RX Family

Example of Using the Pulse Width Measurement Function with MTU2/MTU3

Introduction

This application note describes the operation of the pulse width measurement function by using MTU2/MTU3.

RX66T Group MCUs are equipped with the Multi-Function Timer Pulse Unit 3 (MTU3d), which can be used with the input capture function to measure the width of the pulse that is input to the input capture input pins.

The descriptions in this application note target RX Family devices equipped with the MTU2 or MTU3. The operation has been confirmed with the RX66T Group. When using this application note with Renesas MCUs other than the RX66T Group, careful evaluation is recommended after making modifications to comply with the alternate MCU.

Target Devices

RX Family devices with the MTU2 or MTU3

Confirmed Devices

RX66T Group

The Multi-Function Timer Pulse Unit 3 is referred to as "MTU" throughout this document.

Contents

1. Pulse Width Measurement

The MTU's input capture function is used to measure the width of the high level (referred to as "high width") of a detected external trigger pulse. The measurement starts at the rising edge of an input pulse, and the high width is calculated at the falling edge.

The overflow interrupt processing in the MTUn.TCNT register (n = 0 to 4, 6, 7, 9) and MTU5.TCNTm register (m = U, V, W) counts the number of overflows. After that, the input capture interrupt processing in MTUn (n = 0 to 6, 7, 9) calculates the pulse width based on the number of overflows and the input capture register value.

Pulse width calculation formula: Resolution*[1] x (number of overflows x 10000h + input capture register value)

Note: 1. MTU count clock cycle

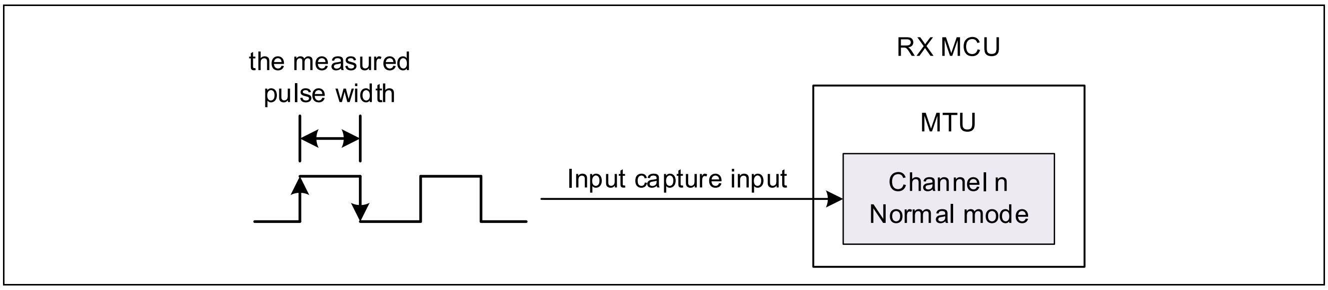


Figure 1.1 Pulse Width Measurement

For details, refer to section 3.2, Operation of Pulse Width Measurement, in this application note.

2. Operation Confirmation Conditions

The operation of the sample code described in this application note has been confirmed under the conditions listed in the table below.

Table 2.1 Operation Confirmation Environments

Item	Description
MCU	R5F566TEADFP (included in Renesas Starter Kit for RX66T)
Operating frequency	Main clock: 8 MHz PLL: 160 MHz (main clock x 1/1 x 20) HOCO: Stopped LOCO: Stopped System clock (ICLK) 160 MHz (PLL x 1/1) Peripheral Module clock A (PCLKA): 80 MHz (PLL x 1/2) Peripheral module clock B (PCLKB): 40 MHz (PLL x 1/4) Peripheral module clock C (PCLKC): 160 MHz (PLL x 1/1) Peripheral Module clock D (PCLKD): 40 MHz (PLL x 1/4) FlashIF clock (FCLK): 40 MHz (PLL x 1/4)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics e² studio Version 2022-10
C compiler*1	Renesas Electronics C/C++ Compiler Package for RX Family V3.04.00
	Compiler options The integrated development environment default settings are used.
RX Smart Configurator	V2.15.0
Board support package (r_bsp)	V7.20
Endian	Little endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample code version	V1.00
Board	Renesas Starter Kit for RX66T (Product number: RTK50566T0CxxxxxBE)
Emulator	E2-Lite

Note: 1. Import the same version of the toolchain (C compiler) as specified in the original project. If the same toolchain is not located in the import destination, the toolchain cannot be selected, and an error will occur.
Check the toolchain selection status on the project settings screen.

Refer to FAQ 3000404 for setting methods.

FAQ 3000404: 'Program "make" not found in PATH' error when attempting to build an imported project (e² studio)

3. MTU Sample Codes

3.1 Common

3.1.1 Sample Code List

This application note provides the following sample codes created with the Smart Configurator.

Sample codes can be downloaded from the Renesas Electronics website.

Table 3.1 MTU Sample Code List

Name	Description	Ref.
Operation of Pulse Width Measurement r01an6748_rx66t_mtu_pulse_width.zip	• Normal mode • Measuring a pulse width by using the input capture function	3.2

3.1.2 Folder Structure

The main folder structure of a sample code is as follows.

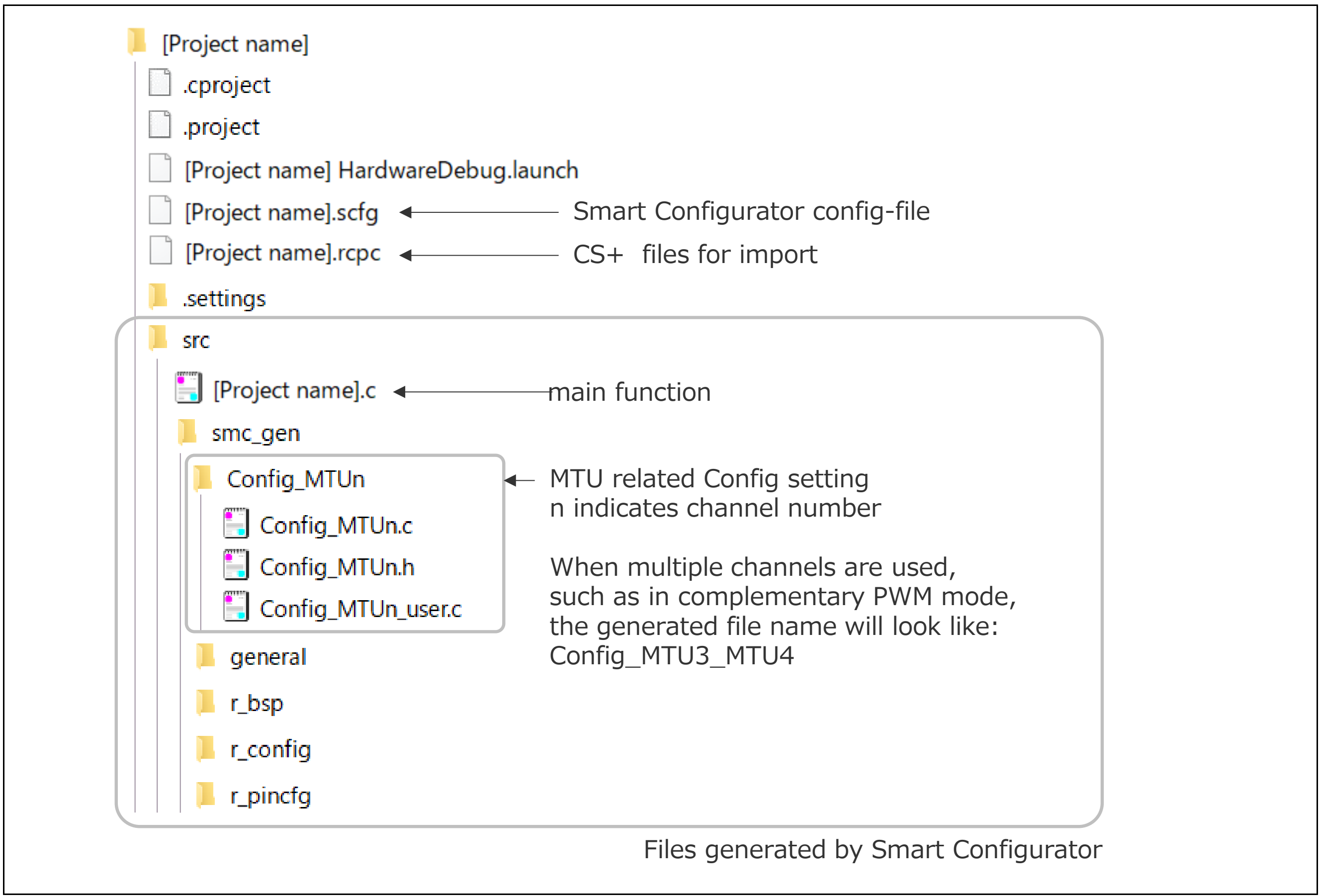


Figure 3.1 MTU Folder Structure

3.1.3 File Structure

The main file structure of a sample code is as follows.

