

## RL78/G16 Group

### Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

---

#### Overview

This application note describes development and tool ecosystem, core architecture, peripheral considerations, and software development kit for when migrating from Microchip Technology® PIC16(L)F1 Family to RL78/G16 Group.

The intent of this document is to highlight the differences between the two products and to leverage existing knowledge of the Microchip PIC16(L)F1 Family development environment to quickly transition to the RL78/G16 Group of MCUs.

#### Device to Be Migrated

Microchip Technology PIC16(L)F1 Family

#### Target Device

RL78/G16 Group

#### Related documents

- RL78/G16 User's Manual: Hardware (R01UH0980)
- RL78 Family User's Manual: Software (R01US0015)
- Integrated Development Environment e<sup>2</sup>studio User's Manual Quick Start Guide Renesas Micro Controller RX/RL78/RH850/RISC-V MCU Family (R20UT5293)
- RL78/G16 Guide for Engineer (EP2P-AA-24-0323)

Please visit [RL78/G16 product website](#) to download related application notes.

**Contents**

1.	RL78/G16 Portfolio Overview .....	4
1.1	Introduction .....	4
1.2	Portfolio Comparison of PIC(16(L)F1 and RL78/G16 .....	4
2.	Ecosystem and Migration.....	6
2.1	Comparison of Ecosystem .....	6
2.1.1	Smart Browser.....	7
2.1.2	IDE Supported by RL78/G16.....	8
2.1.3	Smart Configurator .....	9
2.1.4	Debug Tools .....	12
2.1.5	Evaluation Board .....	13
2.2	Migration Process.....	14
2.2.1	Step 1: Selecting the correct RL78/G16.....	14
2.2.2	Step 2: IDE Setup and Quick Introduction of e <sup>2</sup> studio .....	17
2.2.2.1	IDE Setup .....	17
2.2.2.2	How to Use e <sup>2</sup> studio .....	19
2.2.3	Step 3: Software importing example .....	24
2.2.4	Step 4: Evaluating the Software .....	28
2.2.5	Step 5: Mass Production .....	28
2.3	Sample program Migration Example .....	30
3.	Core Architecture Comparison.....	36
3.1	CPU .....	36
3.2	Built-in Memory Comparison .....	37
3.2.1	Flash Memory Function .....	37
3.2.2	Flash Memory Configuration .....	38
3.2.2.1	Flash Memory Area .....	38
3.2.2.2	Flash Memory Control Register in RL78/G16 .....	38
3.2.2.3	Self-Programming in RL78/G16 .....	38
3.2.3	On-chip RAM .....	39
3.3	Power-On and Reset Overview and Comparison .....	40
3.4	Clock Overview and Comparison .....	42
3.4.1	Oscillator.....	42
3.4.1.1	RL78/G16 Oscillator .....	42
3.4.2	Clock Signal Comparison .....	43
3.5	Overview and Comparison of Operation Modes .....	44
3.5.1	Comparison of Operation Modes .....	44
3.5.2	RL78/G16 Functions in HALT Mode .....	45
3.5.3	RL78/G16 Functions in STOP Mode.....	45

3.6	Comparison of Interrupts and Events.....	46
3.6.1	Interrupts and Exceptions.....	46
3.6.1.1	Interrupt Control in RL78/G16.....	47
3.6.1.2	Interrupt Control in PIC16(L)F183xx.....	47
3.7	Debugging and Programming Comparison.....	48
3.7.1	Debugging Comparison.....	48
3.7.1.1	Debugging RL78/G16.....	48
3.7.1.2	Debugging PIC16(L)F183xx.....	48
3.7.2	Programming Mode Comparison.....	49
3.7.2.1	Serial Programming in RL78/G16.....	49
3.7.2.2	Serial Programming in PIC16(L)F183xx.....	49
4.	Digital Peripheral Comparison.....	50
4.1	General Purpose I/O (PORT, PIOR).....	50
4.2	Universal Asynchronous Receiver-Transmitter (UART, SAU).....	51
4.3	Serial Peripheral Interface (SPI, SAU).....	52
4.4	Inter-Integrated Circuit (IICA, SAU).....	53
4.5	Timer (TAU, 12-Bit Interval Timer).....	54
4.6	Watchdog Timer (WDT).....	55
5.	Analog Peripheral Comparison.....	56
5.1	Analog Digital Converter (ADC).....	56
5.2	Comparator (COMP).....	57
5.3	Internal Reference Voltage ( $V_{REFP}$ ).....	57
	Reference.....	57
	Revision History.....	58

## 1. RL78/G16 Portfolio Overview

### 1.1 Introduction

The RL78/G16 Group (hereafter "RL78/G16") features 16-bit high performance microcontrollers with low power consumption. With 16-bit architecture and highly precise built-in circuits such as high-speed on-chip oscillator and power-on reset, these MCUs help reduce the need for external peripheral components. The RL78/G16 design focuses on low power consumption that enables extremely low operating current and standby power consumption, making it highly suitable for battery-operated devices and applications requiring low power consumption.

The RL78 Family has been used for many years in a wide range of fields, from home appliances and consumer devices to industrial equipment, and is recognized as a highly reliable microcontroller. A generous assortment of sample codes and application notes are provided with the Renesas ecosystem to enable rapid product development.

The Renesas RL78/G16 is a competitive alternative to the Microchip Technology PIC16(L)F1 Family (hereafter referred to as "PIC16(L)F1"). Because PIC16(L)F1 require 4 cycles per 1 order, clock efficiency of RL78/G16 is superior to PIC16(L)F1 in actual use-cases and instruction execution time of RL78/G16 is much shorter than PIC16(L)F1.

This application note assists you with migration from PIC16(L)F1 to RL78/G16 comparing the device features and ecosystem.

### 1.2 Portfolio Comparison of PIC(16(L)F1 and RL78/G16

**Table 1-1** Comparison of Renesas RL78/G16 Group and Microchip PIC16(L)F1 Family (1/2)

Item	Microchip <sup>[1]</sup> PIC16F180xx	Microchip <sup>[2]</sup> PIC16(L)F183xx	Microchip <sup>[3]</sup> PIC16(L)F184xx	Renesas RL78/G15	Renesas RL78/G16
Core	Enhanced Mid-Range CPU			RL78-S2 Core	
Clock frequency	Max 32 MHz Note1	Max 32 MHz Note1	Max 32 MHz Note1	Max 16 MHz	Max 16 MHz
Instruction execution time (Min)	0.125us	0.125us	0.125us	0.0625us	0.0625us
Power supply voltage	2.5 to 5.5 V	2.5 to 5.5 V	2.5 to 5.5 V	2.4 to 5.5 V	2.4 to 5.5 V
Operating temperature range	-40 to +85°C -40 to +125°C	-40 to +85°C -40 to +125°C	-40 to +85°C -40 to +125°C	-40 to +85°C, -40 to +105°C -40 to +125°C	-40 to +85°C, -40 to +105°C -40 to +125°C
Code flash	3.5 to 28KB	3.5 to 28KB	7 to 28KB	4 to 8KB	16 to 32KB
Data flash / EEPROM	128 to 256B	256B	256B	1KB	1KB
RAM	0.25 to 2KB	0.25 to 2KB	0.5 to 2KB	1KB	2KB

**Table 1-2** Comparison of Renesas RL78/G16 Group and Microchip PIC16(L)F1 Family (2/2)

Item		Microchip <sup>[1]</sup> PIC16F180xx	Microchip <sup>[2]</sup> PIC16(L)F183xx	Microchip <sup>[3]</sup> PIC16(L)F184xx	Renesas RL78/G15	Renesas RL78/G16
Interrupt source	Internal	Max 30	Max 33	Max 45	Max 19	Max 30
	External	1 + IOC Max 25	1 + IOC Max 18	1 + IOC Max 25	Max 8	Max 10
I/O port		Max 35	Max 17	Max 25	Max 17	Max 27
Analog	ADC	Max 35-ch (10-bit resolution)	Max 17-ch (10-bit resolution)	Max 24-ch (12-bit resolution)	Max 11-ch (10-bit resolution)	Max 11-ch (10-bit resolution)
	DAC	1-ch (8-bit resolution)	1-ch (5-bit resolution)	1-ch (5-bit resolution)	N/A	N/A
	COMP	1-ch	Max 2-ch	Max 2-ch	Max 2-ch	Max 2-ch
Serial interface (Max)	UART	Max 2-ch	1-ch	Max 2-ch	1-ch	Max 3-ch
	I <sup>2</sup> C	Max 2-ch	Max 2-ch	Max 2-ch	Max 3-ch	Max 4-ch
	SPI	Max 2-ch	Max 2-ch	Max 2-ch	Max 2-ch	Max 3-ch
Timer		8-bit: 4-ch 16-bit: 3-ch	8-bit: Max 4-ch 16-bit: Max 4-ch	8-bit: 4-ch 16-bit: 4-ch	8-bit: 4-ch 16-bit: 8-ch	8-bit: 4-ch 16-bit: 8-ch
Other key peripherals / Features		ZCD Temp. Indicator HCVD <b>PPS</b>	Temp. Indicator PPS PMD	ZCD Temp. Indicator HCVD PPS PMD	PIORn PERn	Temperature sensor Touch sensor RTC2 PIORn PERn
Number of pins		8 to 44 pins	8 to 20 pins	14 to 28 pins	8 to 20 pins	10 to 32 pins
Low power Note2	Normal operation	4.5mA@32MHz, 3V, typical	2.2mA@32MHz, 3V, typical	3.9mA@32MHz, 3V, typical	1.86mA@16MHz, 3V, typical	2.11mA@16MHz, 3V, typical
	Stand-by	N/A	DOZE mode 863uA@16MHz, 3V, typical	DOZE mode 1.5mA@16MHz, 3V, typical	HALT mode 371uA@16MHz, 3V, typical	HALT mode 385uA@16MHz, 3V, typical
			IDLE mode 804uA@16MHz, 3V, typical	IDLE mode 1.5mA@16MHz, 3V, typical		
SLEEP mode 900nA@3V, typical	SLEEP mode 950nA@3V, typical	SLEEP mode 880nA@3V, typical	STOP mode 620nA@3V, typical	STOP mode 620nA@3V, typical		

Note1: PIC16(L)F1 requires 4 cycles per one order.

Note2: The STOP mode in RL78/G16 corresponds to the PIC16(L)F1 SLEEP mode. HALT mode corresponds to the PIC16(L)F1 DOZE mode or IDLE mode.

## 2. Ecosystem and Migration

The RL78/G16 ecosystem provides a wide range of support from hardware and software to debugging tools, libraries, and a support community to assist flexible and efficient development for a variety of applications. Developers can use this ecosystem to advance their product development quickly and effectively.

### 2.1 Comparison of Ecosystem

Table 2-1 Comparison of Ecosystem

Features	PIC16(L)F1	RL78/G16
Code source	Driver Libraries Middleware RTOS Code example	SIS (Software Integration System) Smart Browser (sample code, document references) Middleware Renesas Flash driver / Self programming library Driver RTOS
IDE	MPLAB® X IDE MPLAB® Xpress Cloud-Based IDE	e²studio CS+ IAR Embedded Workbench
Software configuration	MPLAB® Code Configurator (MCC)	Smart Configurator
Software programming tool	MPLAB® X IDE MPLAB® Xpress Cloud-Based IDE	Renesas Flash Programmer
Hardware programming tool	MPLAB® PICKit™ 5 MPLAB® ICD 5 MPLAB® ICE 4	PG-FP6
Debugger	MPLAB® PICKit™ 5 MPLAB® ICD 5 MPLAB® ICE 4	E2 Emulator E2 Emulator Lite
Evaluation board	MPLAB Xpress Evaluation Board PIC16F18446 Curiosity Nano PIC16F18076 Curiosity Nano Evaluation Kit	RL78/G16 FPB RL78/G16 Capacitive Touch Evaluation System

Figure 2-1 shows the overview of the RL78/G16 Ecosystem.

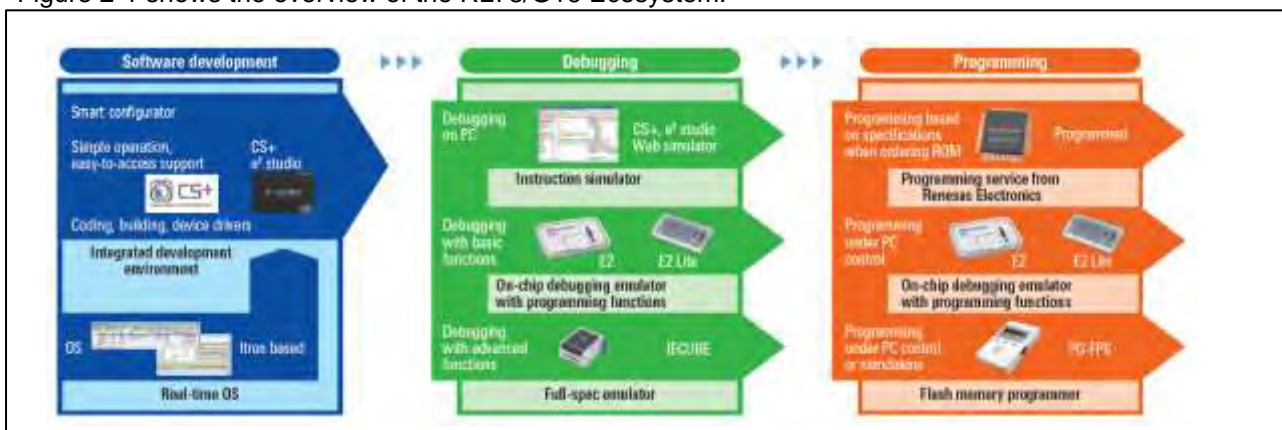


Figure 2-1 Overview of RL78/G16 Ecosystem

2.1.1 Smart Browser

The Smart Browser enables the user to easily search and view the latest hardware manual, technical updates, and application notes for Renesas devices. It also allows the user to download sample codes and import sample codes that include the project.

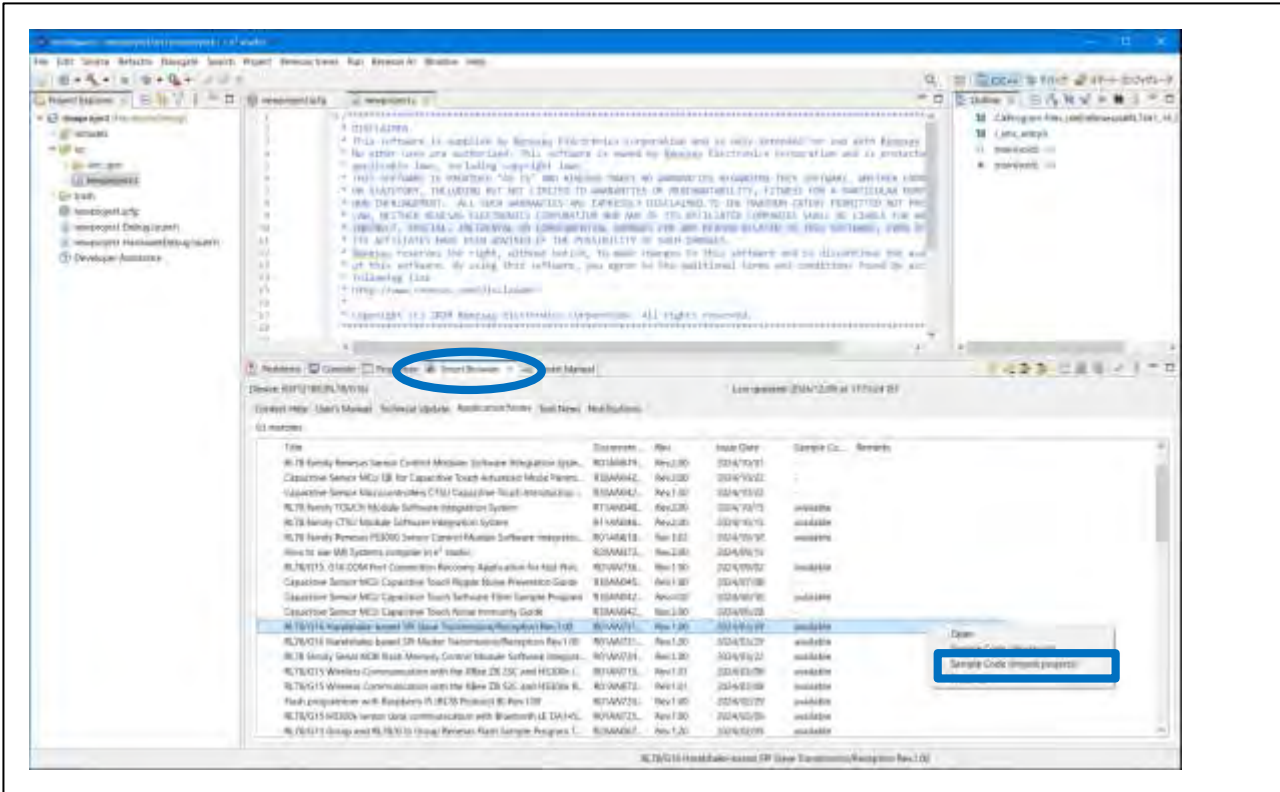


Figure 2-2 Smart Browser

The Smart Browser is included in the e2studio and can be used by installing the e2studio. The Smart Browser has multiple tabs so that developers can quickly access the information necessary for their project.

The User’s Manual tab provides access to various development tools and official documents regarding devices.

The Technical Update tab provides information related to technical updates, new functions, and device updates, allowing developers to check up on the latest technical updates regarding the device and tools used for the project.

The Application Notes tab provides technical information useful to application development, including documents that describe how to use specific peripheral functions and software designs.

Table 2-2 Comparison of Software Ecosystem

Item	PIC16(L)F1	RL78/G16
Driver library	Available	Available
Middleware	Available	Available
Self-programming	Available	Available
Sample program	Available	Available
Free RTOS	Available	Available

The RL78/G16 sample programs are designed to allow easy initialization and configuration of common hardware functions. For example, it includes all necessary code and initialization procedures for major

**Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)**

peripheral functions such as PORT (I/O ports), timers, UART, I<sup>2</sup>C, and SPI. Most of the RL78/G16 sample programs support the Smart Configurator for accelerating software development.

**2.1.2 IDE Supported by RL78/G16**

An Integrated Development Environment (IDE) is a software application that normally includes an editor, compiler, and debugger to help users develop software code efficiently.

The typical IDE supported by PIC16(L)F1 is MPLAB® X IDE which consists of a set of tools used to develop and debug embedded applications, including optimizing C/C++ compiler, source code editor, project build environment, debugger, and many other functions.

