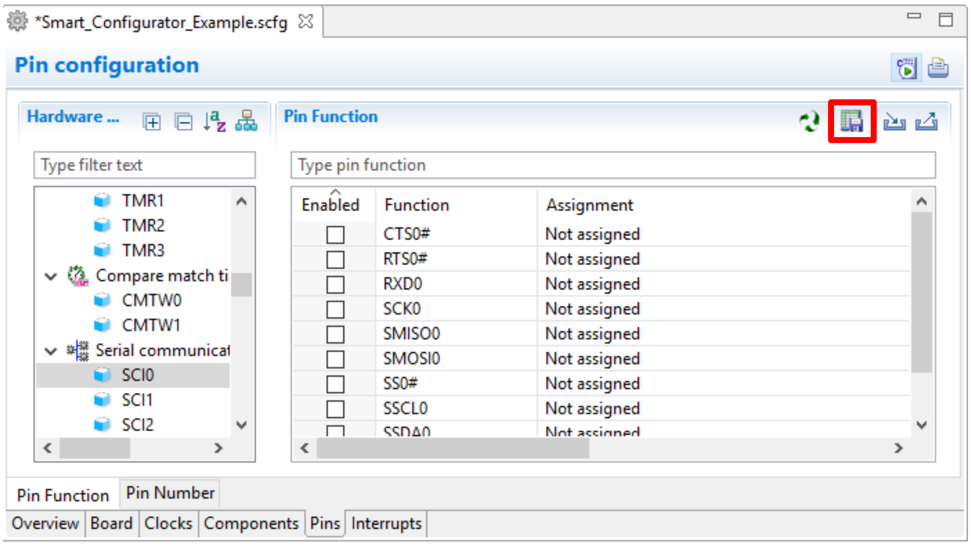
□ Output as text

C:\

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 the list to .csv file)] on the [Pins] page of the Smart Configurator view.



Enabled	Function	Assignment
□	CTS0#	Not assigned
□	RTS0#	Not assigned
□	RXD0	Not assigned
□	SCK0	Not assigned
□	SMISO0	Not assigned
□	SMOSI0	Not assigned
□	SS0#	Not assigned
□	SSCL0	Not assigned
□	SSDA0	Not assigned

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 Package View to external image file)] button of the [MCU Package] view.

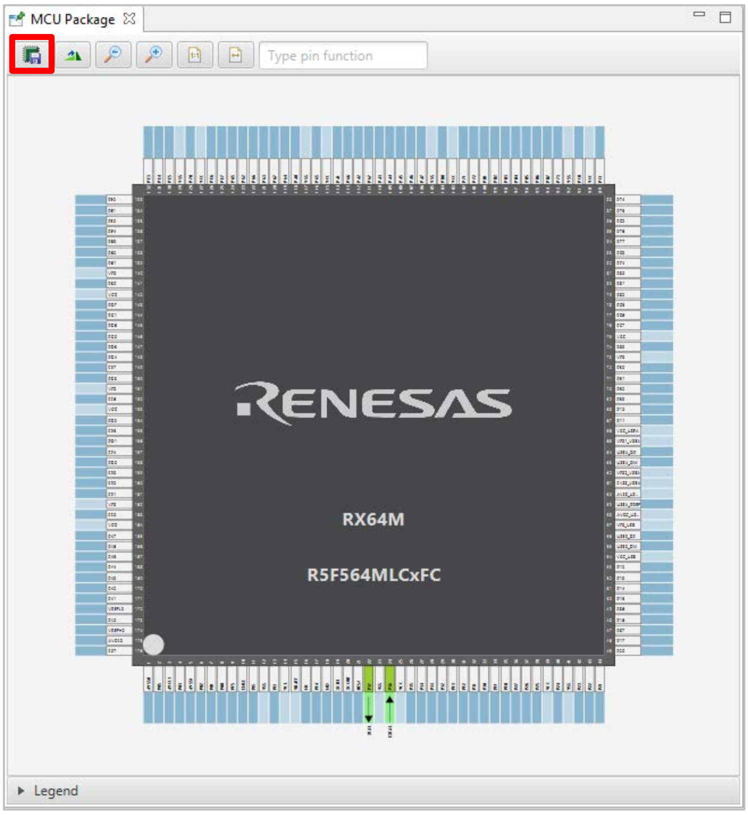


Figure 9-4 Outputting a Figure of MCU Package (in png Format)