Table 3.2 MTU File Structure

File Name	Description
[Project name].c	main Function This is the main function. The Smart Configurator generates an empty function. The necessary processing for each sample code is described here.
Config_MTUn.c*1	R_Config_MTUn_Create function This is the MTU's initialization function. The initialization function based on the settings in the Smart Configurator is generated by the Smart Configurator. The call for this function is generated by the Smart Configurator. This function is called in the R_SystemInit function executed before the main function.
	R_Config_MTUn_Start function This is the MTU's count start function. This function is generated by the Smart Configurator. In the sample codes, this function is called from the main function.
	R_Config_MTUn_Stop function This is the MTU's count stop function. This function is generated by the Smart Configurator. This function is not used in the sample codes.
Config_MTUn_user.c*1	r_Config_MTUn_Create_UserInit function This is the MTU's user initialization function. The Smart Configurator generates an empty function. The necessary processing for each sample code is described here. This is the last function to be called in the R_Config_MTUn_Create function generated by the Smart Configurator.
	r_Config_MTUn_[interrupt name]_interrupt function This is the interrupt handler function. The Smart Configurator generates an empty function. The necessary processing for each sample code is described here.
Config_MTUn.h*1	This is the header file that defines MTU related functions. This file is included in the r_smc_entry.h file generated by the Smart Configurator. To use MTU related functions, be sure to include the r_smc_entry.h file.

Note: 1. n indicates a channel number.

3.1.4 Adding Components

The sample code uses the Smart Configurator to add the MTU as described below.

Table 3.3 Adding Components

Item	Description
Component	Refer to the section for each sample code ((1) in the figure below).
Configuration name	Sample codes use the default setting name.
Operation	Refer to the section for each sample code ((2) in the figure below).
Resource	Refer to the section for each sample code ((3) in the figure below).

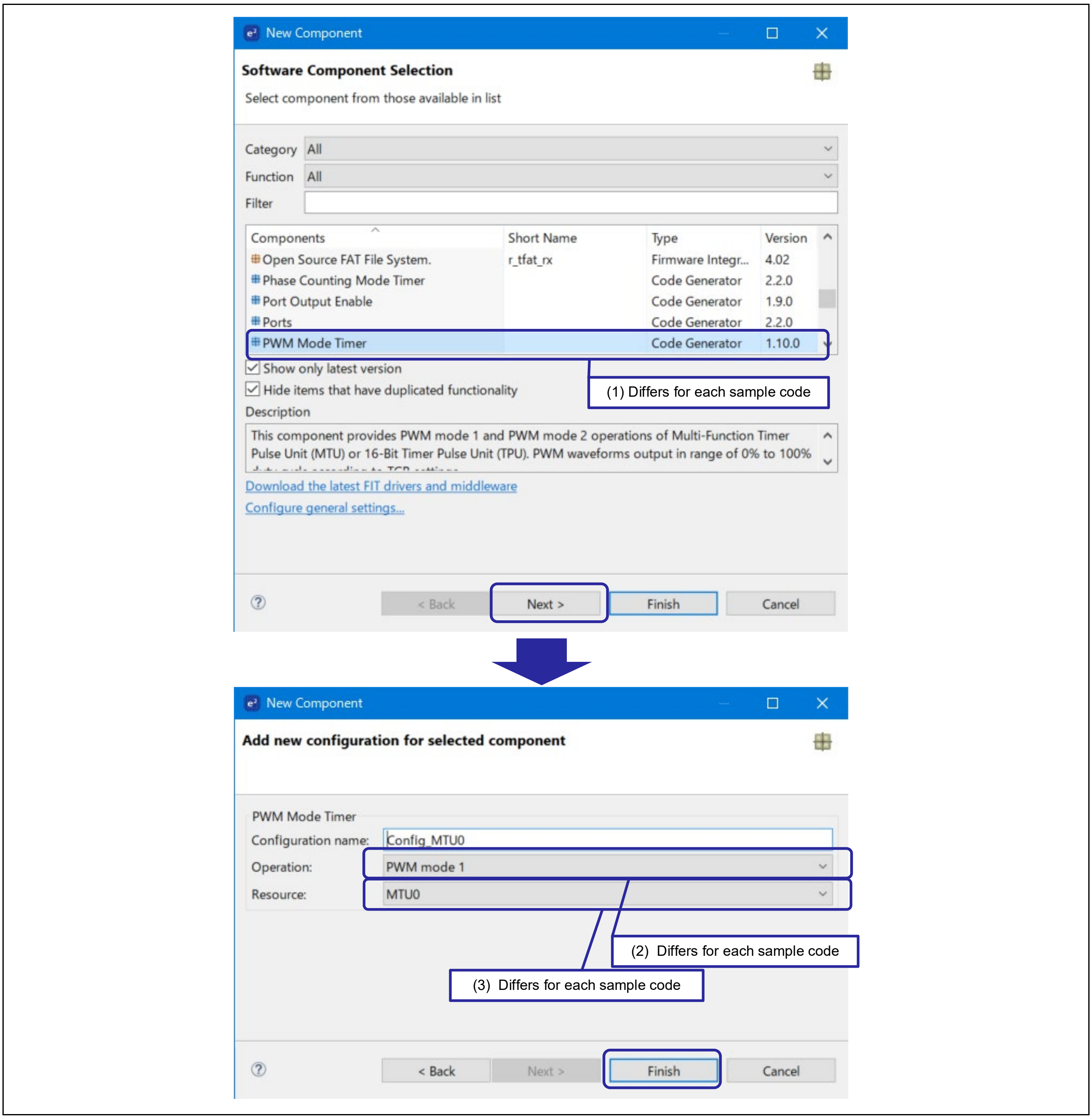


Figure 3.2 Adding Components

3.1.5 Pin Settings

Figure 3.3 shows an example of pin settings using the Smart Configurator.

Configure the pins after setting the MTU. For MTU settings, refer to "Smart Configurator Settings" for each sample code.

Pin settings are carried out in the R_Config_MTUn_Create function generated by the Smart Configurator.

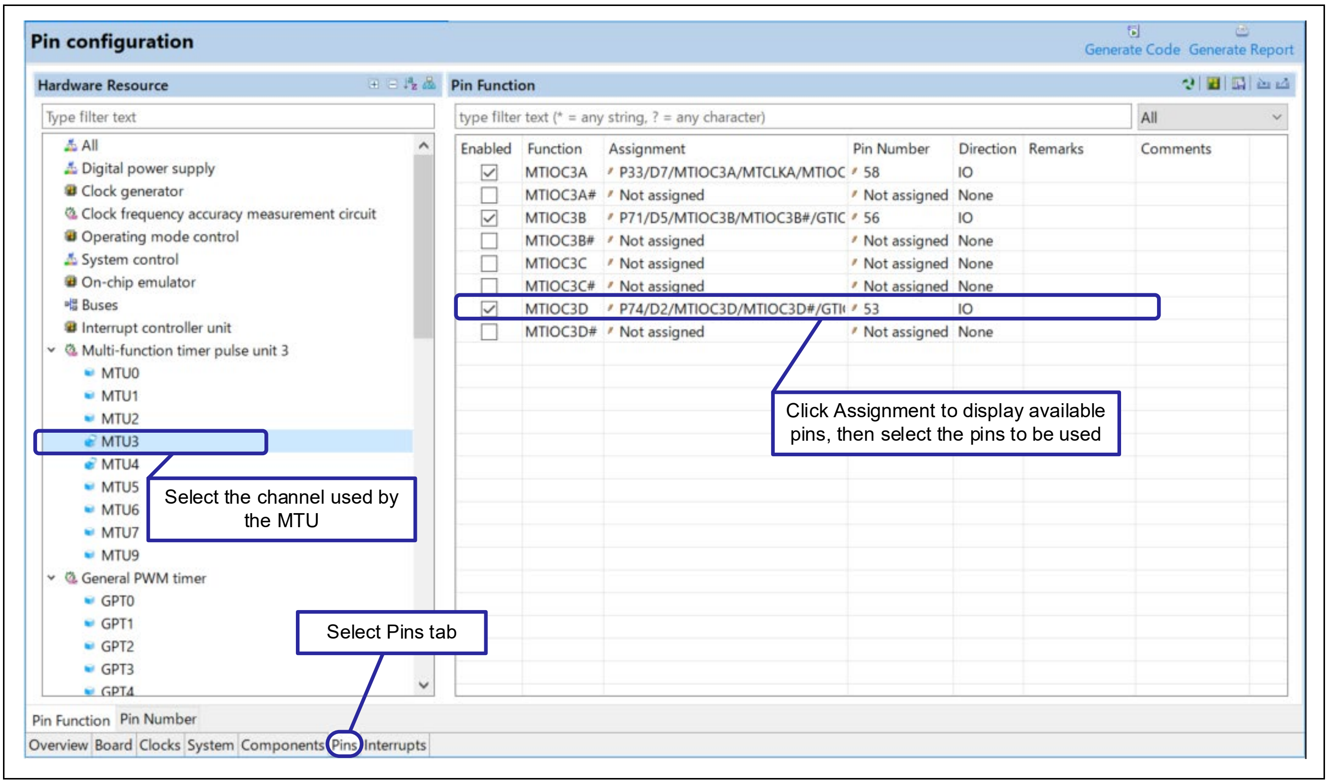


Figure 3.3 Pin Settings

3.1.6 Interrupt Settings

Figure 3.4 shows an example of interrupt settings using the Smart Configurator. For details on software configurable interrupt A, refer to section 14.4.5.1, Software Configurable Interrupt A, in the RX66T Group User's Manual: Hardware.