The RL78/G16 IDE is e<sup>2</sup>studio, an Eclipse-based IDE like MPLAB® X IDE that allows users to get started easily. e<sup>2</sup>studio also consists of a complete set of tools with many features for developing and debugging embedded applications. In addition, the e<sup>2</sup>studio is free to use and integrates automatic code generation from the Smart Configurator, integration, and management of software modules with the SIS, and provision of RL78/G16 sample code and technical documentation through the Smart Browser.

**Table 2-3** Comparison of MPLAB® X IDE and e<sup>2</sup>studio

IDE	MPLAB® X IDE	e <sup>2</sup> studio
License	Free	Free
Compiler	MPLAB® XC Compilers/ GCC	CC-RL/ LLVM
IDE built-in low power consumption solution	Not supported	QE for Current Consumption
Peripheral API function support	Not supported	Supported
Executable file output format	Hex file Binary file Executable and Linkable Format Common Object File Format	Hex file Binary file Motorola S-record file
Code generation GUI	MPLAB® Code Configurator	Smart Configurator

In addition to e<sup>2</sup>studio, RL78/G16 also supports CS+, IAR Embedded Workbench, Arduino IDE, Visual Studio Code.

For e<sup>2</sup>studio usage and functions, refer to **2.2.2.2** How to Use e2studio.

### 2.1.3 Smart Configurator

The Smart Configurator is integrated into e²studio. Like MPLAB® Code Configurator, the Smart Configurator can quickly and smartly achieve the suitable software combination and configurations according to the user's development needs, such as pin settings for MCU peripherals, and embedding middleware and drivers. With the Smart Configurator, the user can visually resolve conflicts between pin settings or peripheral functions, while customizing settings suitable for the project itself.

The Smart Configurator is also available as a standalone version that can be run without an IDE. The standalone version can be used in combination with the RL78/G16 IDE, such as CS+ or IAR Embedded Workbench.

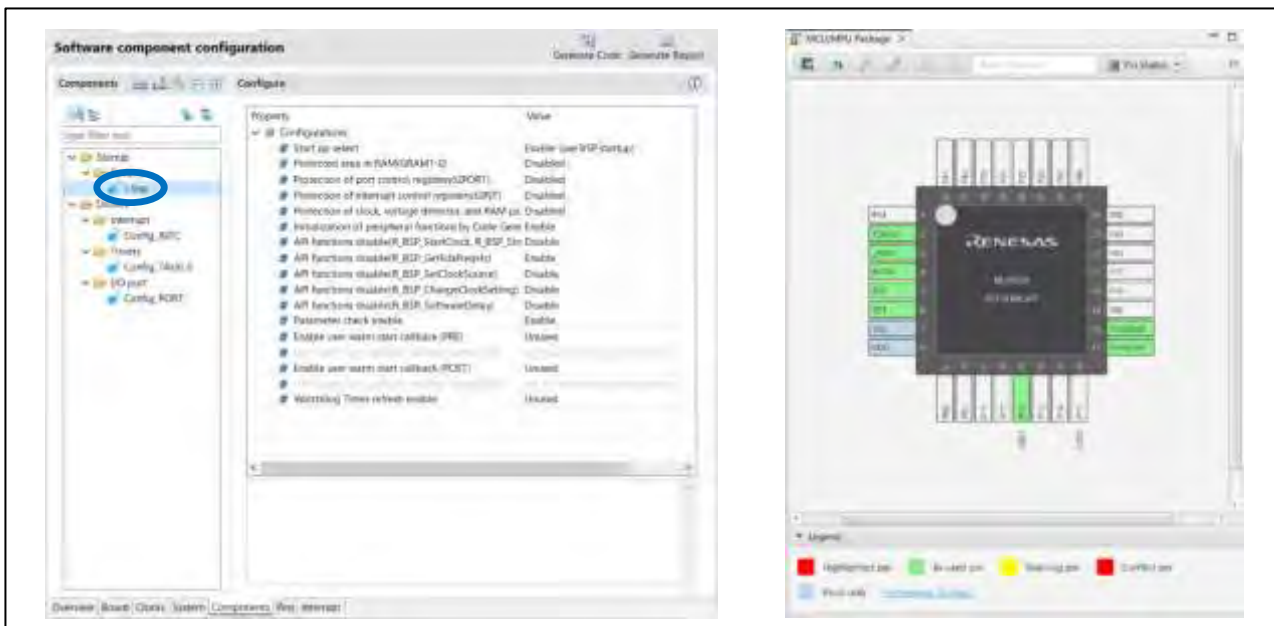


Figure 2-3 Smart Configurator

The Renesas Board Support Package SIS module (r\_bsp) is the basis of projects that use the Software Integration System (SIS) module. The r\_bsp is easy to configure and provides all the code that the MCU and the user board require, from reset to the main() function.

The Smart Configurator, like MPLAB® Code Configurator, provides the hardware circuit diagram display, detailed descriptions of each peripheral function, and detailed functional descriptions of all GUI interface configurations.

Figure 2-4 shows categorized lists of peripheral functions. Figure 2-5 shows a comparison of interrupt settings.

In the Smart Configurator, the interruption settings and pin settings are configured individually for each component, in the same way as MPLAB® Code Configurator. With the Smart Configurator, you can focus on configuring only the components used.

After configuration is complete, pressing the "Generate Code" button triggers the Smart Configurator to update the codes to be generated all at once.

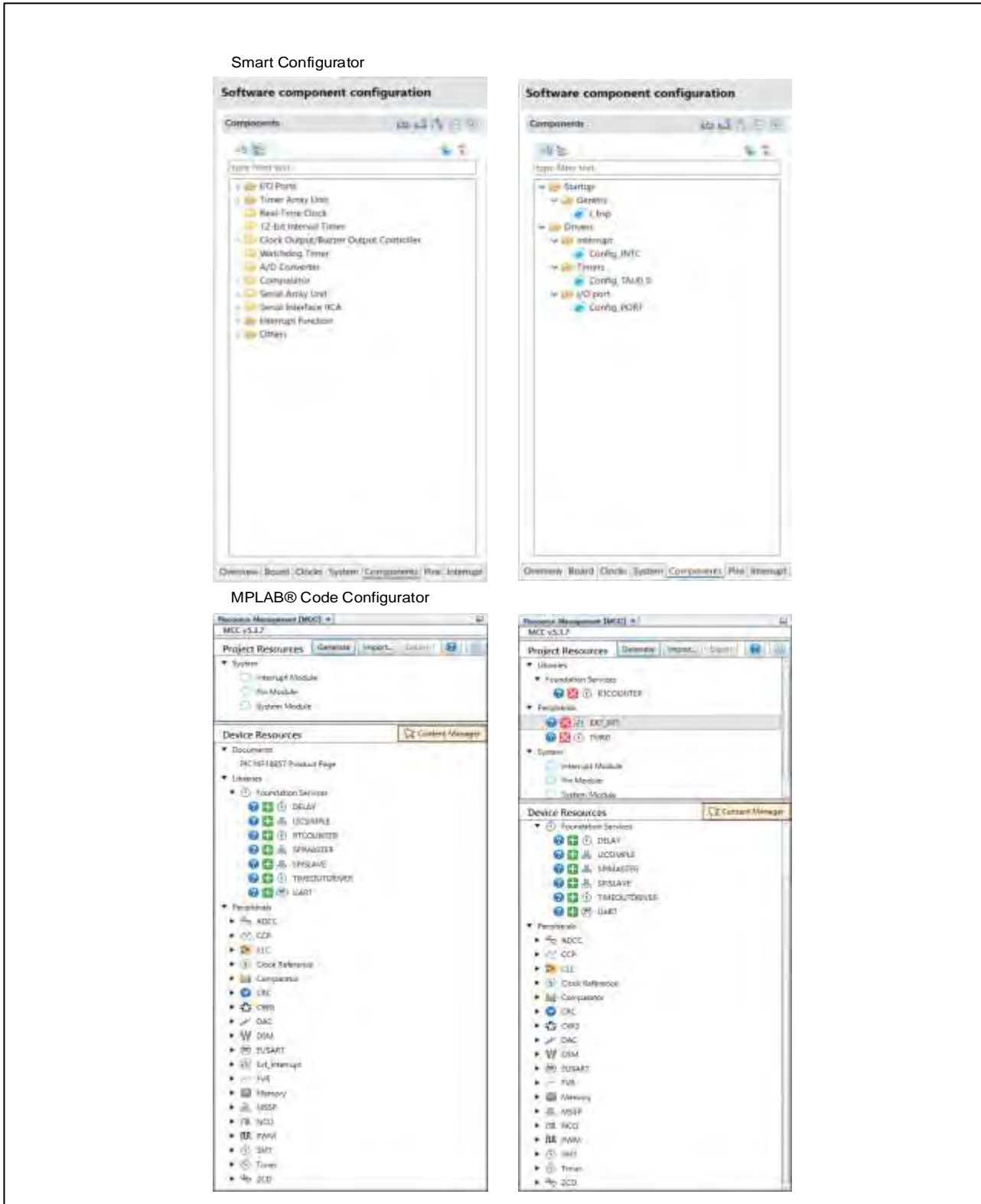


Figure 2-4 Comparison of the Peripheral Functions

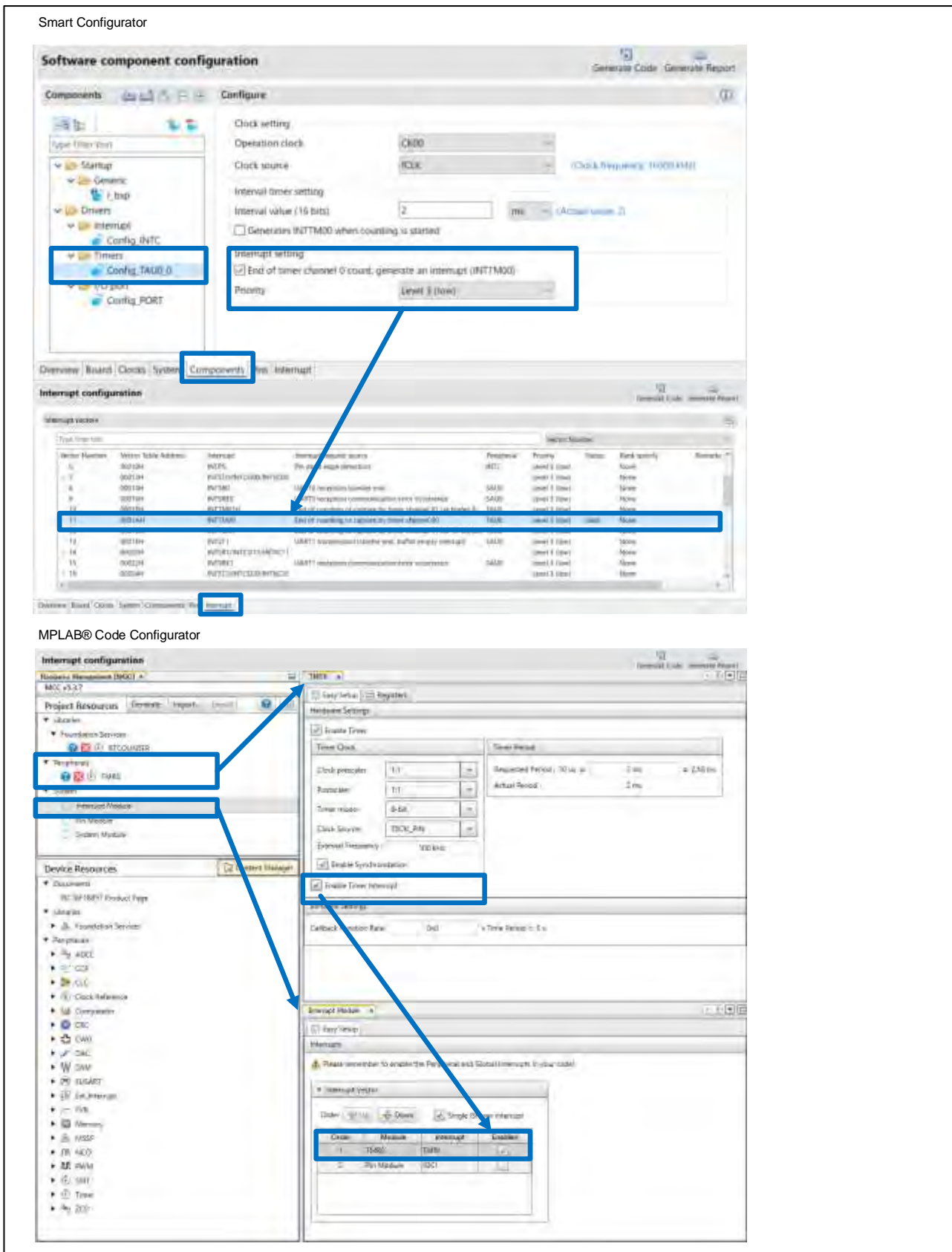


Figure 2-5 Comparison of Interrupt Settings

### 2.1.4 Debug Tools

For PIC16(L)F1, the MPLAB® PICkit™ 5 tool connects to the PGD (Program Data) and PGC (Program Clock) pins using the in-circuit serial programming (ICSP) and in-circuit debugging (ICD) interfaces. It enables real-time monitoring of the device state during program execution and supports debugging in low-power modes.

For RL78/G16, the TOOL0 pin is used as the interface between the debugger (such as E2 emulator and E2 emulator Lite) and the RL78/G16 to manipulate via the dedicated single-line UART. For details on the debugger connection, refer to Figure 2-6.

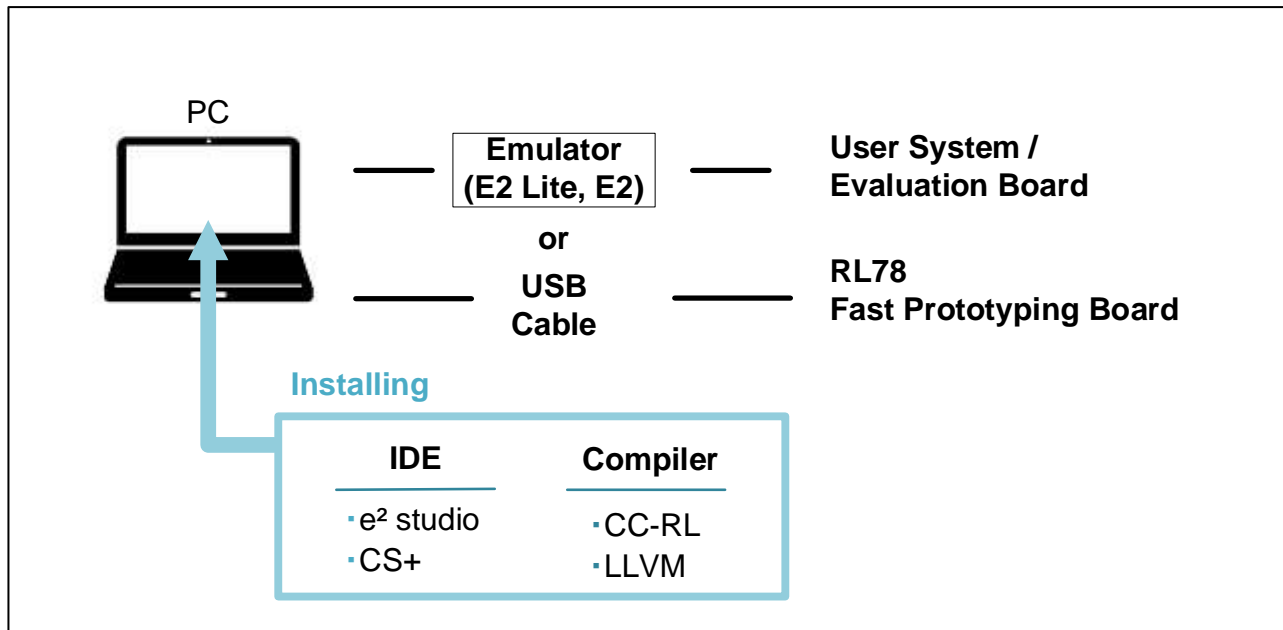


Figure 2-6 RL78/G16 Debug

The RL78/G16 supports on-chip debugging through a COM port connection, the E2 emulator, and the E2 emulator Lite. All these debug functions include a flash memory programming function. The RL78/G16 Fast Prototyping Board is equipped with an on-chip debugging function via a USB connection which enables debugging without an emulator.

The E2 Emulator and E2 Emulator Lite are versatile Renesas emulators that support a wide range of microcontrollers including the RL78/G16. The E2 Emulator can be used for real-time debugging and breakpoint settings to analyze the operation of the program. The advanced trace function allows you to trace the executed code for precise analyzation. Note that the E2 Emulator Lite does not include the advanced trace function. However, it provides a sufficient debug function and enables development at low cost.

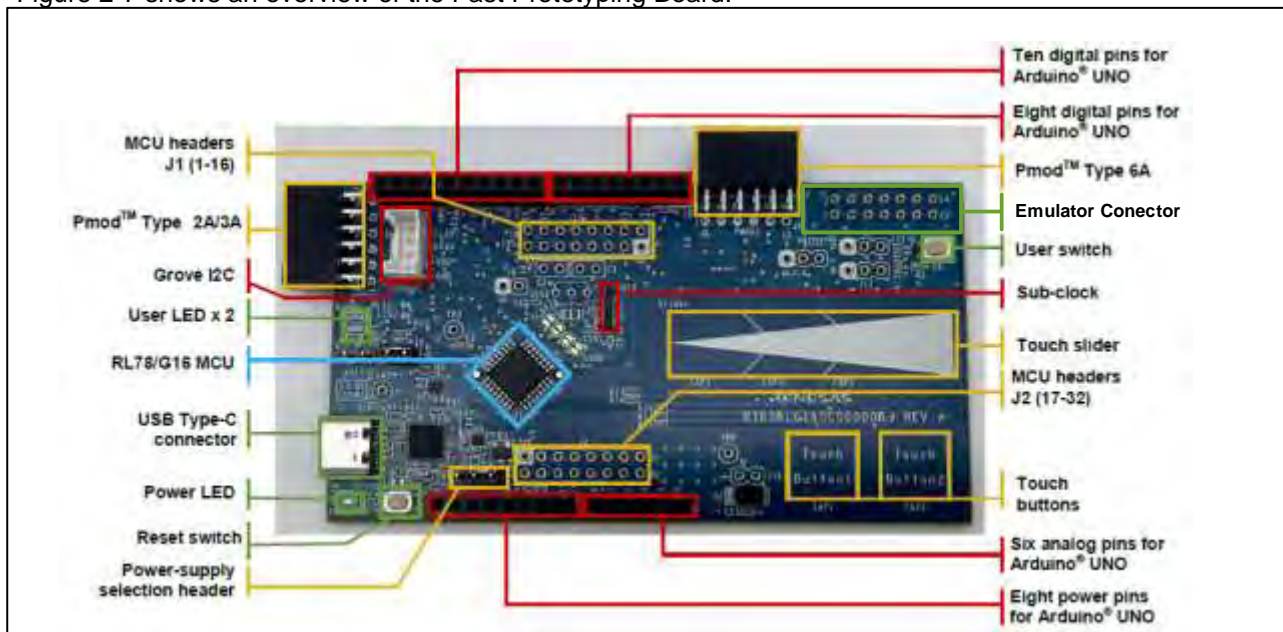
### 2.1.5 Evaluation Board

The Fast Prototyping Board allows you to easily test the operation of the microcontroller, providing strong support for evaluation and development using the RL78/G16 MCU.