10. Help

10.1 Help

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

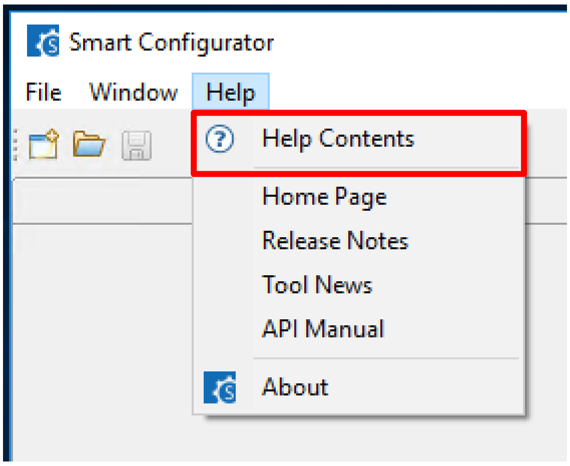
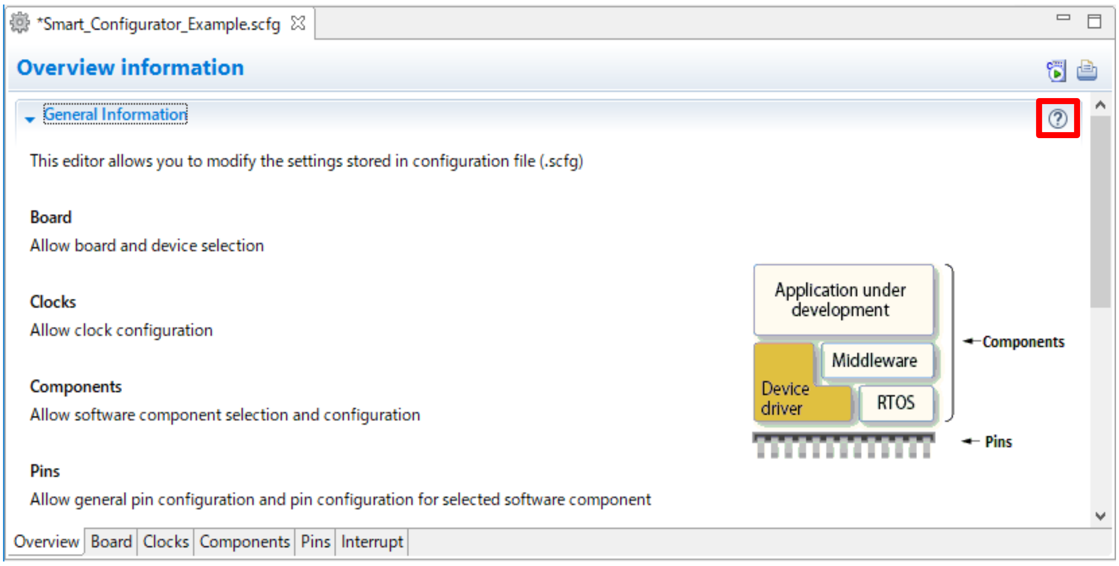


Figure 10-1 Help Menu

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



This editor allows you to modify the settings stored in configuration file (.scfg)

Board
Allow board and device selection

Clocks
Allow clock configuration

Components
Allow software component selection and configuration

Pins
Allow general pin configuration and pin configuration for selected software component

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/

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jan 25, 2019	-	First edition issued
1.10	Sep 09, 2019	4	1.2 Features updated, 1.3 Software Components added
		8	3.3.1 Downloading FIT modules added 3.3.2 Creating a New Configuration File updated
		11	3.4.1 Main menu updated
		19	4.3.2 Adding FIT drivers or middleware added
		26	4.3.7 Setting of the FIT software components added
		27	4.3.8 Changing the version of the FIT software components added
		29	0 Sample program sign of FIT icon added
		30	0 Configure general setting of component added
		45	7.1 Adding Custom Code of FIT added
		50	10.1 Help updated

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

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

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

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

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

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

4. Clock Signals

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

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

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

   — The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.