Configure interrupts after configuring the MTU settings. For MTU settings, refer to "Smart Configurator Settings" for each sample code.

Interrupt settings can be configured in the R_Config_MTUn_Create function, R_Config_MTUn_Start function, and R_Config_MTUn_Stop function, all of which are generated by the Smart Configurator.

The interrupt handler function is created with the name r_Config_MTUn_[interrupt name]_interrupt in the Config_MTUn_user.c file generated by the Smart Configurator.

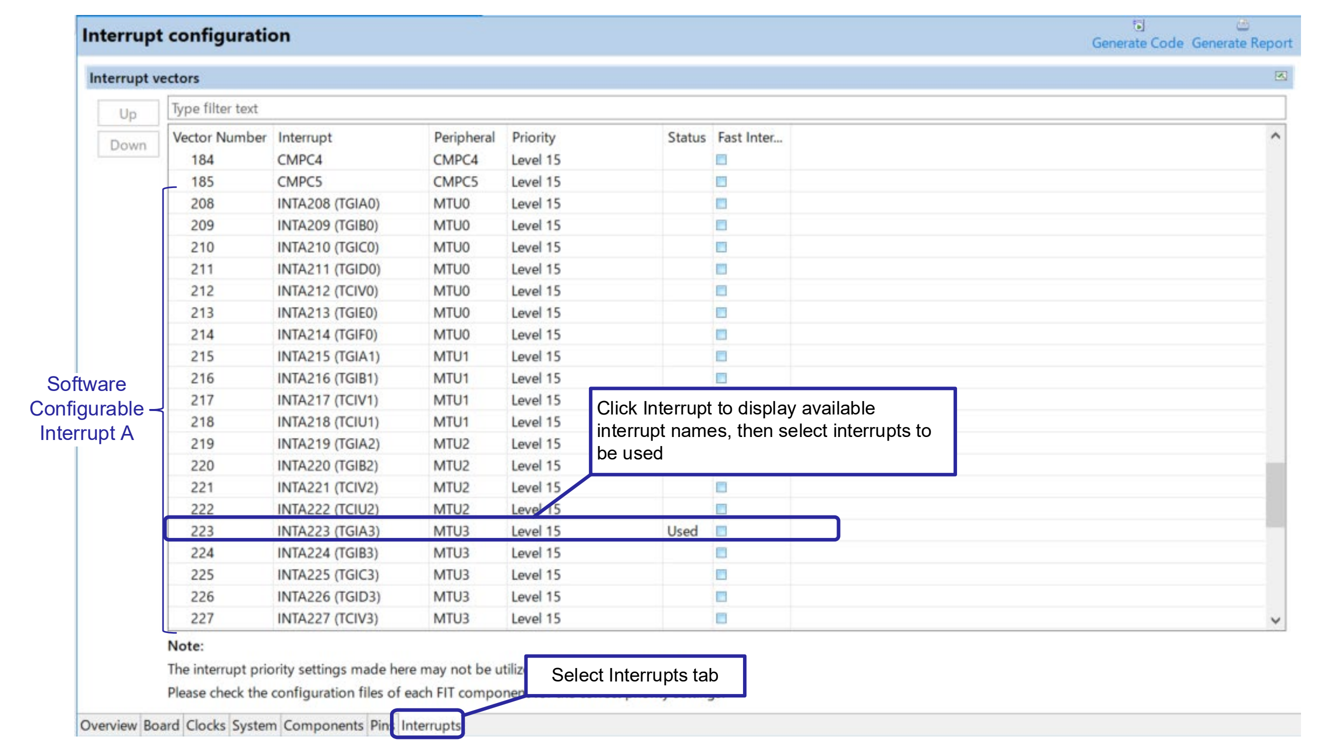


Figure 3.4 Interrupt Settings

3.2 Operation of Pulse Width Measurement

- Target sample code file name: r01an6748_rx66t_mtu_pulse_width.zip

3.2.1 Overview

The MTU’s input capture function is used to calculate the period from the rising edge of a pulse (which is input to the MTIOC1A pin) to the next falling edge.

Overflow interrupts by the MTU1.TCNT register are used to count the number of overflows of the MTU1.TCNT register. If a pulse with an overflow count exceeding 65,535 is input, an error signal is output and the measurement is stopped.

The input capture A interrupt processing of MTU1 calculates the pulse width based on the number of overflows and the value of the MTU1.TGRA register.

- MTU1 details
  - Resolution: Approximately 100 ns
  - Maximum measurable period: Approximately 429 s
- Pulse width calculation formula: 100 ns x (number of overflows x 10000h + MTU1.TGRA)

The following list provides the MTU and port settings used in the sample code.

- MTU1 (channel 1)
  - Use normal mode timer
  - Use channel 1
  - Timer counter clock = 10 MHz (PCLKC/16)
  - Use TGRA as the input capture register
    - Timer counter clear source = TGRA input capture
  - Input capture at both edges of the MTIOC1A pin input
  - Use the noise filter for the MTIOC1A pin
    - Noise filter clock = 160 MHz (PCLKC/1)
  - Enable the TGRA input capture interrupt
    - Priority: Level 3
  - Enable overflow interrupts
    - Priority: Level 4
- PORT
  - Use P95 as a general purpose I/O port

Set in Smart Configurator.

Refer to section 3.2.3 for details on the setting method.

The structure of this sample code is shown below.