The Fast Prototyping Board is an evaluation board equipped with the RL78/G16, made specifically for prototyping and developing various applications.

The RL78/G16 Fast Prototyping Board embeds circuits such as user switches and LEDs. The Arduino header supports various Arduino shields and enables rapid prototyping by creating sketches with the Arduino IDE.

Figure 2-7 shows an overview of the Fast Prototyping Board.



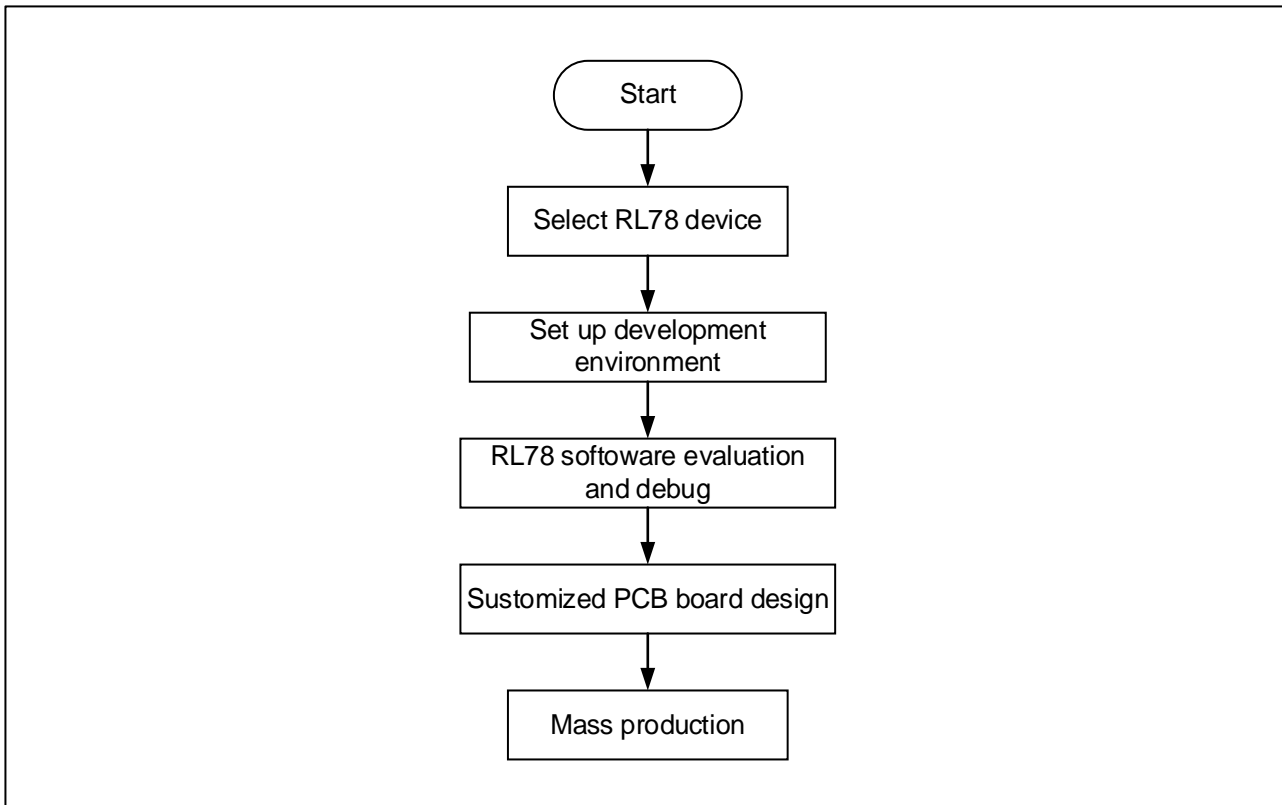
**Figure 2-7** Overview of RL78/G16 Fast Prototyping Board

The FPB enables on-chip debugging with a USB connection. The built-in emulator connector is for Renesas E2 Emulator or E2 Emulator Lite. Connecting to other emulators may require some additional modifications, such as adding a connector. For details, refer to the RL78/G16 Fast Prototyping Board User's Manual.

## 2.2 Migration Process

The process flow for smooth migration to RL78/G16 is shown in Figure 2-8. Each step is described in detail and examples are given in the following sections.

Also refer to RL78/G16 Guide for Engineer (EP2P-AA-24-0323) which introduces the documents regarding each development phase.



**Figure 2-8** Flowchart of Migration to RL78/G16

### 2.2.1 Step 1: Selecting the correct RL78/G16

The first step of migration is to select the correct RL78/G16 for the application. Table 2-4 show the RL78/G16 variations. The devices are shown according to memory and package type for easy selection.

**Table 2-4** ROM and RAM capabilities

Flash ROM	Data Flash	RAM	RL78/G16				
			10 pins	16 pins	20 pins	24 pins	32 pins
32 KB	1 KB	2 KB	R5F1211C	R5F1214C	R5F1216C	R5F1217C	R5F121BC
16 KB	1 KB	2 KB	R5F1211A	R5F1214A	R5F1216A	R5F1217A	R5F121BA

## RL78/G16 Group

### Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

To narrow the choices down to a specific device, first access the RL78/G16 Product page shown in Figure 2-9.

Then select Product Selection and use the filters on the top shown in Figure 2-10 to narrow down the peripheral functions.



Figure 2-9 RL78/G16 Product Page ([Link](#))

For example, to filter out the MCUs that do not meet the number of UART channels required, select the settings as shown in the blue box in Figure 2-10 using the filter tool. Doing so will display the MCUs that meet the specified conditions.

You can set other filters as well to find products that meet your requirements.

When selecting from the list, the popup display shows the stock availability, price, and other information as shown in Figure 2-11.

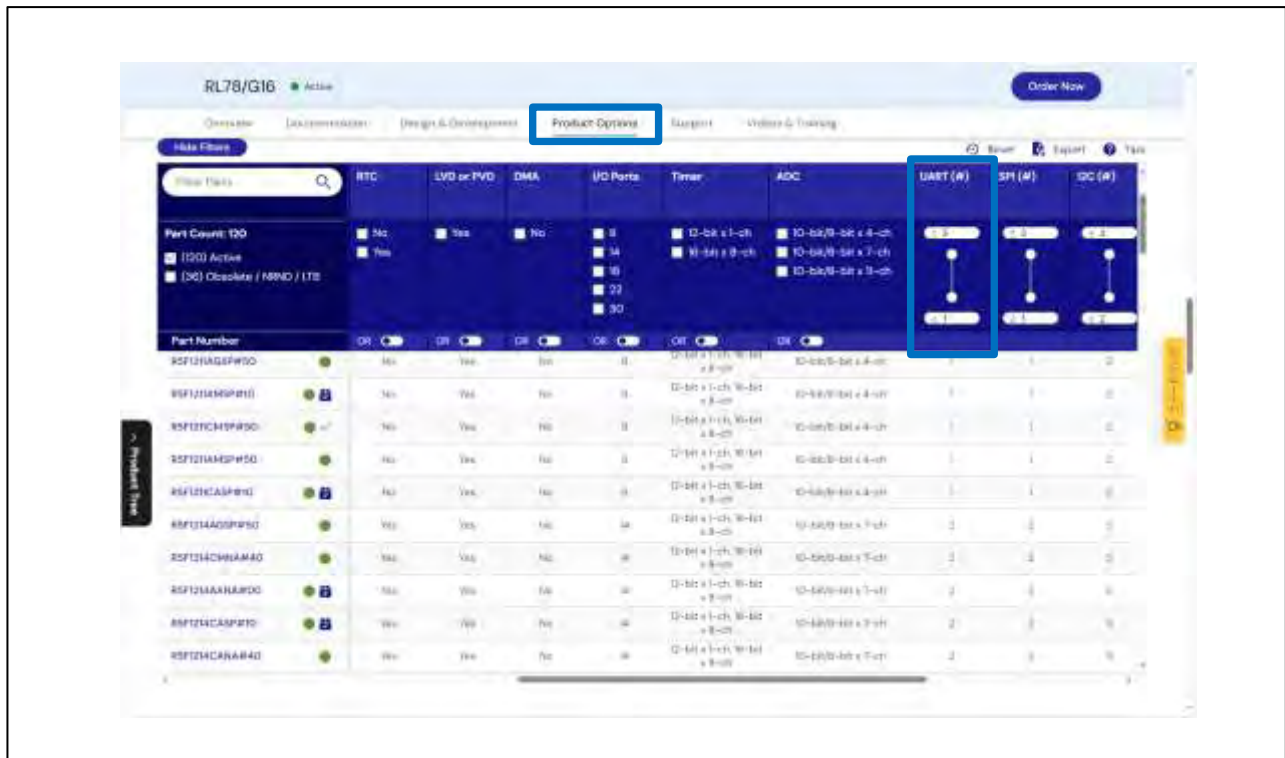


Figure 2-10 RL78/G16 Product Selection

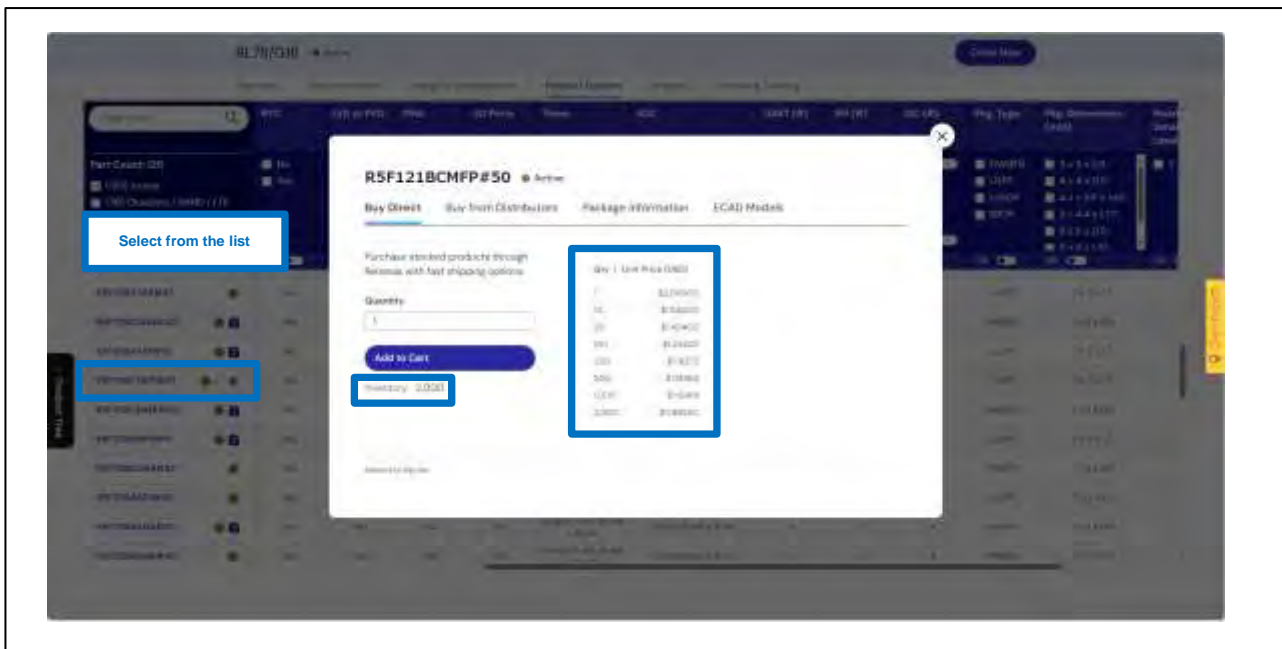


Figure 2-11 RL78/G16 Popup Display

The Documents section of the Products page lists documents such as the datasheets, manuals, and application notes, and allows you to easily select and download necessary documents (see Figure 2-12). The Datasheet provides a simple description of the RL78/G16 specifications. The Manual Hardware Version describes detailed specifications of RL78/G16. The Guide is for RL78/G16 engineers and lists the information and materials required for product development at each development phase. The website also lists technical documents related to RL78/G16 for easy user access.

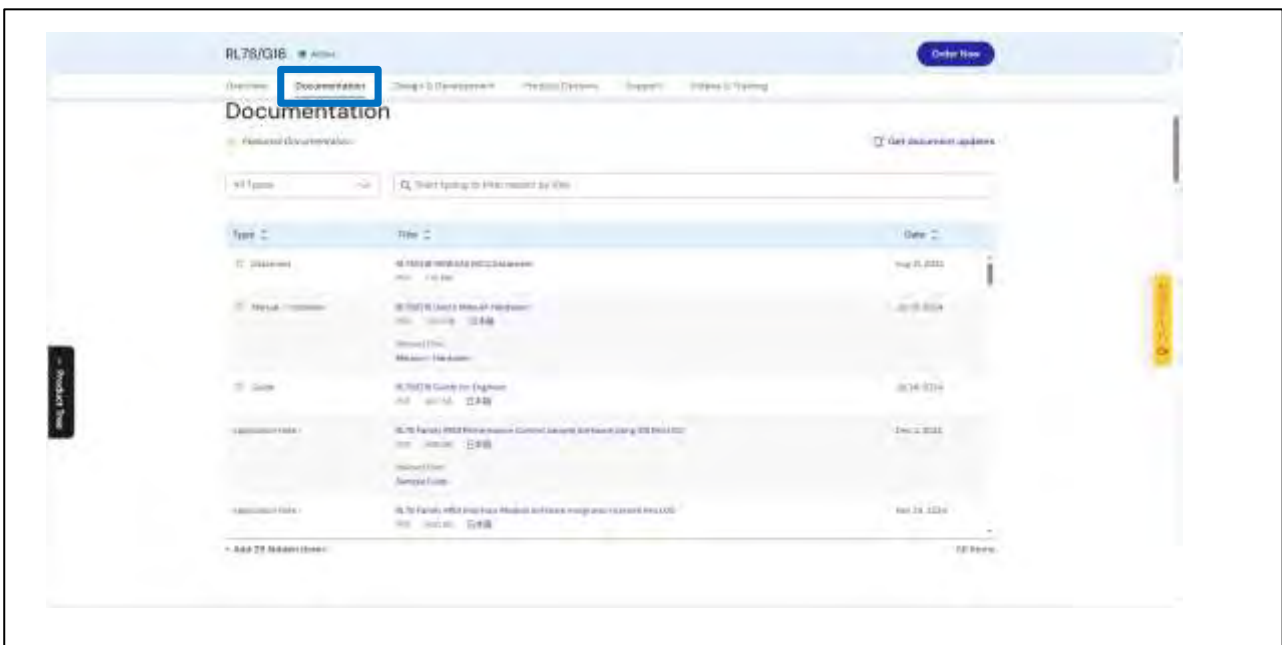


Figure 2-12 List of RL78/G16-related Technical Documents

## 2.2.2 Step 2: IDE Setup and Quick Introduction of e²studio

### 2.2.2.1 IDE Setup

- (1) Download the installer from the e²studio download link and start installation.  
**Select Lite Install for the installation type and click [Next].**

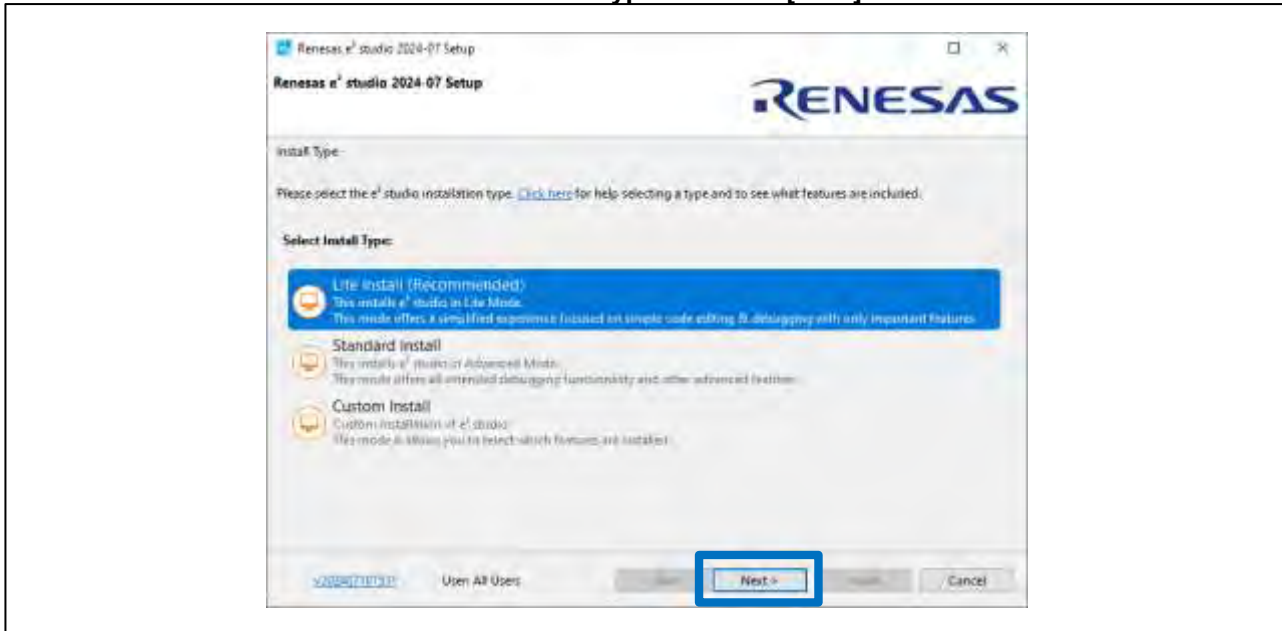


Figure 2-13 Select e²studio Installation Type

- (2) Select RL78 Family as the device.  
**Click [Next] and select the compiler and other necessary software.**

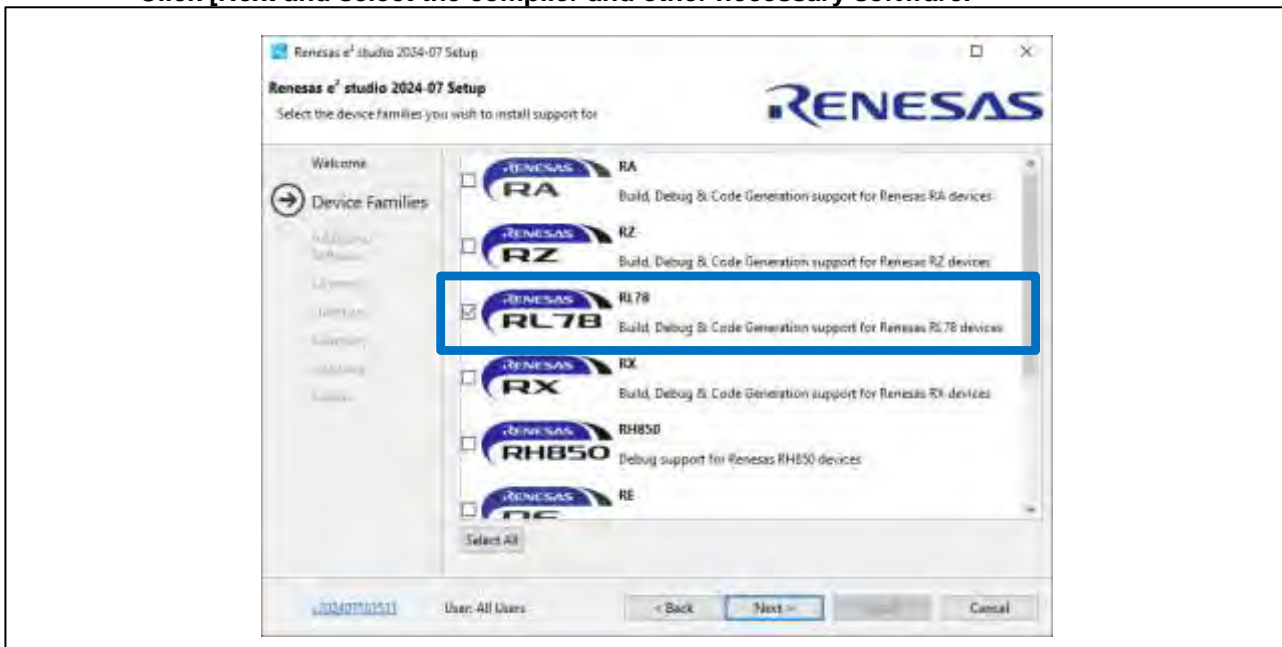


Figure 2-14 Select e²studio Device

(3) If the installation contents are acceptable, click Install to complete the installation of e<sup>2</sup>studio.

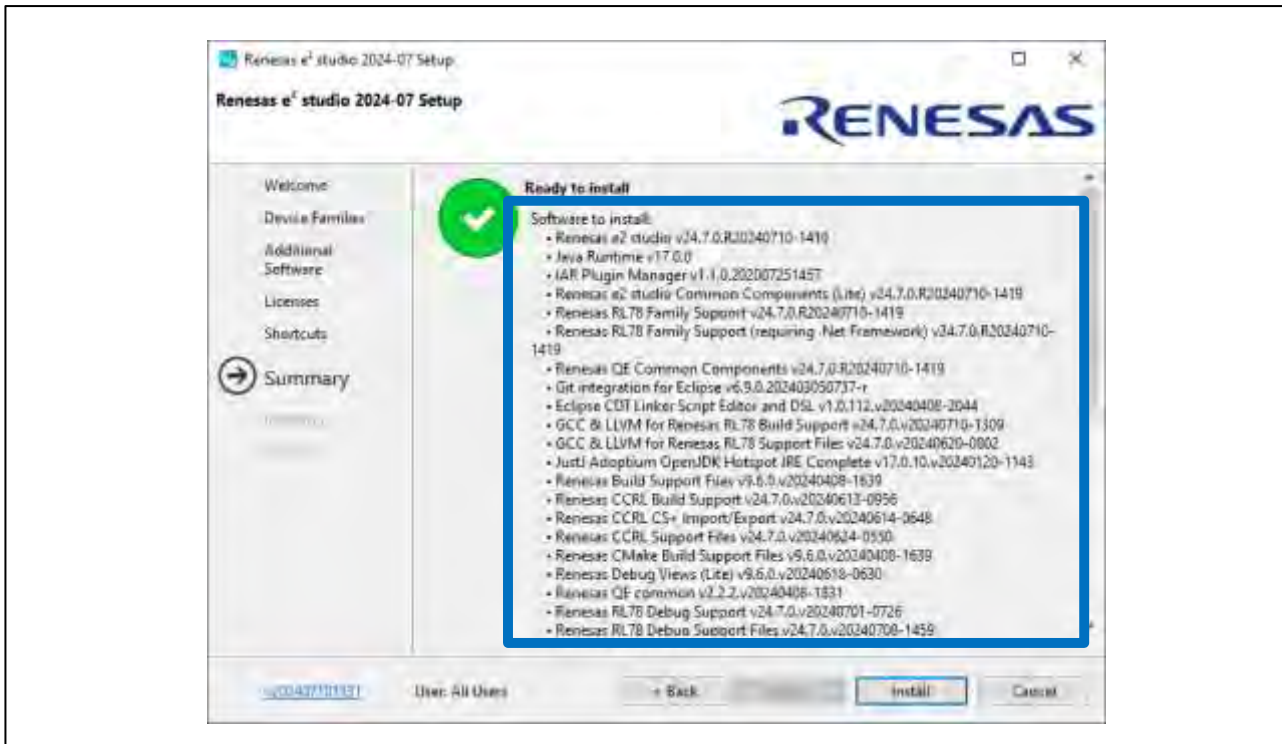


Figure 2-15 Confirm e<sup>2</sup>studio Installation Overview

2.2.2.2 How to Use e<sup>2</sup>studio

- (1) Start a new workspace. “Workspace” is the path for copying the imported projects. e<sup>2</sup>studio functions in the same way as MPLAB® X IDE.

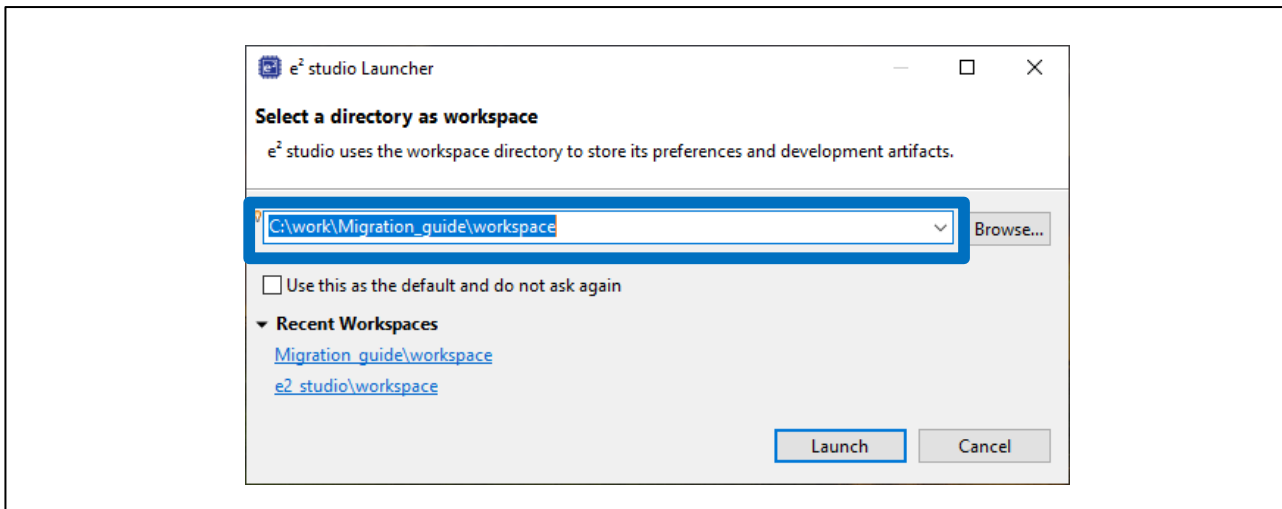


Figure 2-16 Select e<sup>2</sup>studio Workspace

- (2) To create a new project, select [File] -> New] -> [Renesas C/C++ Project] -> [Renesas RL78]. As with creating a new project in MPLAB® X IDE, configure the device, toolchain, and project name

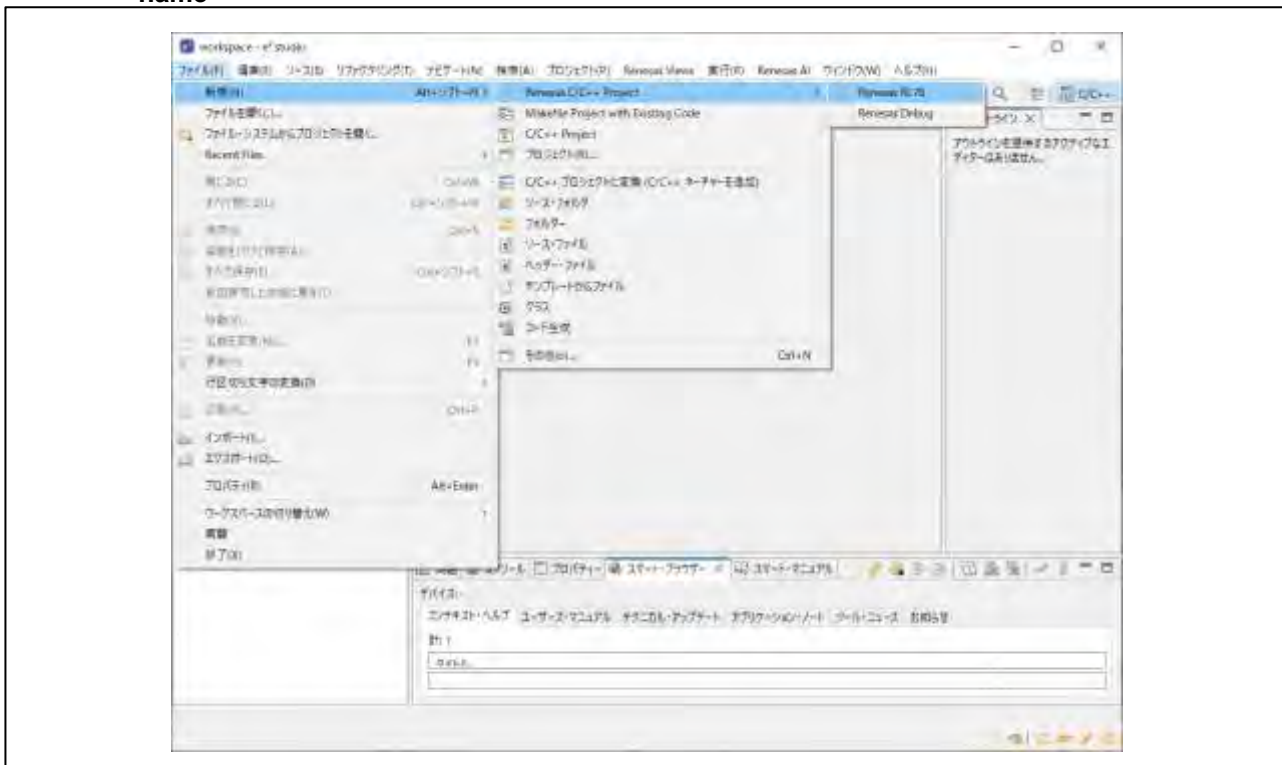


Figure 2-17 Create e<sup>2</sup>studio New Project and Select Device

(3) The new project confirmation screen will appear. Click Finish to generate the project.

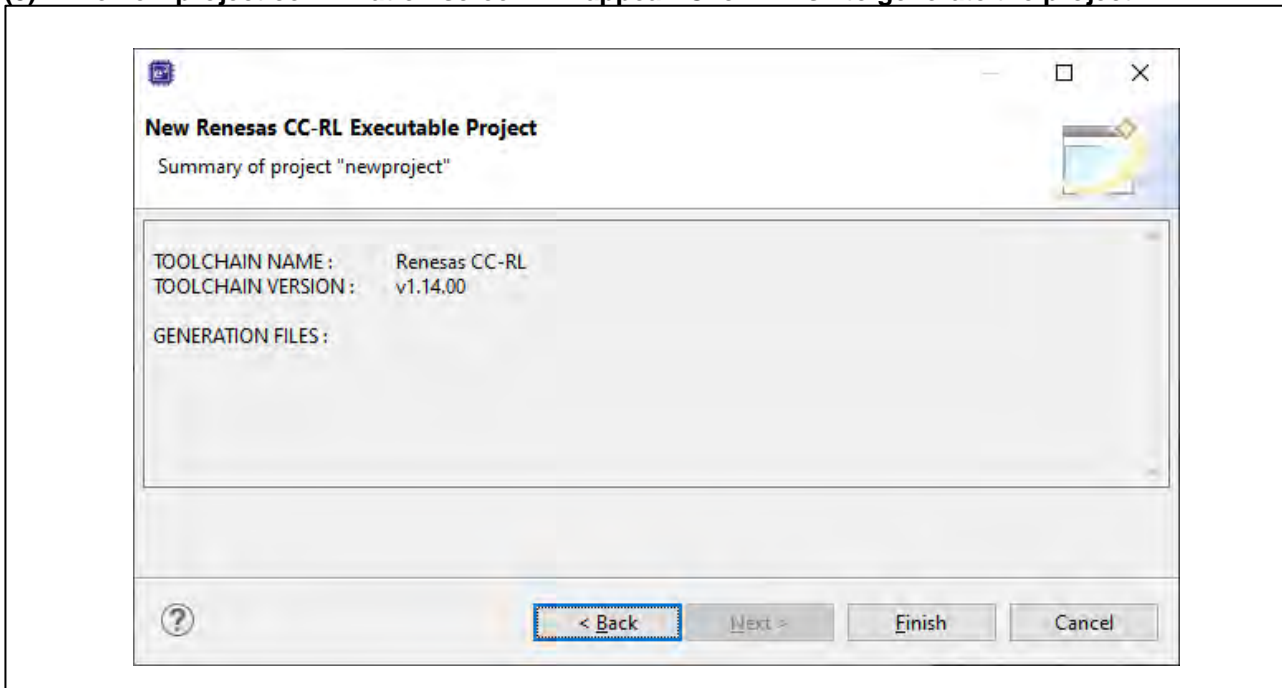


Figure 2-18 Create a New Project Overview

(4) The project is generated and opened in the workspace.

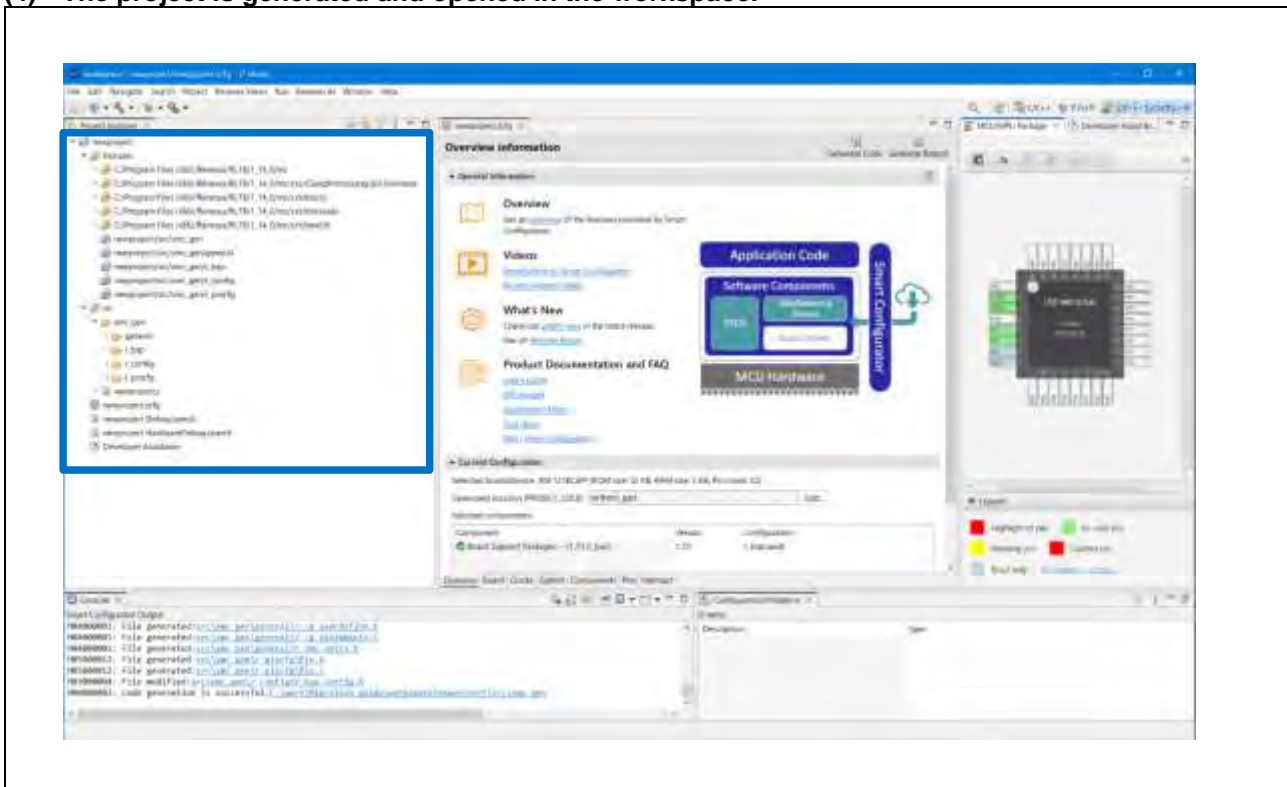


Figure 2-19 New Project Generation Completed

(5) Figure 2-20, Figure 2-21, Figure 2-22, and Figure 2-23 are quick introduction of the e<sup>2</sup>studio functions.