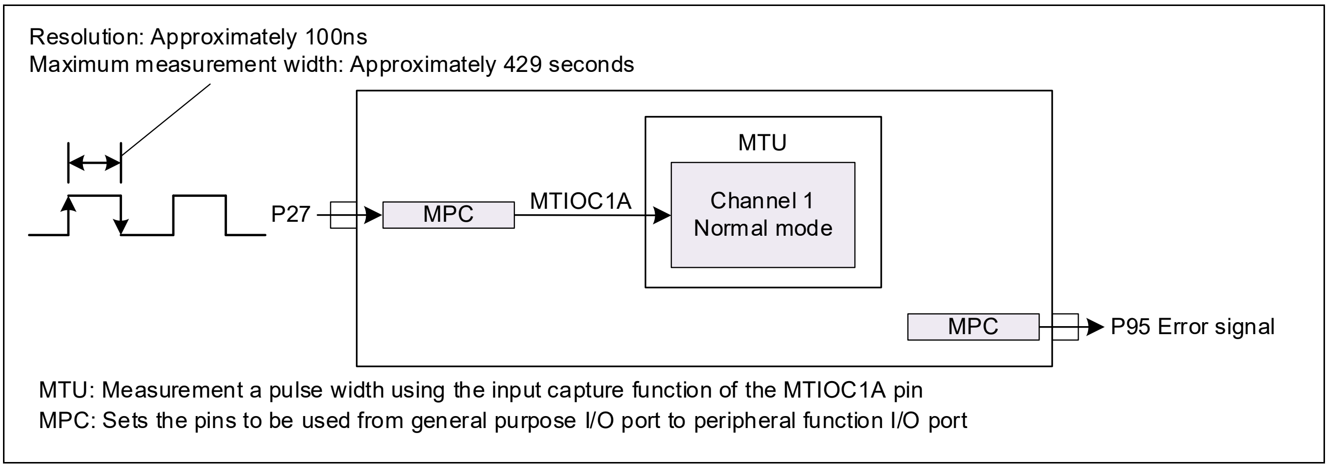


Figure 3.5 Sample Code Structure

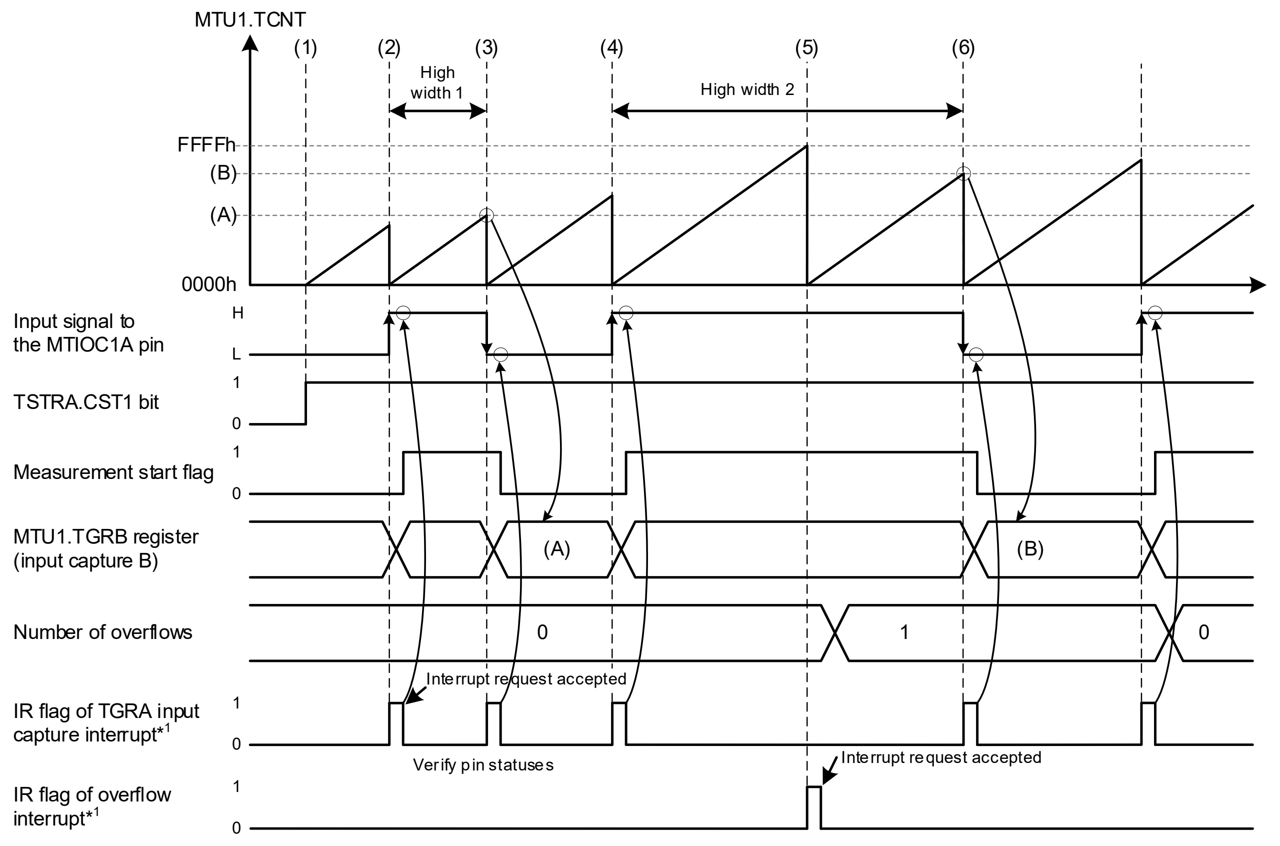
3.2.2 Operation Details

This section describes the operation of this sample code.

(1) When the TSTRA.CST1 bit is set to 1 (start counting), MTU1 starts counting.
(2) When an edge is input to the MTIOC1A pin, the value of the MTU1.TCNT register is transferred to the MTU1.TGRA register and the counter is cleared. At the same time, an MTU1 input capture A interrupt request is generated. The input capture A interrupt processing checks the state of the MTIOC1A pin. If the pin is high, the sample code judges that high pulse measurement has started, sets the measurement start flag to 1 (measurement in progress), and clears the overflow count.
(3) When an edge is input to the MTIOC1A pin again, an input capture A interrupt request of MTU1 is generated. The input capture A interrupt processing checks the state of the MTIOC1A pin. If the pin is low, the sample code judges that high pulse measurement has ended, calculates the pulse width (high width 1 in Figure 3.6) based on the number of overflows in the MTU1.TCNT register (0 in (3) in Figure 3.6) and the value in the MTU1.TGRA register ((A) in Figure 3.6). It also clears the measurement start flag.
(4) When a rising edge is input to the MTIOC1A pin again, the same operation as (2) starts.
(5) When the MTU1.TCNT register overflows, an overflow interrupt request is generated. The overflow interrupt processing counts the number of overflows.
(6) When a falling edge is input to the MTIOC1A pin again, the same operation as (3) starts, and the sample code calculates the pulse width (high width 2 in Figure 3.6) based on the number of overflows in the MTU1.TCNT register (1 in (6) in Figure 3.6) and the value in the MTU1.TGRA register ((B) in Figure 3.6). It also clears the measurement start flag.

In this sample code, each pulse width is calculated with the resolution set to approx. 100 ns. To change the timer count clock of MTU1, change the following setting in Config_MTU1_user.c to the resolution value corresponding to the count clock.

```
#define RESOLUTION_VALUE            (100) /* Count clock cycle (ns) */
```



Note: 1. Allocated to software configurable interrupt A

Figure 3.6 Sample Code Operations

3.2.3 Smart Configurator Settings

The sample code uses the Smart Configurator to add the MTU as described below. For details on how to add components, refer to section 3.1.4, Adding Components.

Table 3.4 Adding Components (MTU)

Item	Description
Component	Normal mode timer
Configuration name	Config_MTU1
Input capture/output compare pins	2 pins
Resource	MTU1

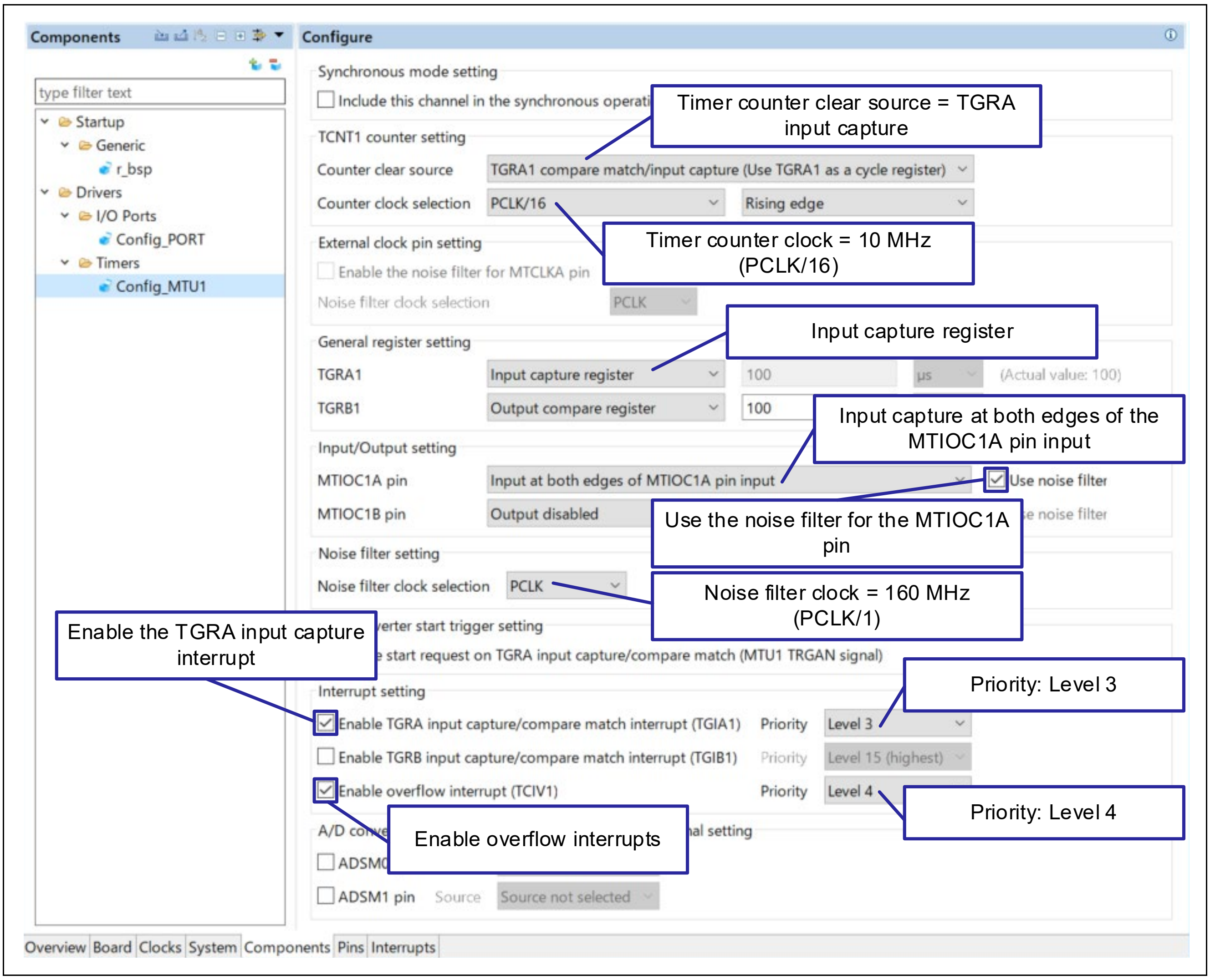


Figure 3.7 MTU1 Settings

To use P95 as a general purpose I/O port, add PORT as follows.

Table 3.5 Adding Components (PORT)

Item	Description
Component	Port
Configuration name	Config_PORT
Resource	PORT

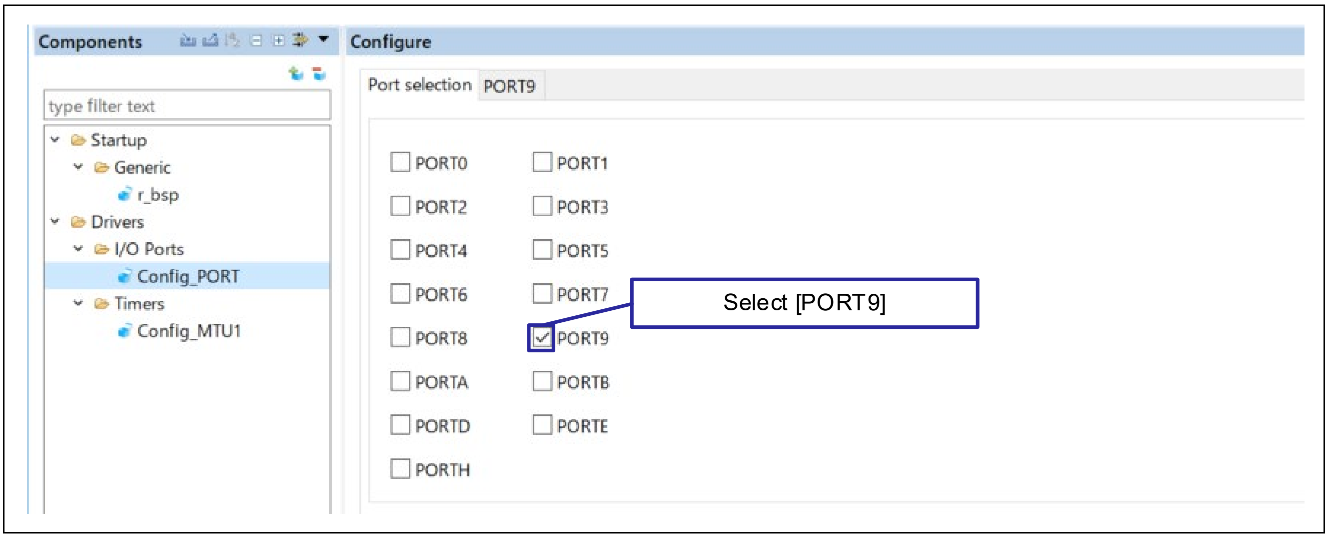


Figure 3.8 P95 Settings (1/2)

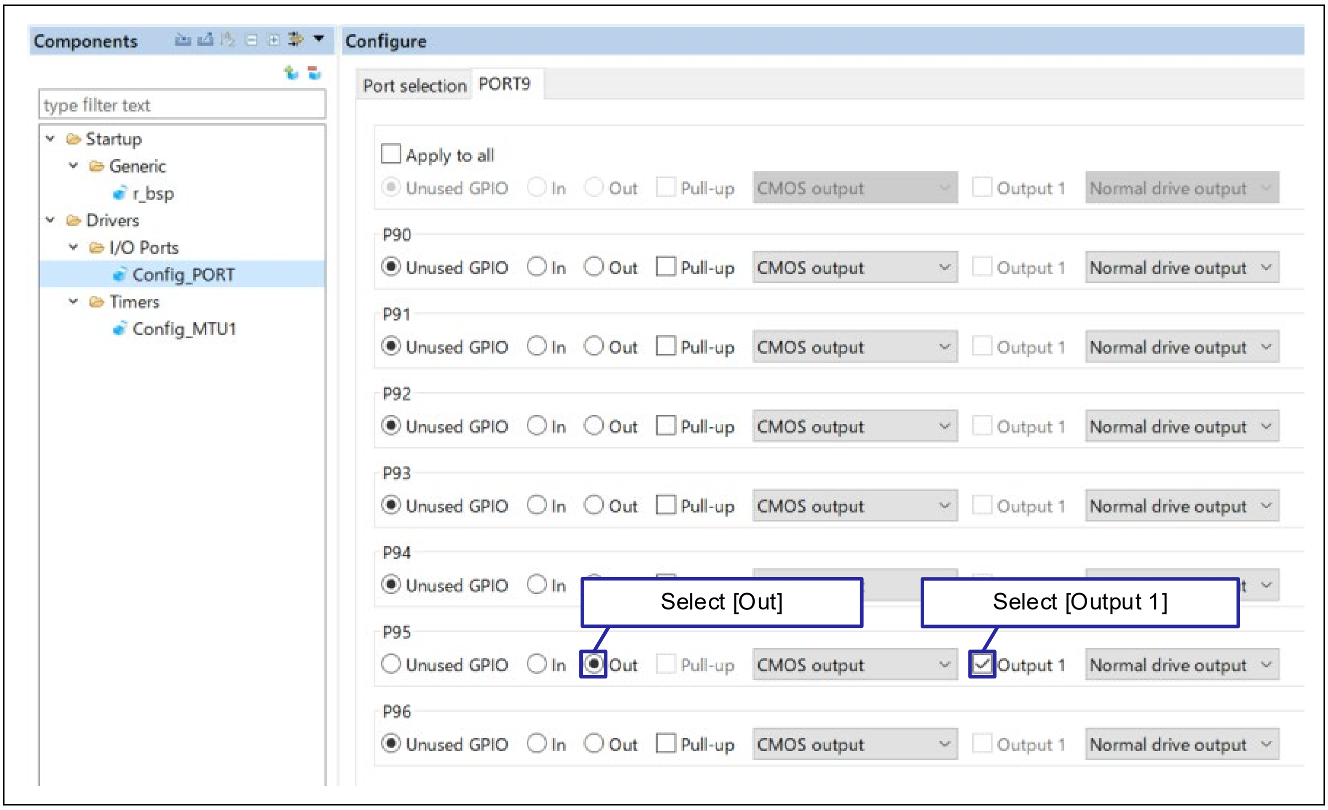


Figure 3.9 P95 Settings (2/2)

3.2.4 Flowcharts

The following flowchart shows the main function processing added after code generation by the Smart Configurator.

Counting starts in the main function.

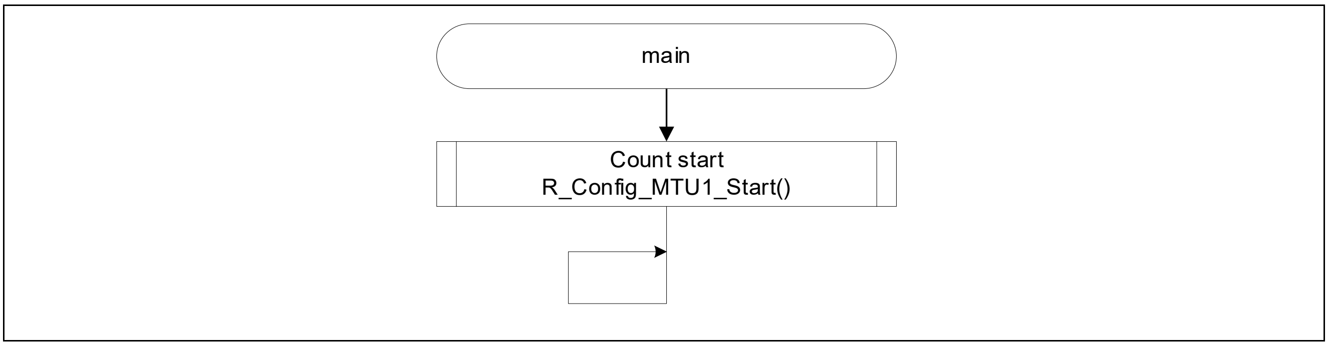


Figure 3.10 main Function

The user initialization function R_Config_MTU1_Create_UserInit, which is executed before the main function, initializes variables. This function is called in the R_Config_MTU1_Create function.