Shortcut Key Functions:

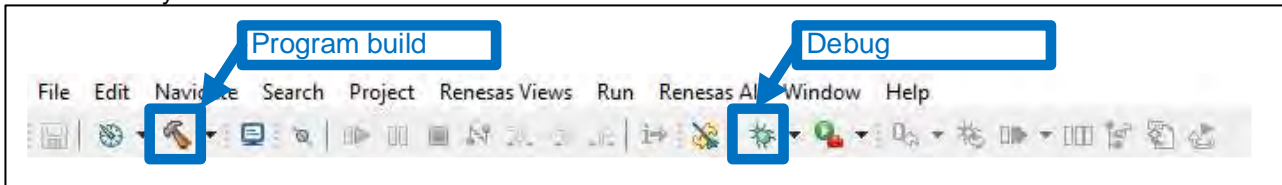


Figure 2-20 Commonly Used Functions

Debug Functions:

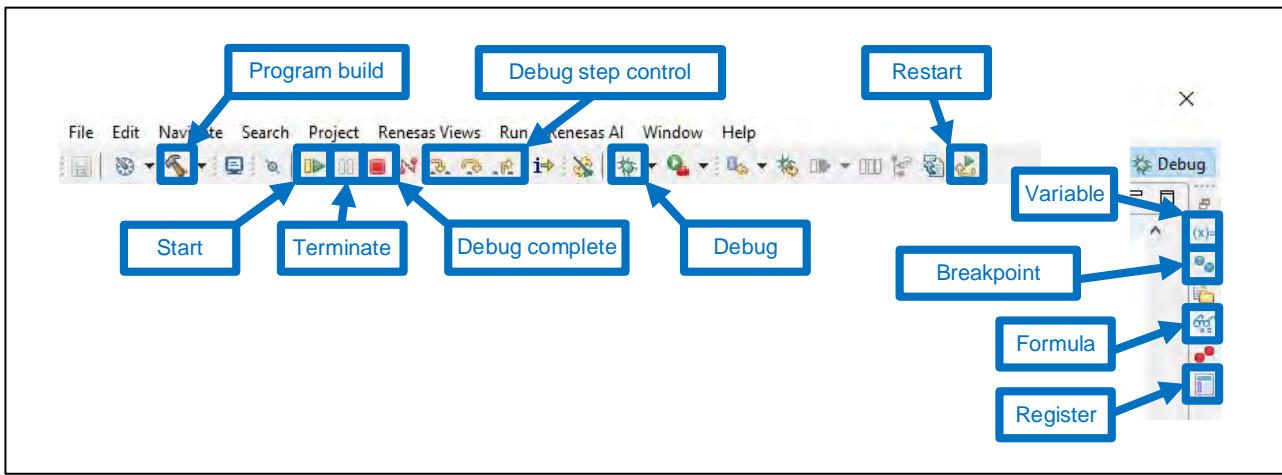


Figure 2-21 Commonly Used Debug Functions

Commonly Used Settings in the Project Properties:

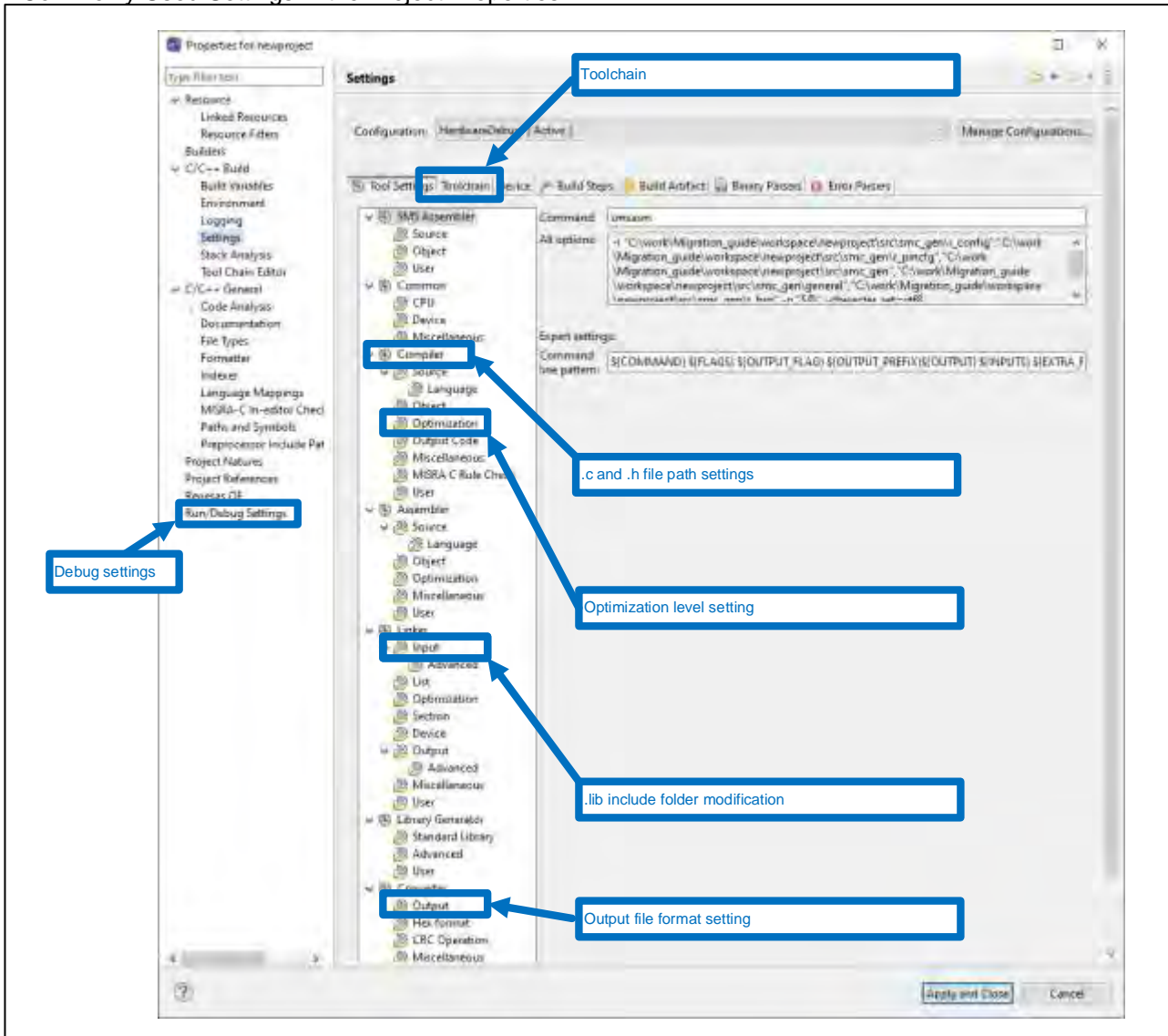


Figure 2-22 Commonly Used Project Settings (1/2)

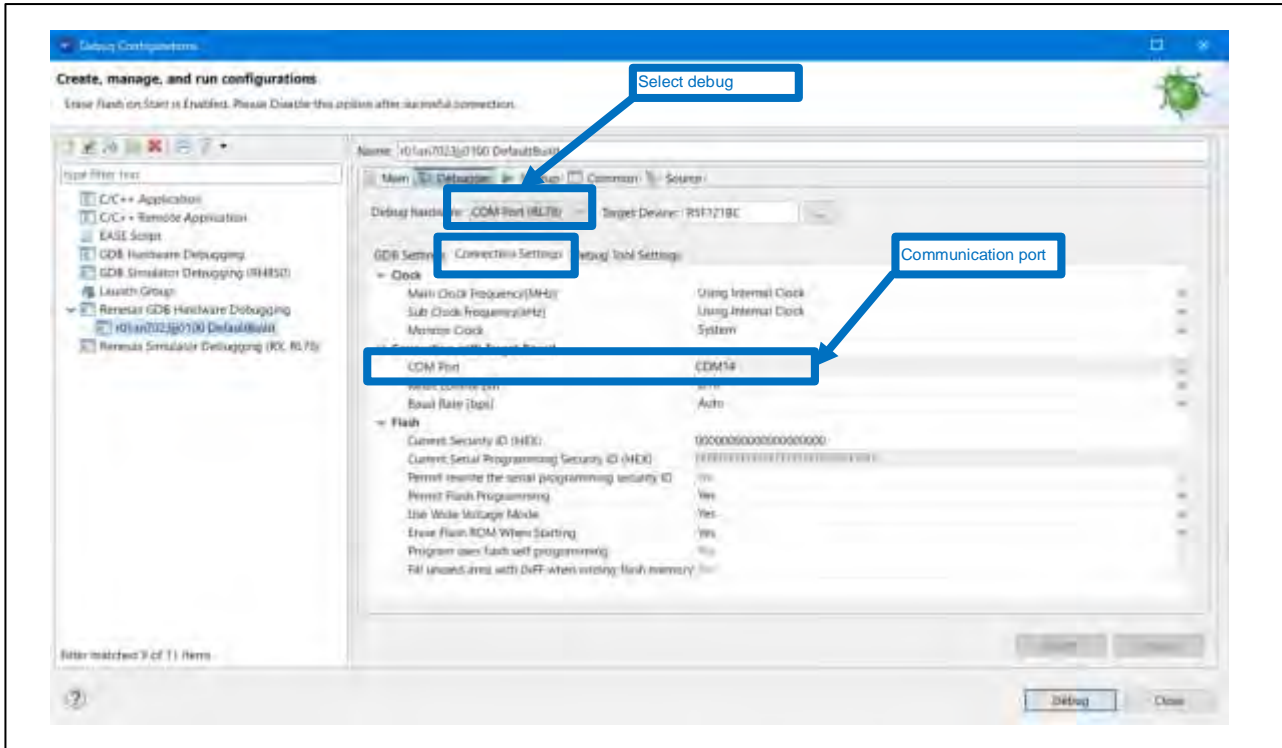


Figure 2-23 Commonly Used Project Settings (2/2)

Visit the Renesas Electronics website for details.

### 2.2.3 Step 3: Software importing example

This section provides quick guidance on how to import sample software to the e<sup>2</sup>studio.

#### (1) Import the downloaded sample code.

Select **[File] -> [Import]** from the menu to open the **Import** dialog box. In the **Import** dialog, select **[Projects]** from **[Folders or Archives]** and click **[Next]** to open the **Import Project** dialog box.

In the **Import Project** dialog box, click **[Archive]** button to select the sample program, then click **[Finish]** to add the sample code to the workspace.

Here, the sample program downloaded from the Renesas website is imported. In this example, the sample program used is “RL78/G16 Timer Array Unit (Interval Timer) Rev. 1.00 – Sample Code”.

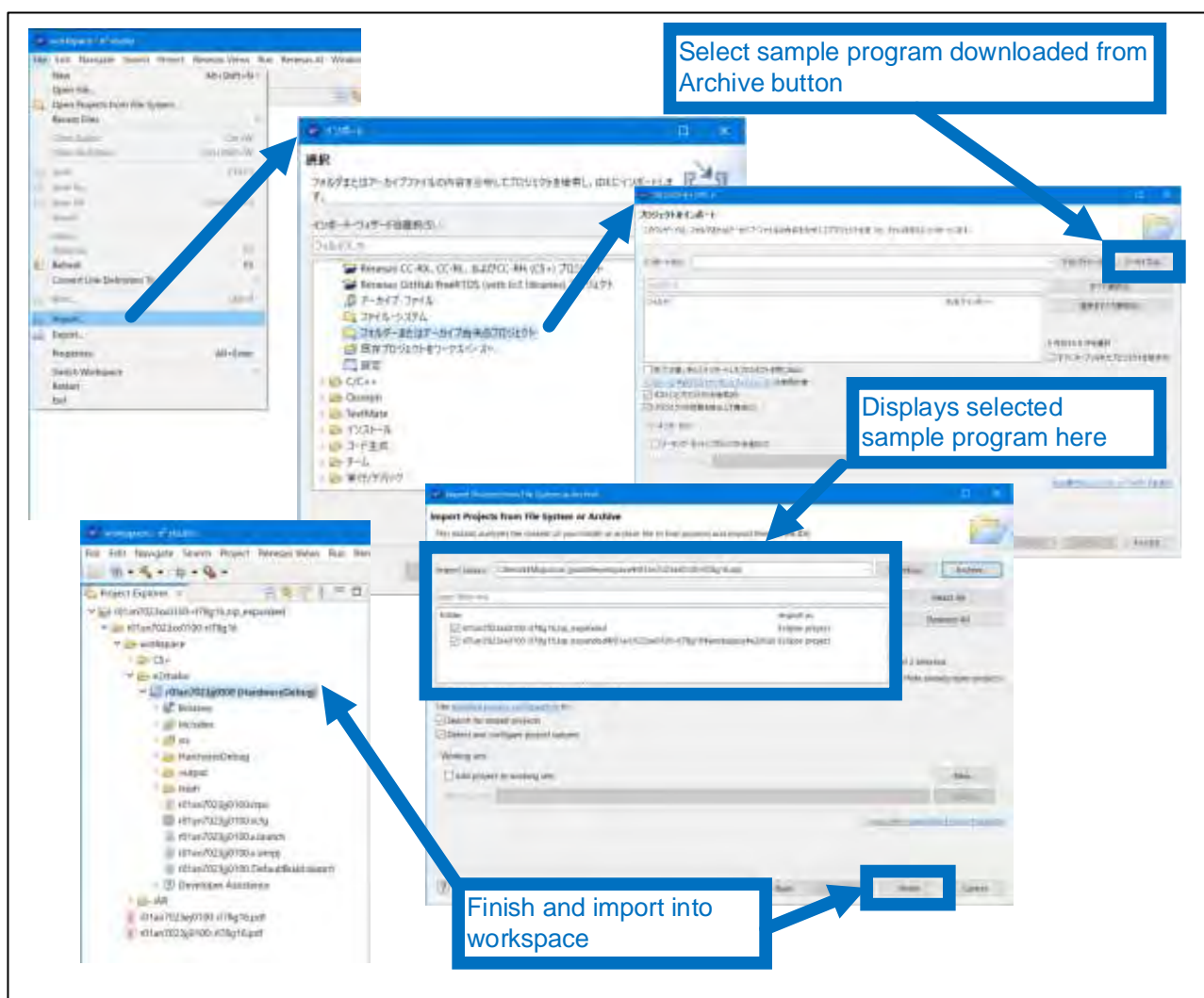


Figure 2-24 Import the Sample Program

(2) Figure 2-25 shows the most important files in the project.

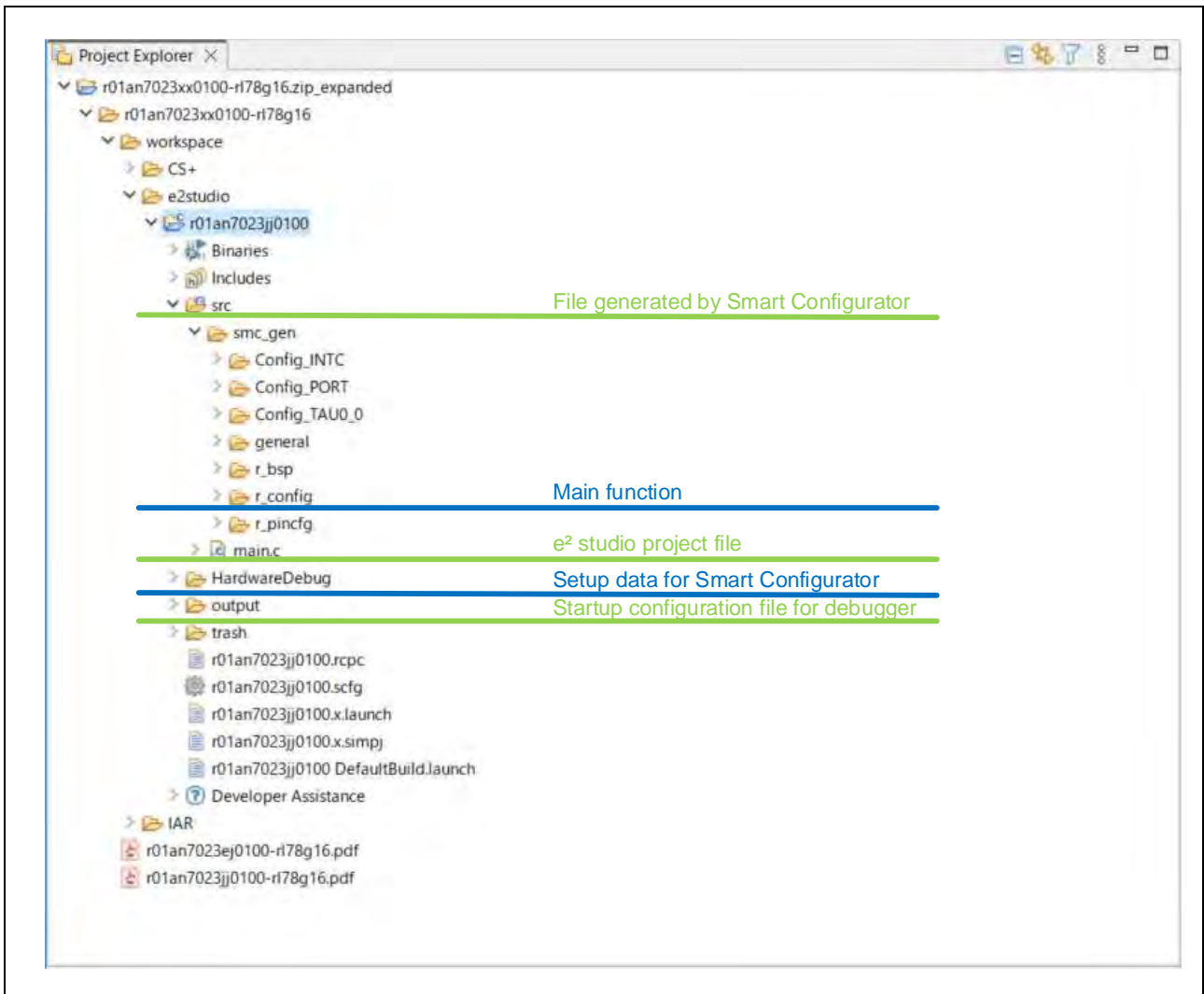


Figure 2-25 e2studio Project Files

(3) Double click the .scfg file to start the Smart Configurator. This will allow you to configure the necessary peripheral functions via the graphical interface.