The R_Config_MTU1_Create_UserInit function initializes the following variables that are used by this sample code.

- s_mtu1_ovf_cnt : Overflow counter of the MTU1.TCNT register
- s_pulse_cnt : Pulse width measurement counter
- s_pulse_width*1 : Calculated pulse width value
- s_start_flag : Measurement start flag (0: before measurement, 1: measurement in progress)
- s_error_flag*1 : Measurement error flag (0: Normal, 1: Error)

Note: 1. This sample code generates a build warning because it is not used after setup.

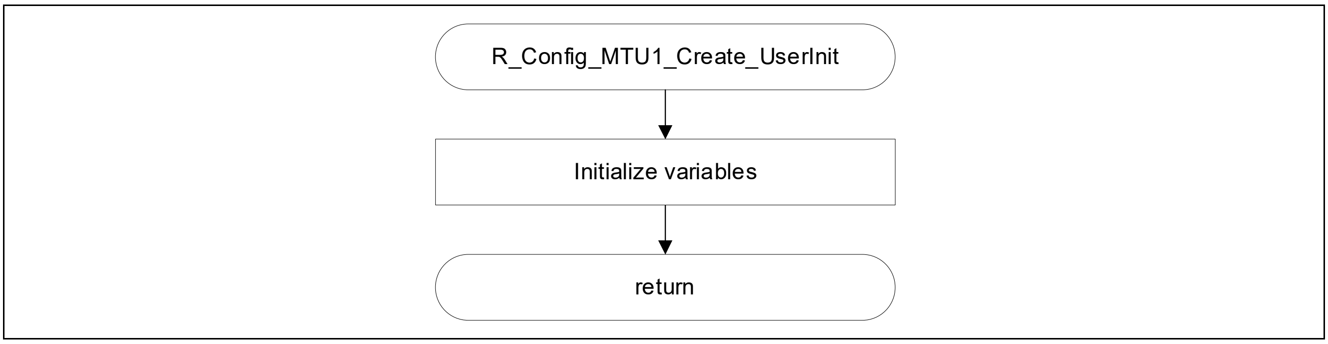
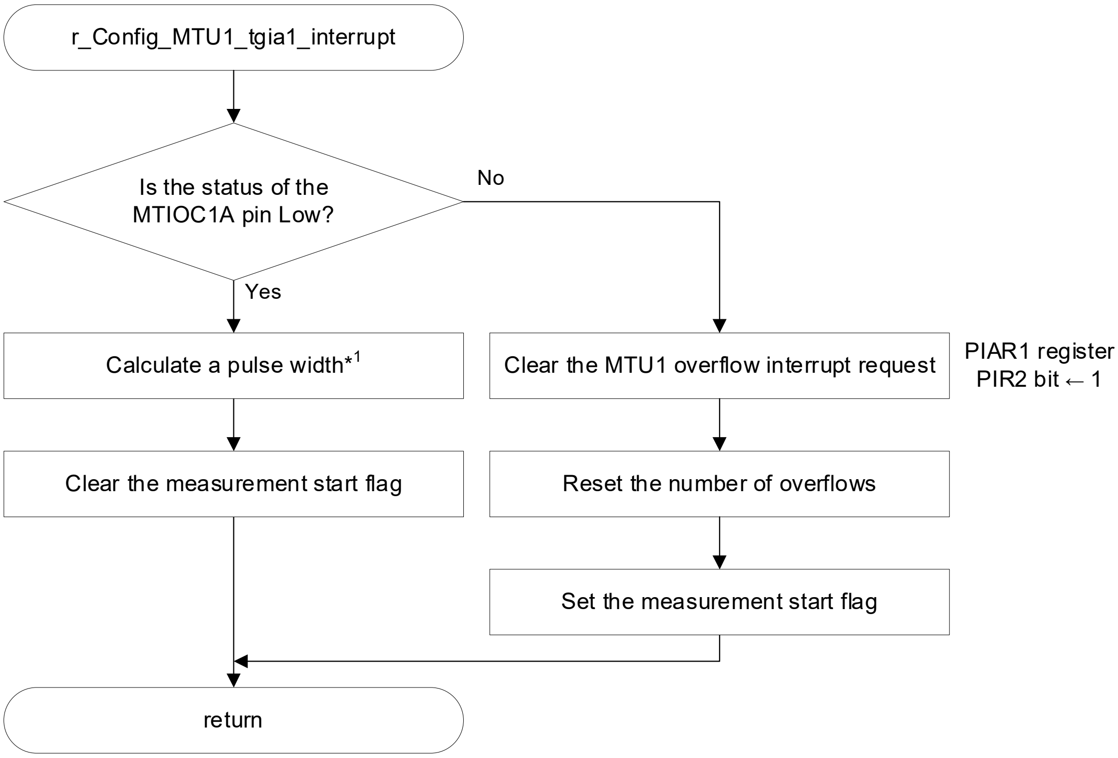


Figure 3.11 User Initialization Function

The TGIA1 interrupt handler function calculates the pulse width when the MTIOC1A pin is low. It also clears the measurement start flag.



Note: 1. Pulse width measurement counter ← Number of overflows<<16 + MTU1.TGRA register value
Calculated pulse width value (ns) ← Resolution * pulse width measurement counter

Figure 3.12 TGIA1 Interrupt Handler Function

The TCIV1 interrupt handler function counts the number of overflows when the measurement start flag is 1 (measurement in progress). If the number of overflows exceeds 65,535, a transition is made to error processing.

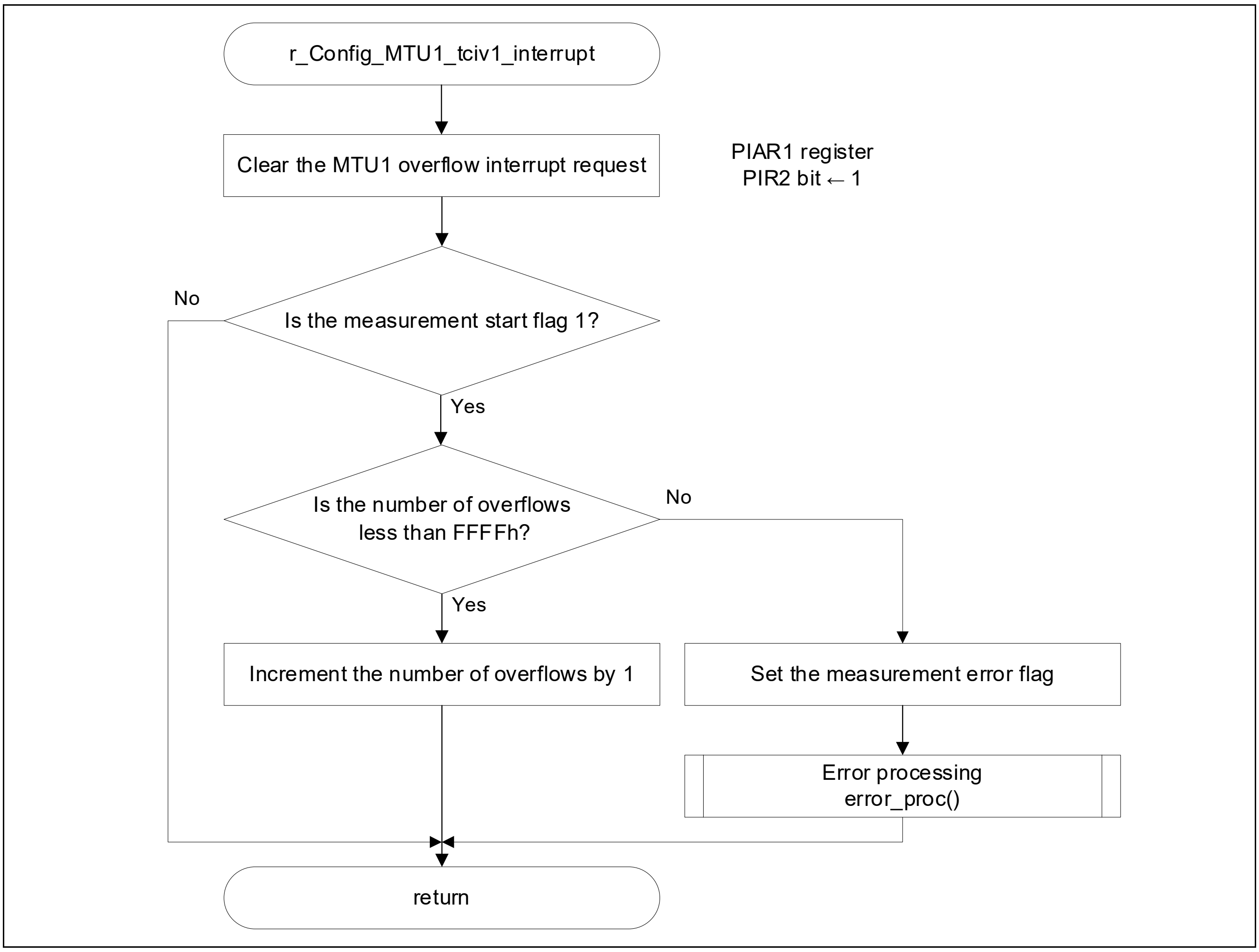


Figure 3.13 TCIV1 Interrupt Handler Function

The error processing function outputs an error signal (lights LED0) and transitions to an infinite loop.

This function is newly created after code generation by the Smart Configurator.

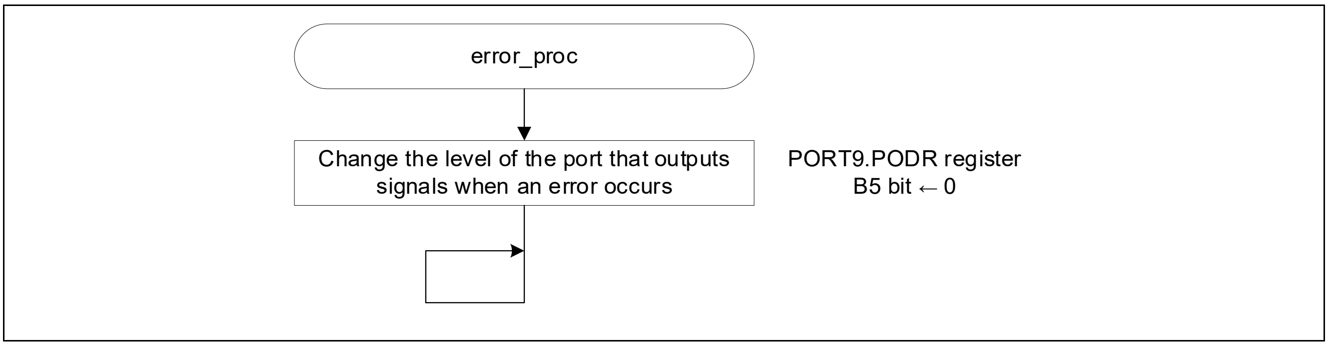


Figure 3.14 Error Processing Function

3.2.5 Related Operation

3.2.5.1 Operation When Input Capture and Overflow Occur Simultaneously

This section describes the operation when an input capture and an overflow occur simultaneously.

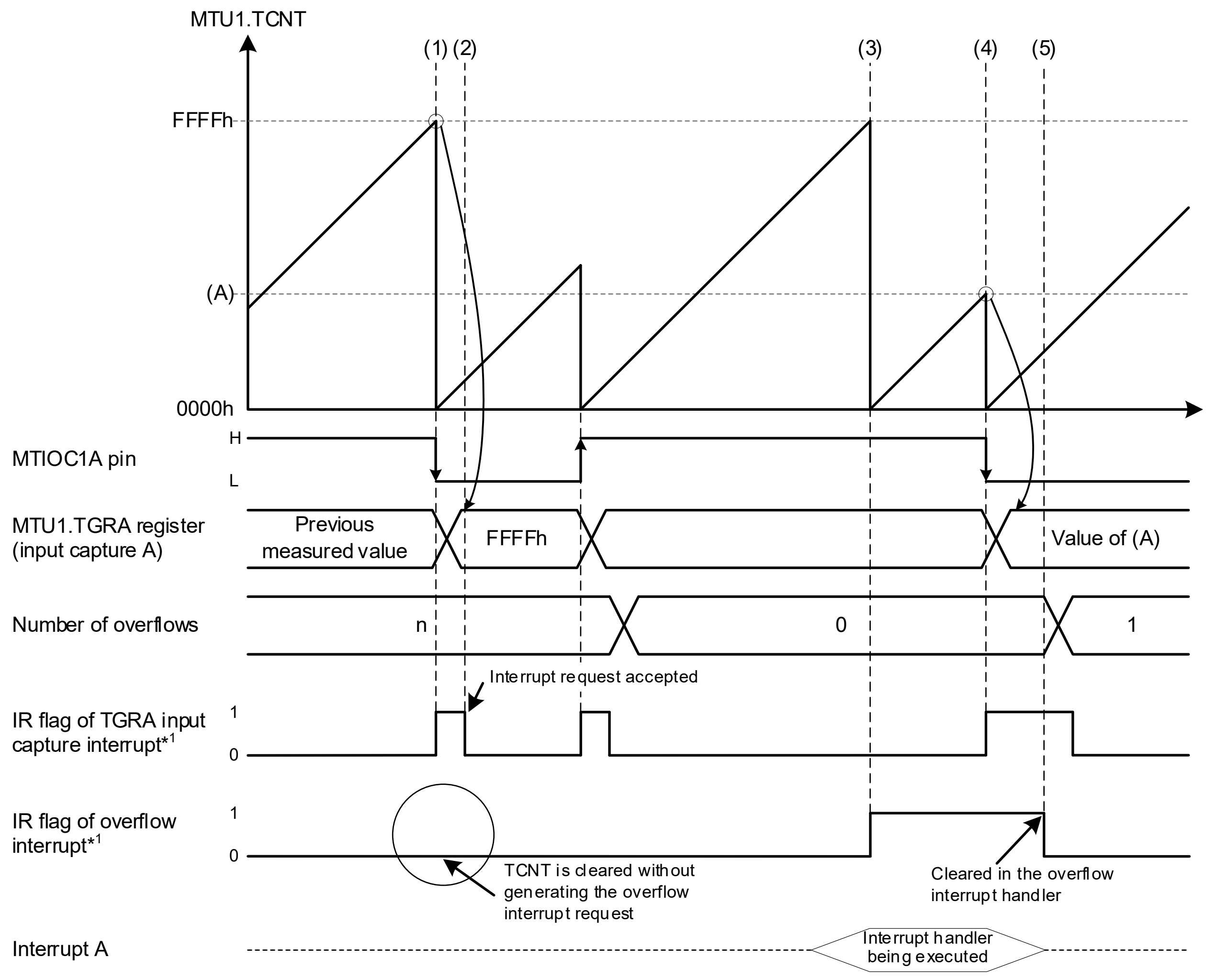
(1) If a falling edge is input to the MTIOC1A pin while the value of the MTU1.TCNT register is FFFFh, the value FFFFh of the MTU1.TCNT register is transferred to the MTU1.TGRA register, and then the MTU1.TCNT register is cleared and an input capture A interrupt request is generated. If an overflow and counter clear occur at the same time, the counter clear takes priority and no overflow interrupt request is generated.

(2) The input capture A interrupt processing reads the value (FFFFh) of the MTU1.TGRA register and calculates the pulse width.

(3) When the value of the MTU1.TCNT register overflows while executing interrupt processing (hereafter referred to as interrupt process A) other than an overflow interrupt or input capture A interrupt, the overflow interrupt processing is kept waiting.

(4) When a falling edge is input to the MTIOC1A pin during execution of interrupt processing A, the value of the MTU1.TCNT register is transferred to the MTU1.TGRA register and an input capture A interrupt request is generated.
(Input capture A interrupt processing is kept waiting.)

(5) When interrupt process A completes, an overflow interrupt with a higher interrupt priority level is executed first. The overflow interrupt processing increments the number of overflows by one. The pulse width is calculated with the input capture A interrupt which is accepted next.



Note: 1. Allocated to software configurable interrupt A

Figure 3.15 Operation When Input Capture and Overflow Occur Simultaneously

3.2.6 Usage Notes

3.2.6.1 Notes When Incorporating into a System

When using the sample code in this application note in an actual system, the following phenomena might occur.

- If the interrupt used in this application note is kept waiting for a long time because of the processing of other interrupts, etc., it may not operate properly.

- If the measurement pulse period does not meet the minimum value of input capture pulse input width specified in the electrical characteristics of the RX66T, it cannot be measured correctly.
- For details, refer to section 45.4.6.3, MTU, in the RX66T Group User's Manual: Hardware.

- Even when the input capture pulse input width specified in the electrical characteristics is met, if the measurement pulse period is short, the software may not process the measurement correctly in time.

4. How to Import the Project

The sample code is provided in the format of an e² studio project. This chapter describes how to import a project into e² studio and CS+. After the import is complete, confirm the build and debugger settings.

Also visit the following Renesas Electronics web page:

https://www.renesas.com/software-tool/migration-e2studio-to-csplus

4.1 Importing with e² studio

To use the sample code in e² studio, import it into e² studio using the following steps.

(The actual screen may vary according to the version of e² studio you are using.)