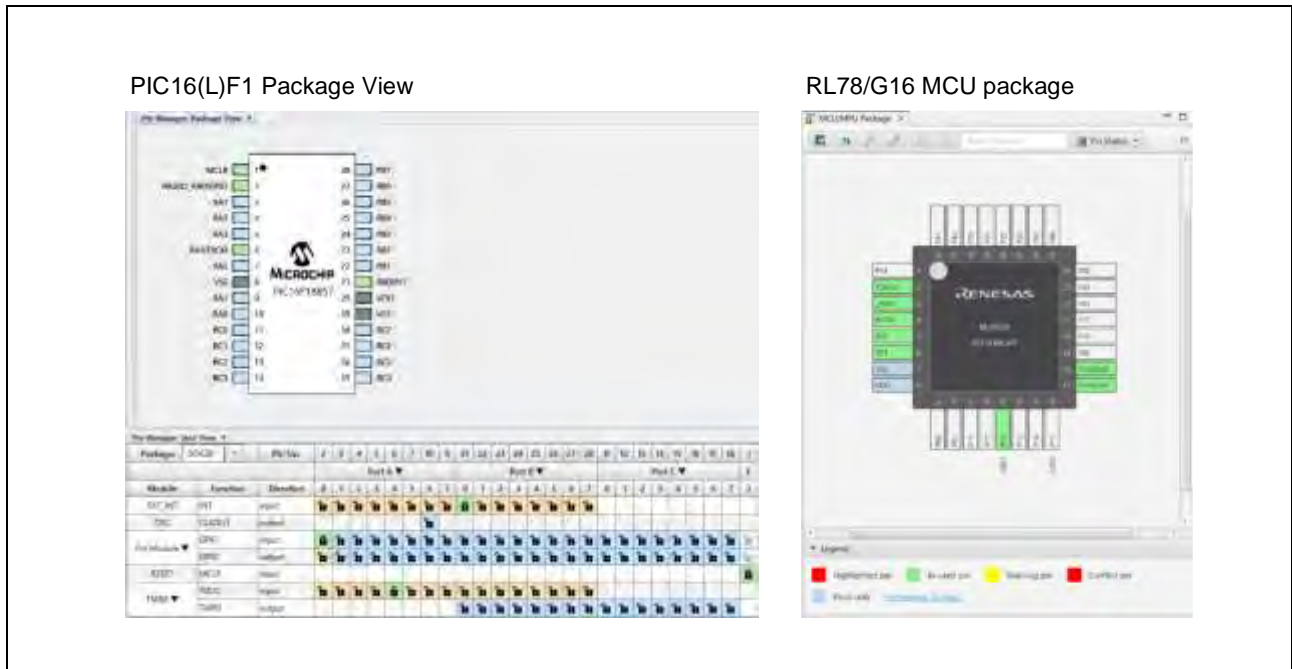


Figure 2-26 MCU View of MPLAB® Code Configurator and Smart Configurator

(4) The user can improve and add functions to the sample program as needed by referring to the corresponding application notes published by Renesas.

(5) When the software configurations or modifications are complete, click the [Build] icon on the Main toolbar as shown in Figure 2-27.

When the building is done successfully, “Build Finished” is displayed.

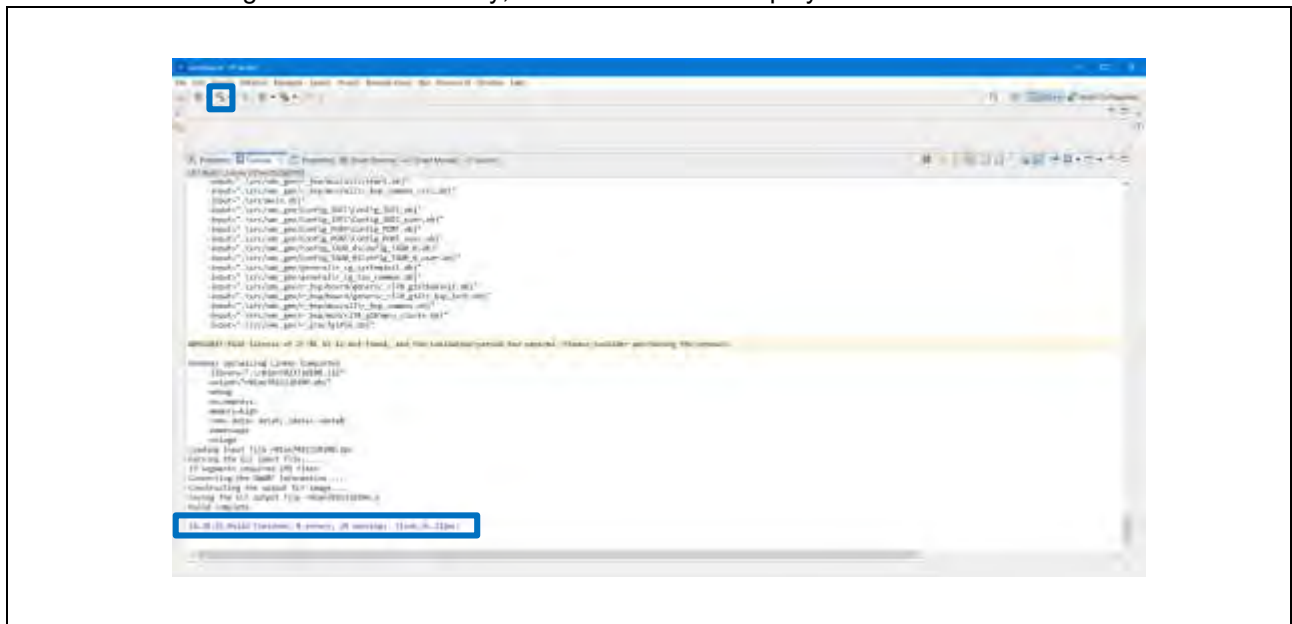


Figure 2-27 Build Finished

## Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

When building the sample program, errors related to the Toolchain may appear. If this kind of error appears, select the project and select **[Project] - [Property]** to open the **Properties** dialog box. In the **Properties** dialog box, select **[C/C++Build] - [Settings] - [Toolchain]** tab to change the toolchain settings.

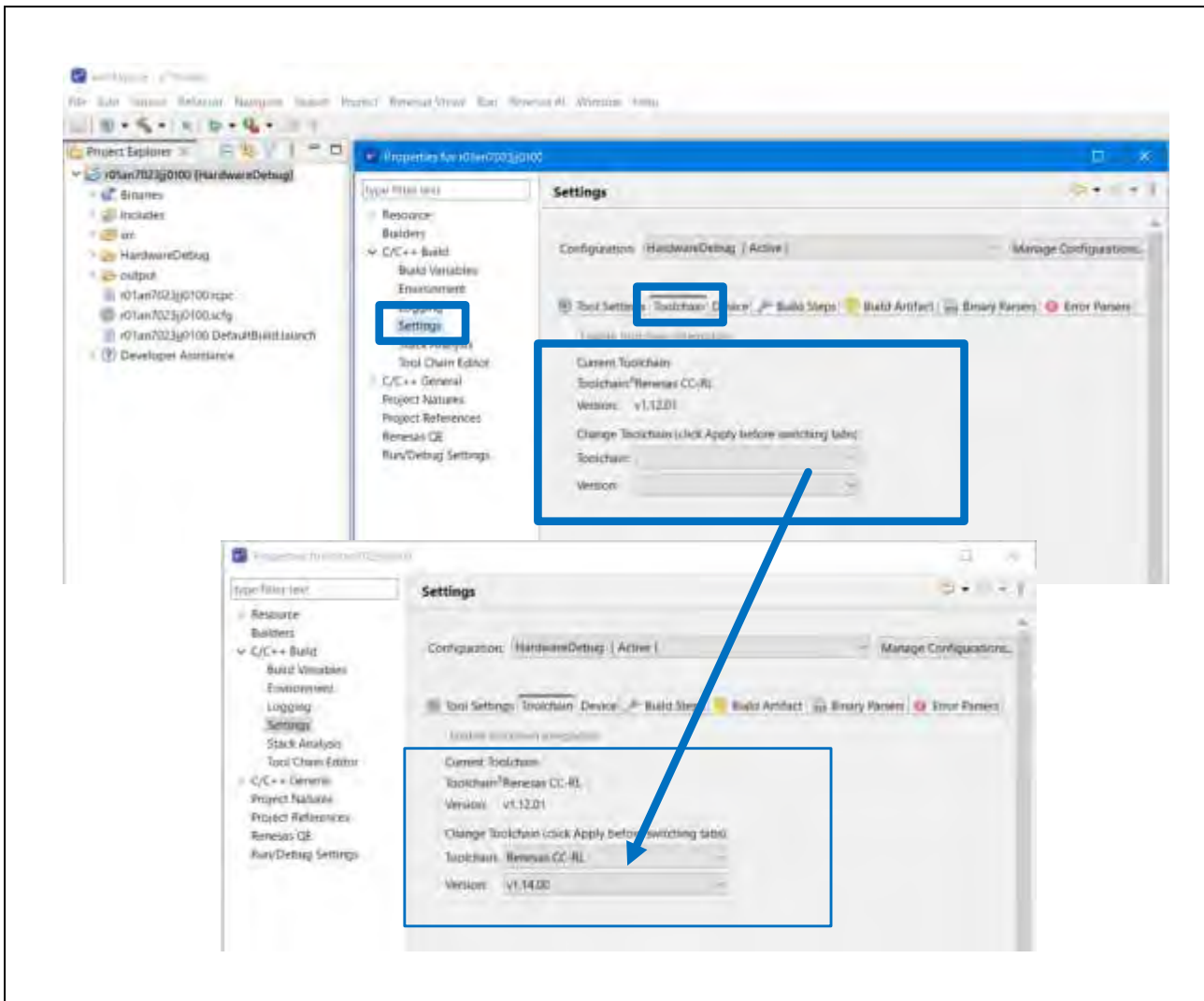


Figure 2-28 Toolchain Settings

### 2.2.4 Step 4: Evaluating the Software

This section shows the way to check software's operation and performance by using debugging function described in 2.2.2.2 How to Use e<sup>2</sup>studio chapter.

### 2.2.5 Step 5: Mass Production

(1) Generate the load module (.hex, .mot, .bin) using e<sup>2</sup>studio.

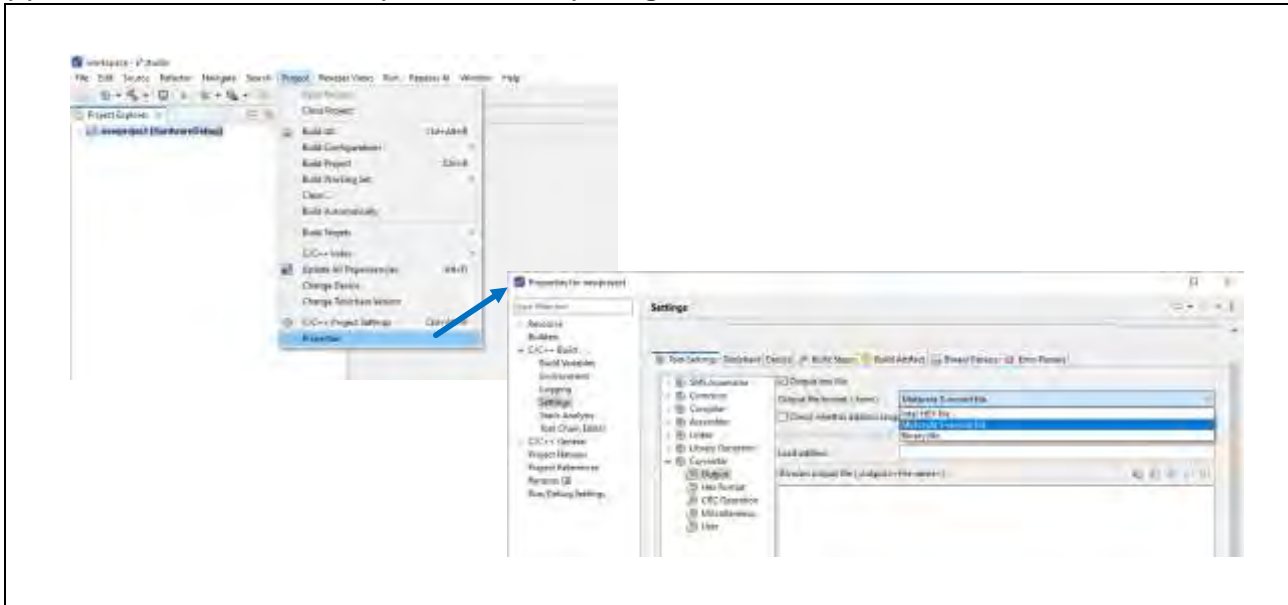


Figure 2-29 Create the Program File

(2) Select the programmer/debugger to program RL78/G16.

Up to 12 PG-FP6 units can be connected to one PC, allowing simultaneous programming to the target MCUs.

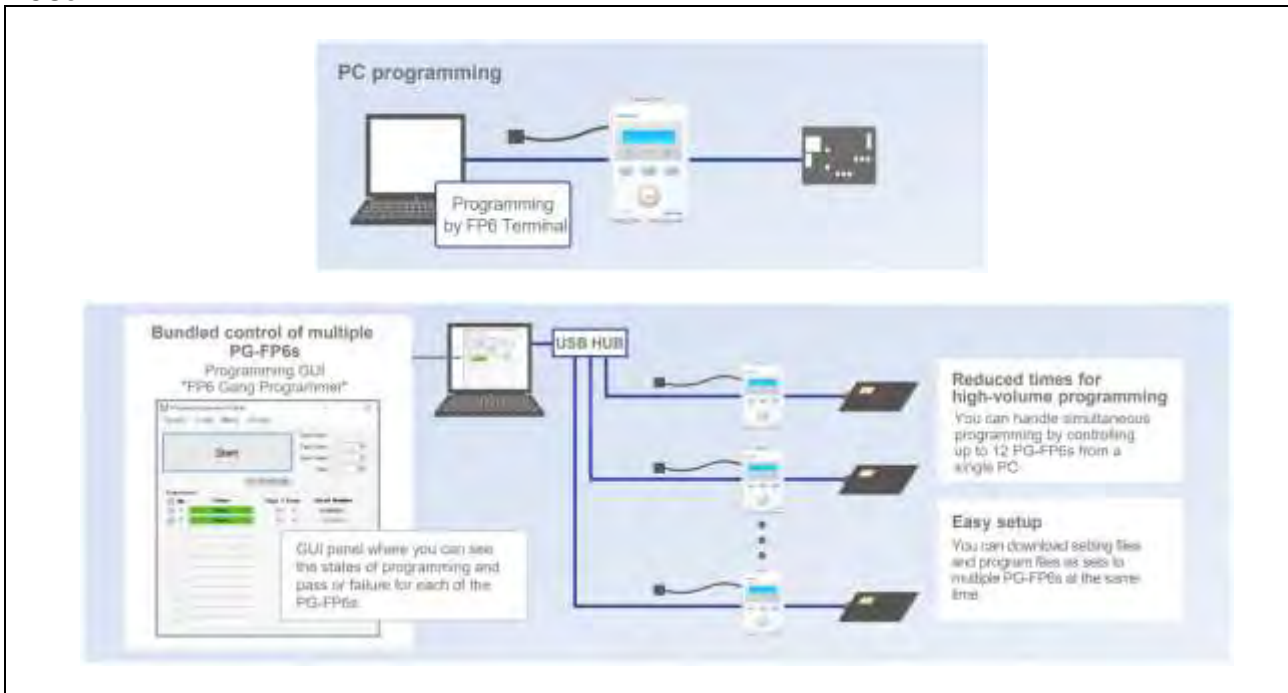


Figure 2-30 Flash Programming and Tools

## RL78/G16 Group

### Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

---

For details regarding terminals PG-FP6 and FP6, refer to the Flash Memory Programmer User's Manual (R20UT5518).

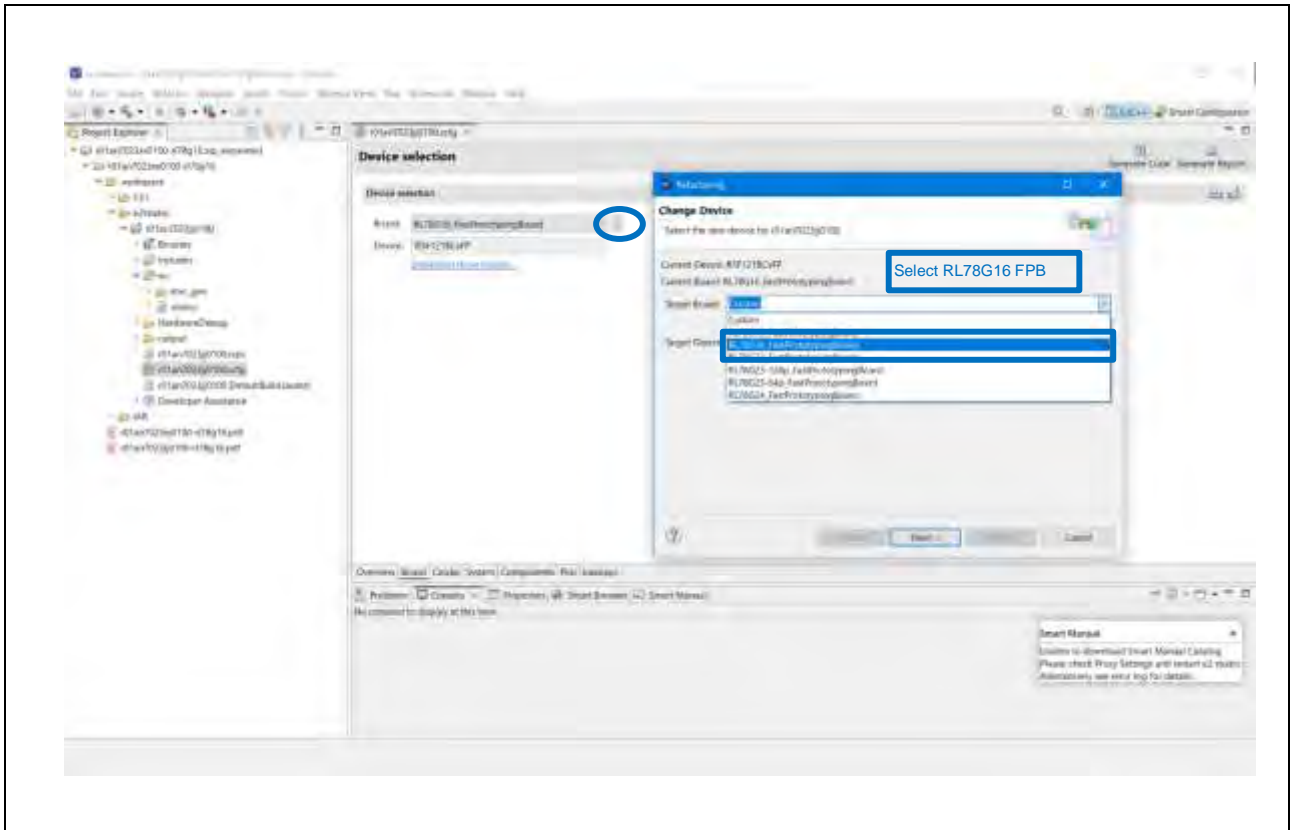
For details regarding PG-FP6 Gang Programmer, refer to the Gang Programmer User's Manual (R20UT5403).