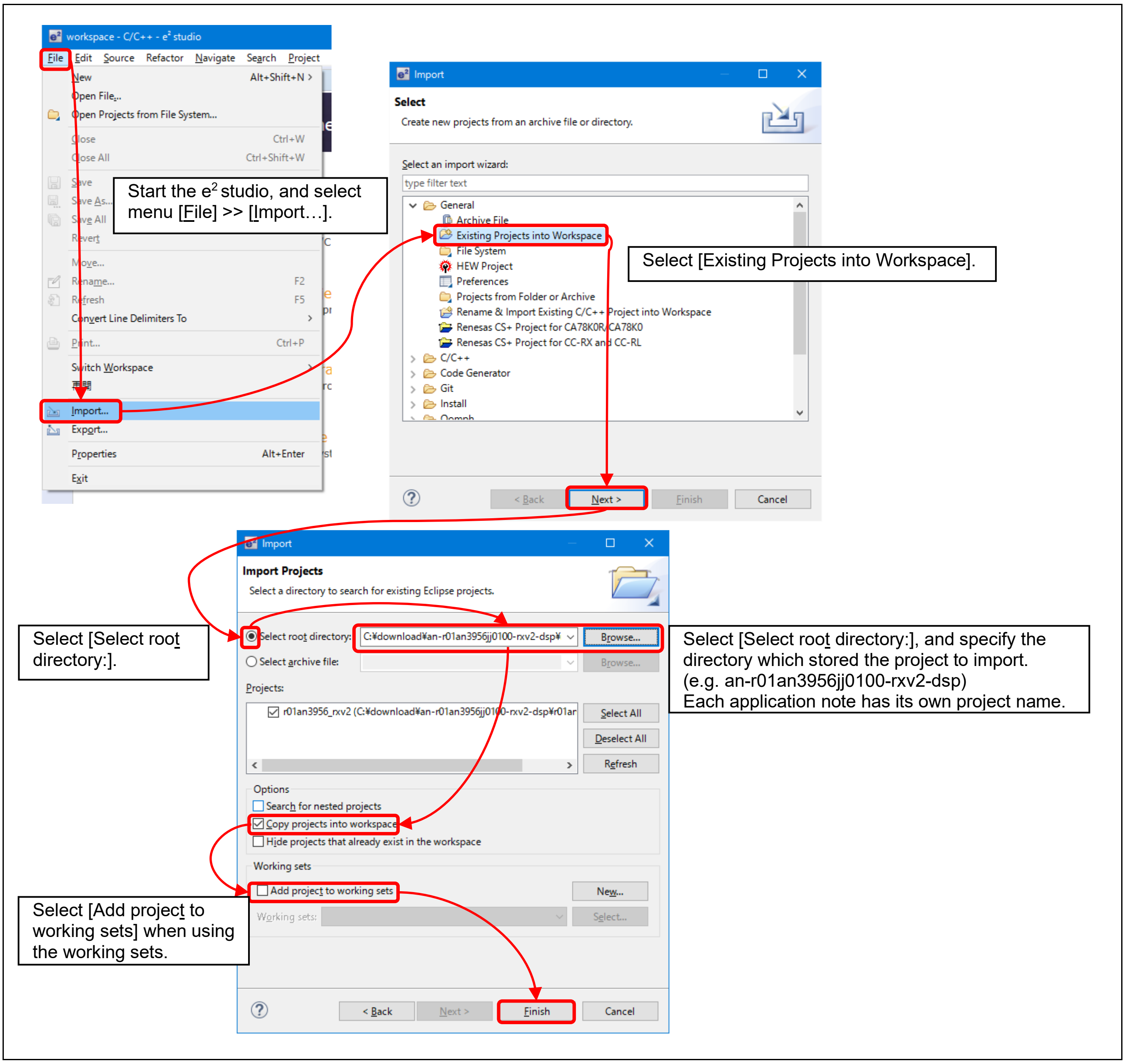


Figure 4.1 How to Import a Project into e² studio

4.2 Importing with CS+

When using the sample code with CS+, import the code to CS+ using the following steps.

(The actual screen may vary according to the version of CS+ you are using.)

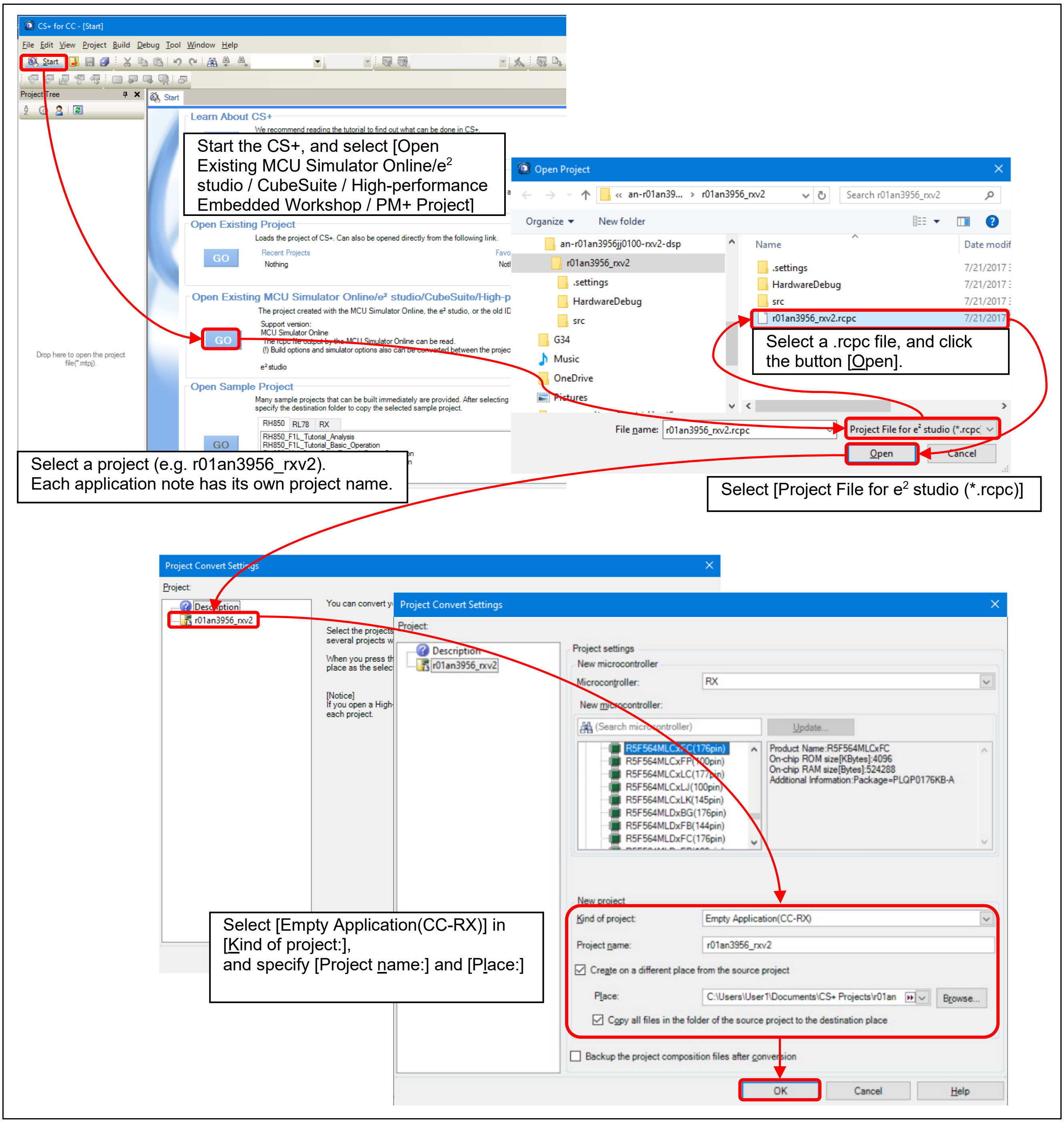


Figure 4.2 How to Import a Project into CS+

5. Reference Documents

- User’s Manual: Hardware
  RX66T Group User’s Manual: Hardware (R01UH0749)
  (Please obtain the latest version from the Renesas Electronics Corp. website.)
- Technical Updates/Technical News
  (Please obtain the latest version from the Renesas Electronics Corp. website.)
- User’s Manual: Development Environment
  RX Family CC-RX Compiler User’s Manual (R20UT3248)
  (Please obtain the latest version from the Renesas Electronics Corp. website.)
- User’s Manual: Development Environment
  RX66T Group Renesas Starter Kit User’s Manual (R20UT4150)
  (Please obtain the latest version from the Renesas Electronics Corp. website.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Feb. 3, 2023	—	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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.

© 2023 Renesas Electronics Corporation. All rights reserved.