**(4) Modify the project.**

To check the configuration with the Smart Configurator, open the Smart Configurator with the .scfg file.

If the hardware is different from the hardware in the sample program, you will need to modify the configuration. To change the board, select the Board tab from the Smart Configurator. Click [...] as shown in Figure 2- to open the **Refactoring** dialog box and select the target board from the dropdown list in the Target Device field. “RL78G16\_FastPrototypingBoard” is selected in the example.



**Figure 2-32** Select the Device in Smart Configurator

Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

LED1, which is controlled by the sample software, is assigned to I/O port P12 and can be confirmed from the Smart Configurator. To check the port setting, select **[Driver] - [I/O port] - [Config\_PORT]** in the **Components** tab, then select the **PORT1** tab, as shown in Figure 2-.

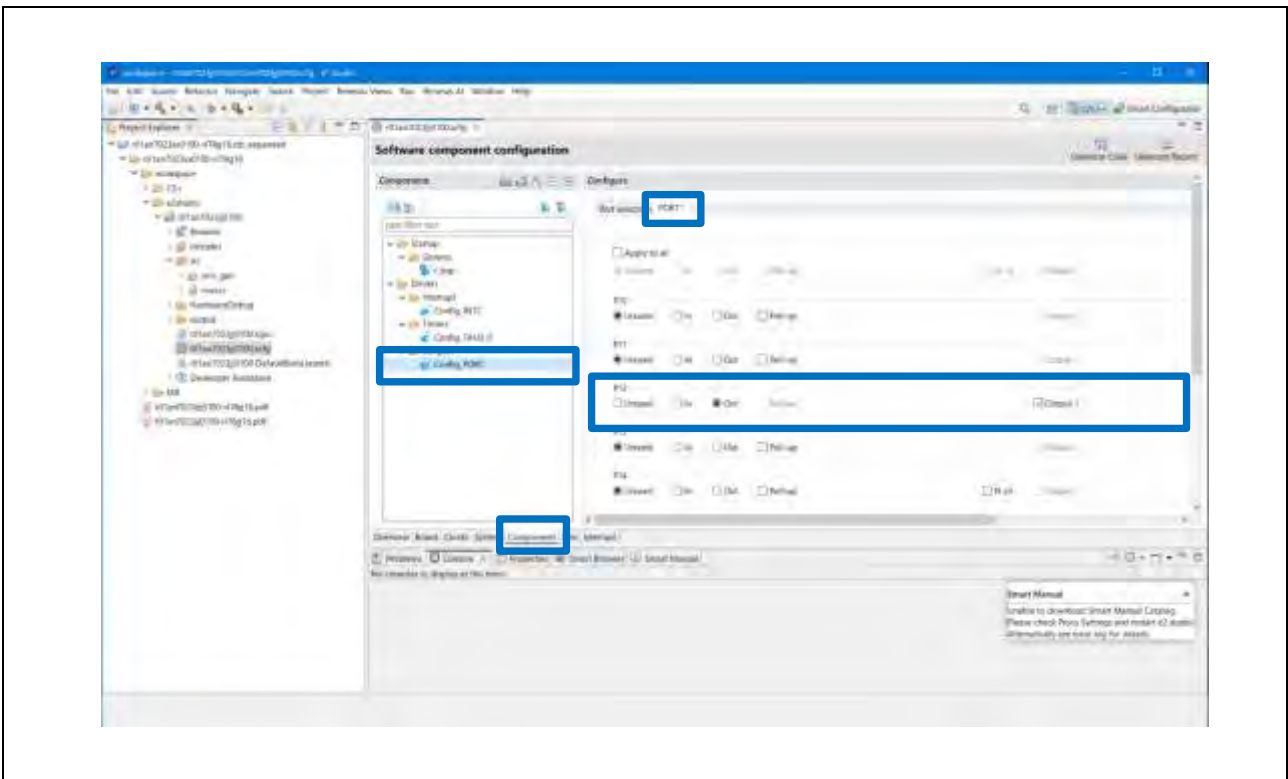


Figure 2-33 Port Settings in Smart Configurator

The port is controlled by the function called from the interrupt processing.

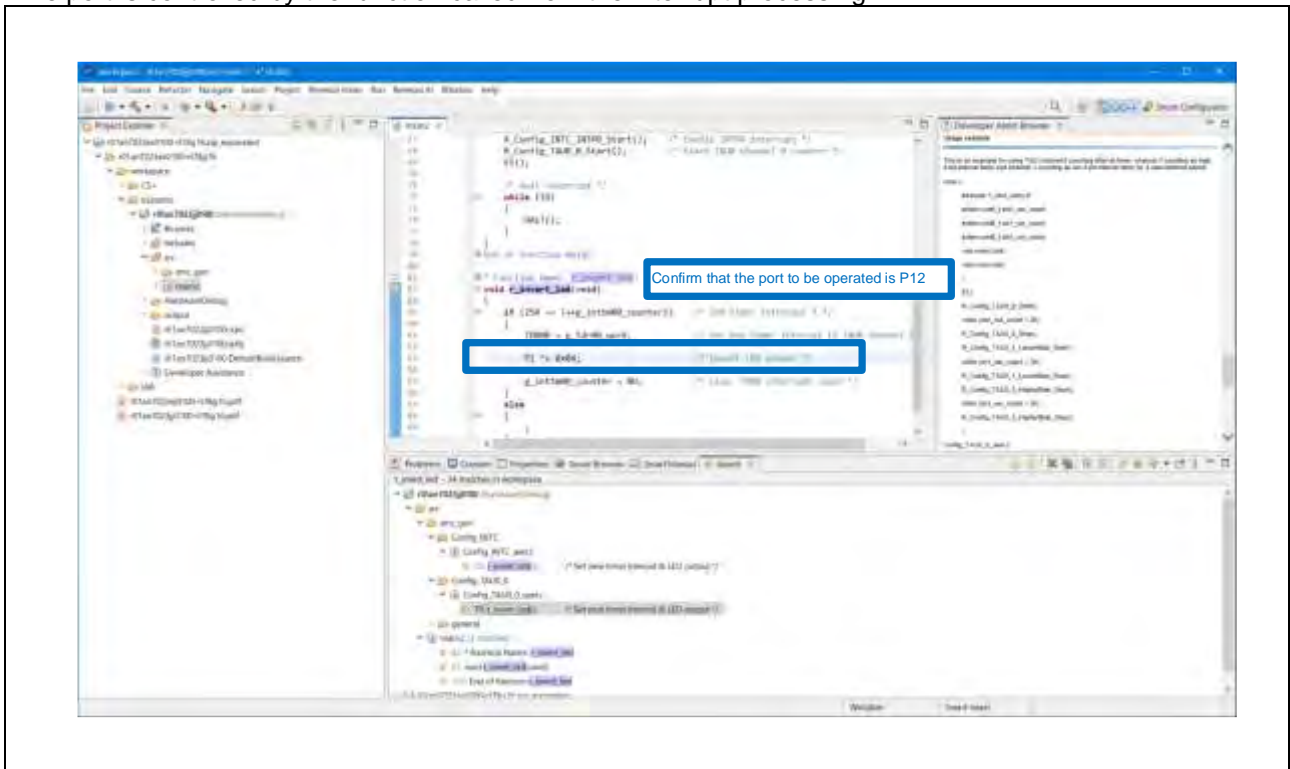


Figure 2-34 Interrupt Processing

Next check the timer setting. Select **[Driver] - [Timer] - [Config\_TAU0\_0]** sections as shown in Figure 2-. This enables you to confirm timer clock configurations such as clock source and interval duration. The interval duration is set to 2 ms here.

Click the **Generate Code** button in the Smart Configurator to generate the source code.

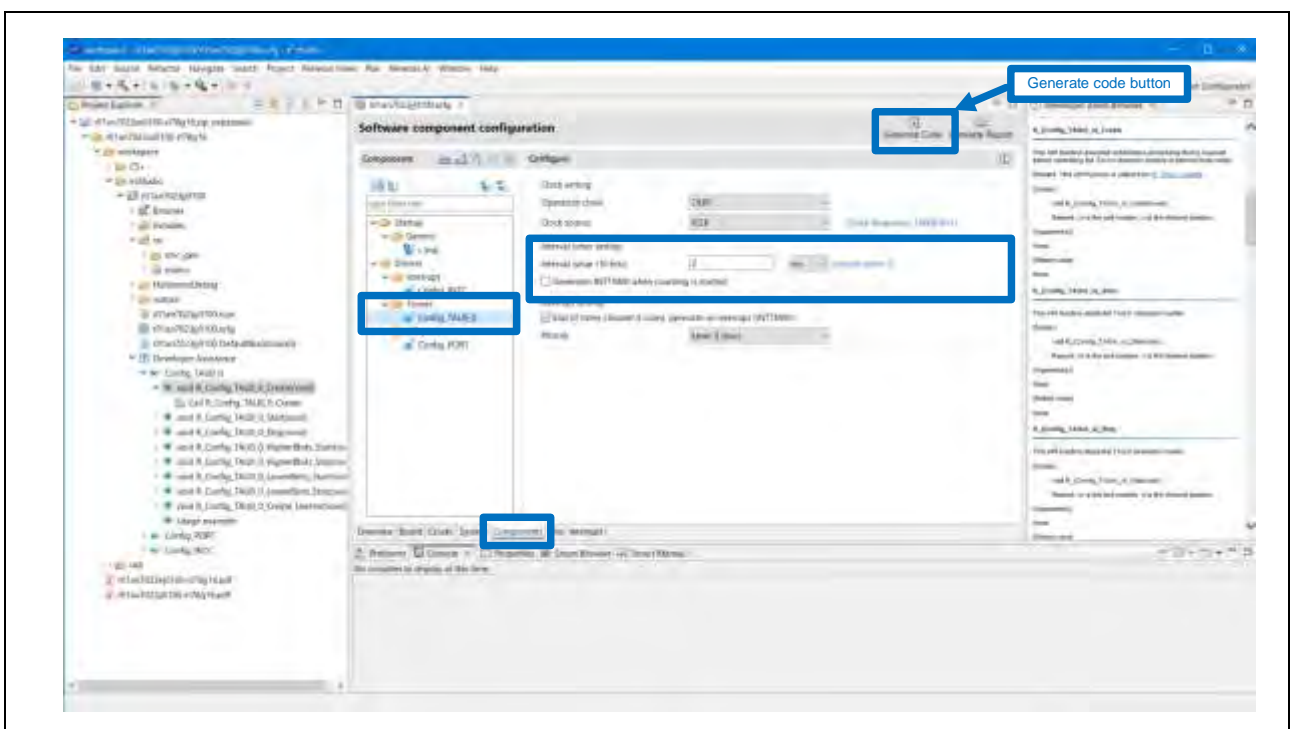


Figure 2-35 Interval Timer Setting in Smart Configurator

## Migration Guide from Microchip PIC®16(L)F1 to RL78/G16 (Software Edition)

The Developer Assistance can be used to display the API information and sample code for each component to assist in your development.

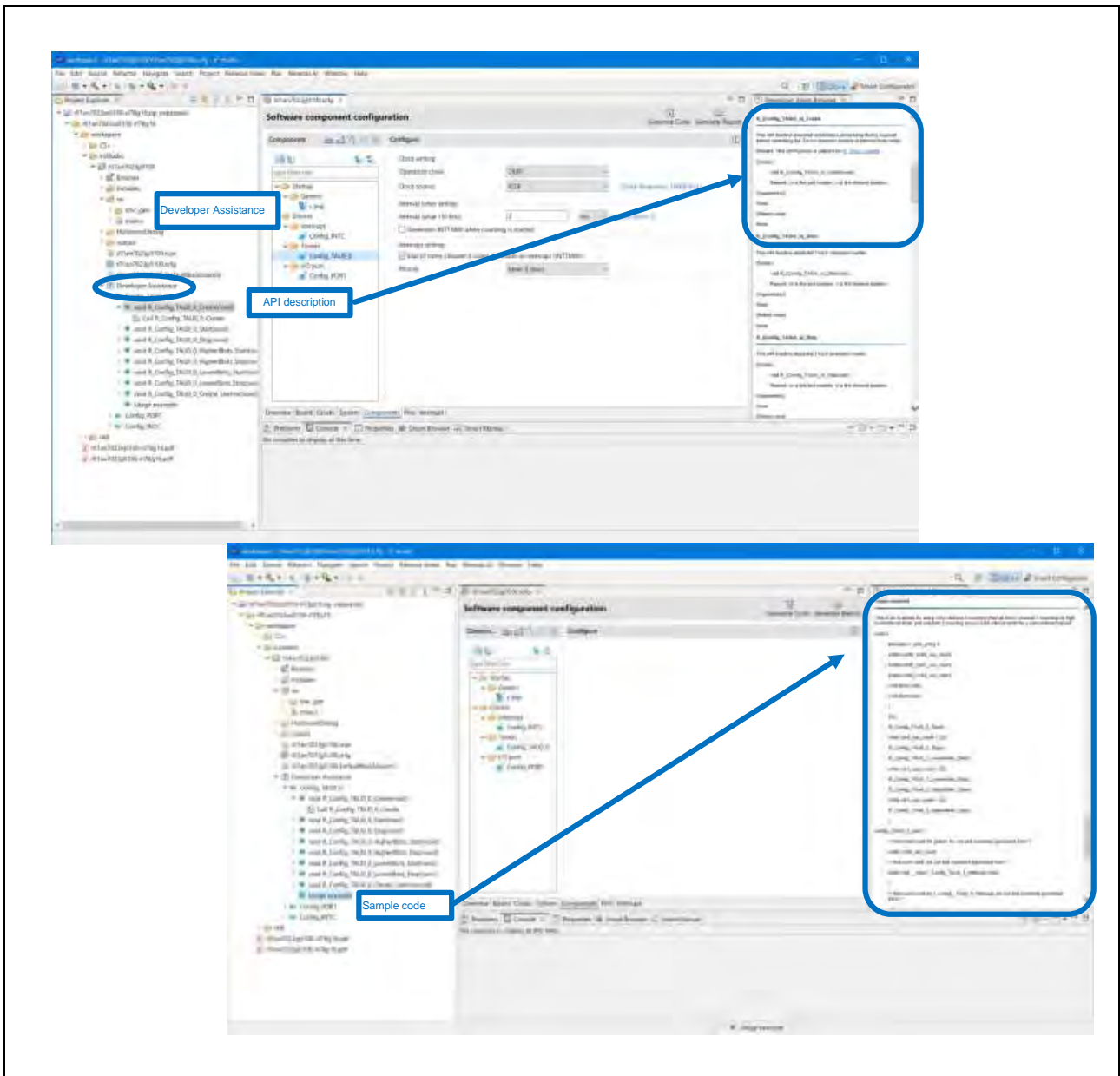


Figure 2-36 API Information Displayed with Developer Assistance

**(5) Build the Project**

When the software configurations or modifications are complete, click the **Build** icon in the Main toolbar as shown in Figure 2-.

When the building is done successfully, “Build Finished” is displayed.

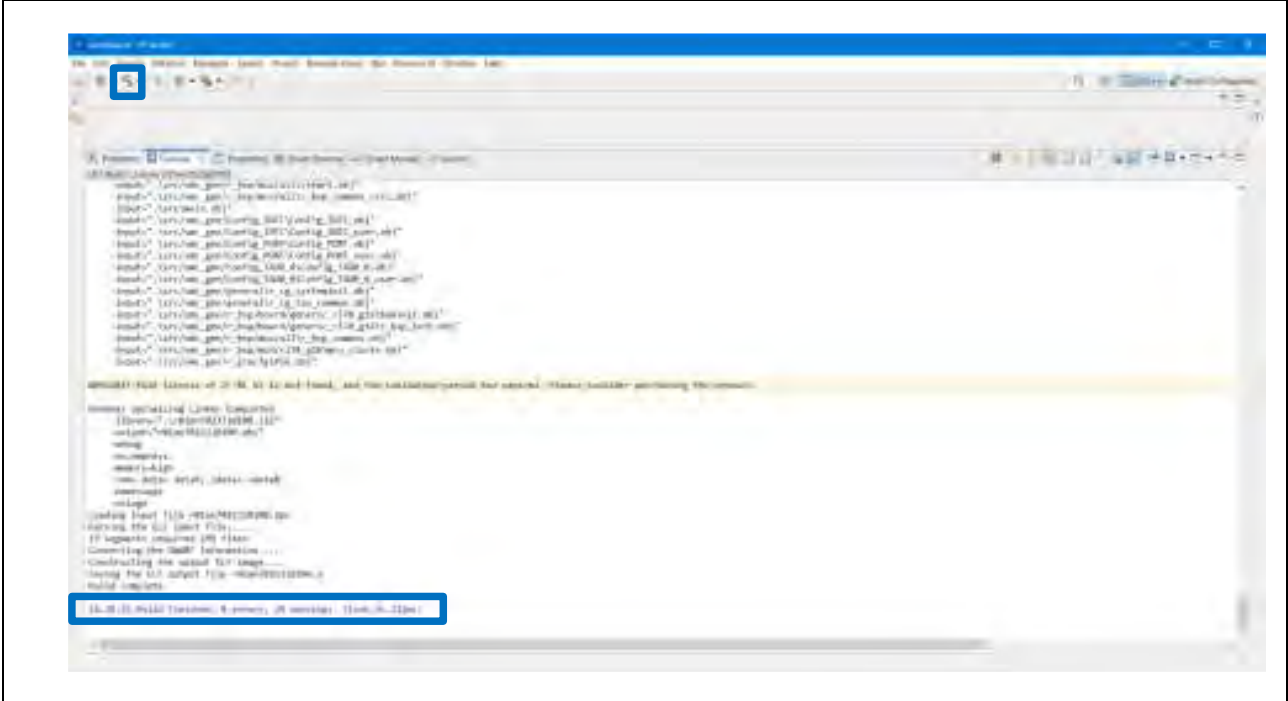


Figure 2-37 Build the Target

**(6) Setup the Hardware.**

Connect the RL78/G16 FPB to the PC using a commercially available USB cable (one end should be MicroB male).

**(7) Debug to verify.**

Configure the debugger according to エラー! 参照元が見つかりません。 before debugging.

Click the **Debug** icon to start debugging. To set a breakpoint, double click the space before the line number.

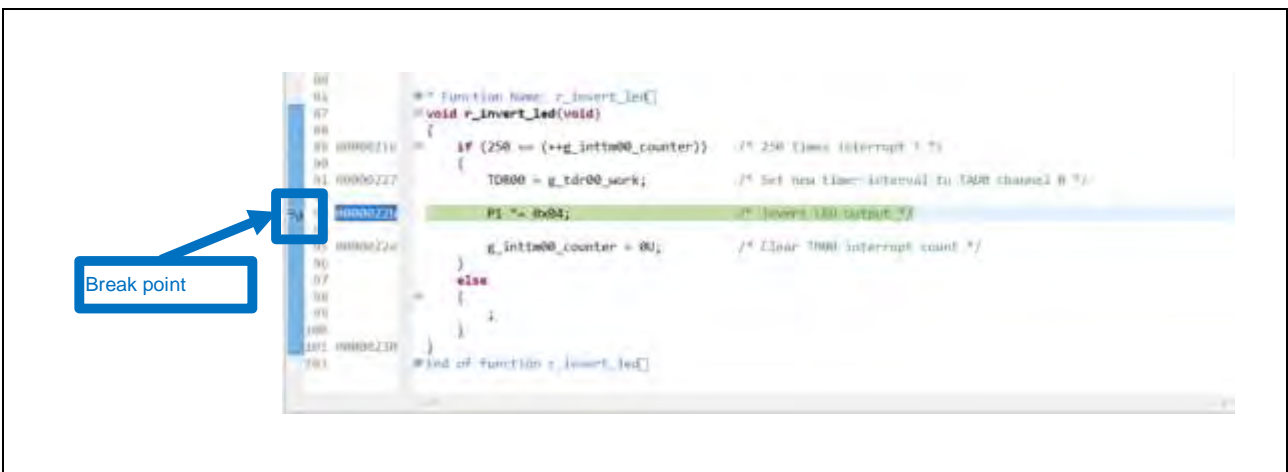


Figure 2-38 Adding Break Point Solution

Use the debug function to check if the performance sufficiently meets the requirements and no errors occur in the program (refer to 2.2.2.2 How to Use e2studio for details). During debugging, the LED blinks while the code is being executed. In addition, the blinking cycle can be changed with the user switch.

### 3. Core Architecture Comparison

#### 3.1 CPU

The RL78 Family is based on the innovative RL78 core architecture and PIC16(L)F1 is based on the Enhanced Mid-Range CPU core architecture. Table 3-1 shows a general functional overview, comparing the RL78/G16 CPU to the PIC16(L)F1.

**Table 3-1** CPU Comparison

Item	PIC16(L)F1	RL78/G16
Architecture	Enhanced Mid-Range CPU	RL78-S2 Core
Instruction set	RISC	CISC
Pipeline	2-stage	3-stage
Operating clock (max.)	32 MHz	16 MHz
Minimum instruction execution time	0.125µs @32 MHz	0.0625µs @16MHz
Address space	32KB	1MB
Number of instructions	49	75
Multiplier	-	8-bit x 8-bit = 16-bit

## 3.2 Built-in Memory Comparison

### 3.2.1 Flash Memory Function

RL78/G16 and PIC16(L)F1 have non-volatile flash memories for storing the executable program code and the application data. Table 3-2 shows a comparison of flash memories.

**Table 3-2** Flash Memory Function Comparison

Function	PIC16(L)F183xx	RL78/G16	
Flash memory	3.5 to 28 KB	Code flash	16 to 32 KB
		Data flash	1 KB
Memory configuration	Word size: 14 bits Row size: 32 words When the memory size is 28 KB: 16384 words (512 rows)	Code flash	Block size: 1 KB
		Data flash	Block size: 512 bytes
Access	14 bits (1 word)	8 bits or 16 bits	
Programming unit	1 row (32 words)	Code flash	32 bits
		Data flash	
Erasing unit	1 row (32 words)	Code flash	1 KB
		Data flash	512 bytes
Data error correction/detection	N/A	General-purpose CRC (peripheral function)	
Specific sequence	Unlock Sequence	N/A	
Security	Code Protection Write Protection	Read Protection from debugger	
Programming method	In-circuit serial programming Self-programming	Serial programming Self-programming	
Programming count	10 k (min.)	Code flash: 1 k (min.) Data flash: 1000 k (typ.)	

### 3.2.2 Flash Memory Configuration

The RL78/G16 flash memory is equipped with a “code flash” for executing programs and a “data flash” used as a data storage area.

The code flash and the data flash can be rewritten by the flash memory programmer, serial programming with the external device (UART communication), or self-programming.

#### 3.2.2.1 Flash Memory Area

Table 3-3 shows the flash memory areas in PIC16(L)F183xx and RL78/G16.

**Table 3-3** Flash Memory Area Comparison

PIC16(L)F183xx		RL78/G16 (Code Flash)	
Programming area	Reset vector	Program area	Application
	Interrupt vector	Other than above	On-chip debug security ID setting
	Application		Option bytes
Other than above	User ID		CALLT table
	Revision ID		Vector table
	Device ID		
	Configuration Words		
Calibration Words			

RL78/G16 and PIC16(L)F1 are equipped with the following non-volatile memories (EEPROM, Data flash) as the data storage area.

Table 3-4 shows a comparison of the data storage areas in PIC16(L)F183xx and RL78/G16.

**Table 3-4** Data Storage Area Comparison

Type	PIC16(L)F183xx	RL78/G16
Data storage area	EEPROM	Data flash
Access	8 bits	8 bits or 16 bits
Programming unit	8 bits	32 bits
Programming count	100 k (min.)	1000 k (typ.)

#### 3.2.2.2 Flash Memory Control Register in RL78/G16

The flash memory control registers in the special function registers (SFR and 2nd SFR) are used to control the flash memory self-programming. For example, the FLPMC register can enable or disable programming of the code flash and data flash memories, and the FSSQ register is used to control operations and select commands for the flash memory sequencer.

#### 3.2.2.3 Self-Programming in RL78/G16

RL78/G16 supports the self-programming mode that enables rewriting the flash memory with the user program.

In this mode, the flash memory can be rewritten with a user application by using the RL78/G16 Renesas Flash Driver.

### 3.2.3 On-chip RAM

RL78/G16 and PIC16(L)F1 are both equipped with an on-chip RAM for storing the application data.

**Table 3-5** On-chip RAM Comparison

Item	PIC16(L)F183xx	RL78/G16
Memory size	PIC16F180xx: 0.25 to 2 KB PIC16(L)F183xx: 0.25 to 2 KB PIC16(L)F184xx: 0.5 to 2 KB	2 KB
Parity check	N/A	Available
ECC	N/A	N/A
Write protection (RAM guard)	N/A	Available
Instruction execution	N/A	Available

RL78/G16 embeds a low-power on-chip RAM that can be efficiently accessed within the CPU frequencies supported by the device. The on-chip RAM can be used to store information such as call stack, heap, and global data.

The on-chip RAM can be used as the data area as well as the program area, where instructions can be executed (instructions cannot be executed in the area where the general-purpose register is allocated).

Additionally, RL78/G16's write protection mechanism allows the application to specify the write protected area in the on-chip RAM. With this function, the instruction code or data can be protected from unintentional overwriting, ensuring safer storage of the executable instruction code.

### 3.3 Power-On and Reset Overview and Comparison

PIC16(L)F1 and RL78/G16 have the minimum operating voltage and are equipped with the circuit which allows the device to start up properly by holding the device or a part of the device in the reset state. Table 3-6 shows the comparison of how this function works and which modules control the power-on process and reset.

**Table 3-6** Power-On Overview and Comparison

PIC16(L)F183xx		RL78/G16	
Power-on reset (POR)	When $V_{DD} > V_{POR}$ , the POR state is released. When $V_{DD} < V_{PORR}$ , the POR state is held.	Selectable power-on-reset circuit (SPOR) Note 1	When $V_{DD} > V_{SPOR}$ , the SPOR state is released. When $V_{DD} < V_{SPDR}$ , the SPOR state is held.
Brown-out reset (BOR) Note 1	When $V_{DD} > V_{BOR}$ , the BOR state is released. When $V_{DD} < V_{BOR}$ , the BOR state is held.		
Low-power brown-out reset (LPBOR)	When $V_{DD} > V_{LPBOR}$ , the LPBOR state is released. When $V_{DD} < V_{LPBOR}$ , the LPBOR state is held.		

Note 1 The level of voltage detection can be selected with option bytes.

The relationship between RL78/G16 voltage thresholds is  $V_{SPDR} < V_{SPOR} < \text{lower limit of the operating voltage}$ . The relationship between PIC16(L)F183xx voltage thresholds is  $V_{PORR} < V_{POR} < V_{BOR} < V_{LPBOR}$ .  $V_{LPBOR}$  is the smallest value of  $V_{DD}$  specified to enable the internal circuit's normal operation.

Figure 3-1 shows the block diagram of the RL78/G16 reset function. The following 7 events can generate a reset signal in RL78/G16.

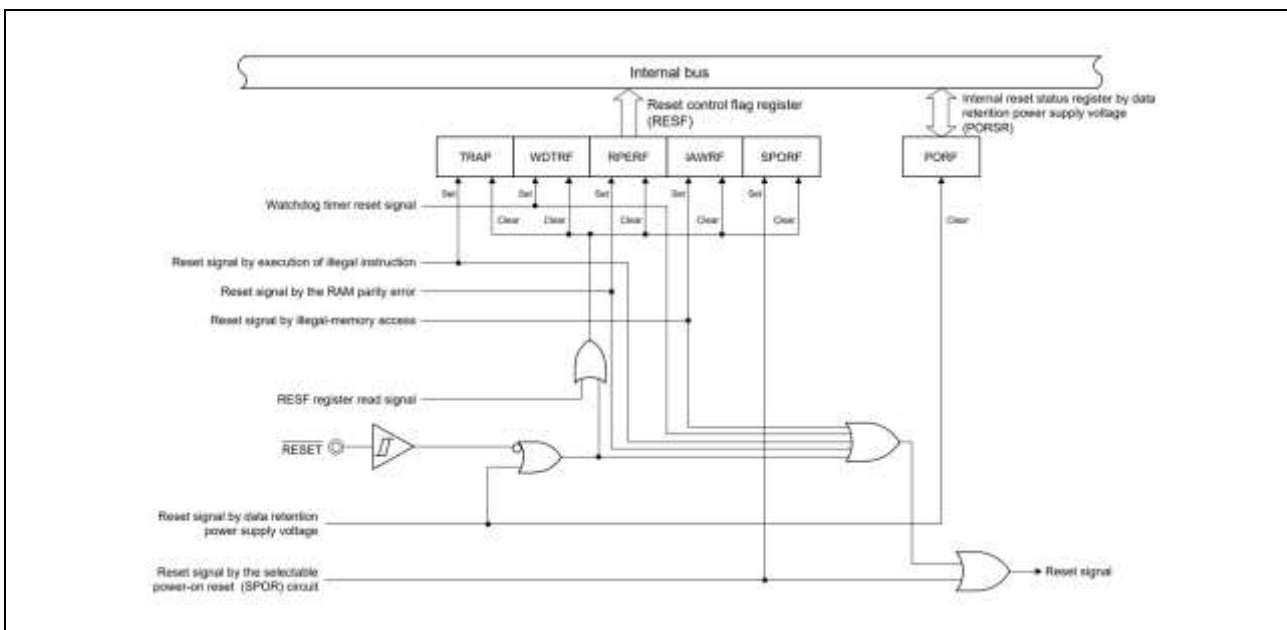
- (1) External reset input with the  $\overline{\text{RESET}}$  pin
- (2) Internal reset due to runaway of the watchdog timer
- (3) Internal reset due to comparison of power supply voltage and detection voltage in the selectable power-on reset circuit (SPOR)
- (4) Internal reset due to execution of an illegal instruction <sup>Note</sup>
- (5) Internal reset due to the data retention supply voltage
- (6) Internal reset due to an illegal memory access
- (7) Internal reset due to a RAM parity error

There are many events that trigger an internal reset in RL78/G16. Use the reset control register (RESF) to check which event triggered the reset request.

**Table 3-7** RESF Register Status when Reset is Requested

Reset Event / Flag	RESET Input	Execution of Illegal Instruction	WDT	RAM Parity Error	Illegal Memory Access	SPOR	Low Limit of Data Retention Supply Voltage
TRAP	Clear (0)	Set (1)	Hold	Hold	Hold	Hold	Clear (0)
WDTRF		Hold	Set (1)	Hold	Hold	Hold	
RPERF		Hold	Hold	Set (1)	Hold	Hold	
IAWRF		Hold	Hold	Hold	Set (1)	Hold	
SPORF		Hold	Hold	Hold	Hold	Set (1)	

**Note** An internal reset occurs when the instruction code “FFH” is executed.  
 A reset due to the execution of an illegal instruction does not occur during emulation using the on-chip debug emulator.



**Figure 3-1** Block Diagram of Reset Function

### 3.4 Clock Overview and Comparison

#### 3.4.1 Oscillator

RL78/G16 and PIC16(L)F1 are equipped with many types of internal and external clock generation circuits to reduce system cost and power consumption. Table 3-8 shows the comparison of the clock circuits in RL78/G16 and PIC16(L)F183xx.

**Table 3-8** Oscillator Comparison

Type	PIC16(L)F183xx	RL78/G16
Internal oscillator	1/2/4/8/12/16/32 MHz high-speed on-chip oscillator ( $F_{HFOSC}$ ) Max 32 MHz output via PLL (×2)	1/2/4/8/16 MHz (typ.) high-speed on-chip oscillator ( $f_{IH}$ )
	31 kHz low-speed oscillator ( $F_{LFOSC}$ )	15kHz (typ.) low-speed on-chip oscillator ( $f_{IL}$ )
External oscillator	External oscillator LP mode: up to 100 kHz ( $F_{LP}$ ) XT mode: up to 4 MHz ( $F_{XT}$ ) HS mode: up to 20 MHz ( $F_{HS}$ ) Max 32 MHz output via PLL (×4)	1 to 12 MHz X1 oscillator ( $f_X$ )
	32.768 kHz secondary oscillator ( $F_{T1}$ )	32.768 kHz of XT1 oscillator ( $f_{XT}$ )

##### 3.4.1.1 RL78/G16 Oscillator

RL78/G16 is equipped with an on-chip oscillator that eliminates the need for an external resonator, as well as many types of oscillator circuits, including a low-speed oscillator circuit that achieves low-power consumption. Table 3-9 shows RL78/G16's clock generation circuits. Refer to the User's Manual: Hardware for details.

**Table 3-9** RL78/G16 Oscillators

Clock Generation Circuit	Clock Signal	Description
Internal	$f_{IH}$	High-speed on-chip oscillator clock (16 MHz (max.))
	$f_{IL}$	Low-speed on-chip oscillator clock (15 kHz (typ.))
External	$f_X$	X1 clock oscillator (12 MHz (max.))
	$f_{XT}$	XT1 clock oscillator (32.768 kHz (typ.))

### 3.4.2 Clock Signal Comparison

Different clock signals can be divided to provide as other input clocks or distributed to many peripheral devices.

**Table 3-10** Clock Signal Comparison

Clock Description		PIC16(L)F183xx Clocks	RL78/G16 Clocks
External clock input	High-speed	OSC1	EXCLK ( $f_{EX}$ )
	Low-speed	SOSCI	EXCLKS ( $f_{EXS}$ )
High-speed system clock		EXTOSC circuit output	$f_{MX}$ <sup>Note 1</sup>
Sub system clock		SOSC circuit output	$f_{SUB}$ <sup>Note 2</sup>
Main system clock		Clock set in the OSCCON2 Register	$f_{MAIN}$ <sup>Note 3</sup>
CPU/Peripheral hardware clock		System clock Peripheral clock	$f_{CLK}$

Note 1  $f_{MX}$  is the high-speed system clock that is supplied from the high-speed external oscillator ( $f_x$ ) or high-speed external clock input ( $f_{EX}$ ).

Note 2  $f_{SUB}$  is the sub system clock that is supplied from the low-speed external oscillator ( $f_{XT}$ ) or low-speed external clock input ( $f_{EXS}$ ).

Note 3 RL78/G16's main system clock is supplied from  $f_{MX}$  or  $f_{IH}$ .

### 3.5 Overview and Comparison of Operation Modes

RL78/G16 provides three main operation modes (power mode) to optimize device power consumption based on the application requirements. In descending order of power, the modes are normal operation, HALT, and STOP. In the normal operation mode, the CPU is active and executes code. Unmasked peripheral interrupt requests can wake up the device from HALT or STOP mode to normal operation mode.

In HALT mode, all peripheral devices are active while the CPU stops. In the STOP mode, the high-speed system clock oscillator circuit and high-speed on-chip oscillator are stopped, and the entire system stops. STOP mode can significantly reduce the CPU operating current.

#### 3.5.1 Comparison of Operation Modes

Table 3-11 shows the comparison of operation modes in RL78/G16 and PIC16(L)F183xx.

**Table 3-11** Comparison of Operation Modes

PIC16(L)F183xx		RL78/G16	
Operation Mode	Description	Operation Mode	Description
RUN	CPU operates with clock output from internal and external oscillators. All peripherals are active.	Normal operation	CPU operates with the main system clock or the sub system clock. All peripherals are active.
DOZE	The CPU operation and program memory access are reduced without affecting peripheral operations (wait time is inserted per one instruction processing). Internal and external oscillators keep operating.	HALT	CPU stops operating. The main system clock and the sub system clock keep operating. All peripherals are active.
IDLE	CPU completely stops while peripherals keep operating; the CPU clock is completely stopped.		
SLEEP	CPU stops operating. The peripherals that use the internal oscillator or the secondary oscillator as the clock source can operate.	STOP	CPU and the main system clock stop operating. The sub system clock holds the state it was in before STOP mode is entered. The entire system stops except for the peripherals that operate with the sub system clock.

### **3.5.2 RL78/G16 Functions in HALT Mode**

In HALT mode, the clock supply to the CPU is stopped but the peripherals continue to operate, which reduces power consumption in comparison to normal operation.

HALT mode can be released automatically with an interruption, allowing normal operations to resume immediately.

This mode is suitable for applications that require low power consumption and quick response.

### **3.5.3 RL78/G16 Functions in STOP Mode**

In STOP mode, the main clock, the high-speed on-chip oscillator, and the low-speed on-chip oscillator stop, allowing power consumption to be reduced to almost none.

The contents of the memories and the registers are held even when the CPU completely stops, so the processing can be continued after returning to normal mode. When any specific interruption occurs, such as an external interrupt, the watchdog timer (WDT), or the RTC interrupt trigger, the CPU returns to normal mode from STOP mode.

Although returning to normal mode takes longer than usual since the clock is stopped and needs to start oscillating, this mode is suitable for applications requiring minimum power consumption.

For implementation detail, and support and descriptions of the peripherals, refer to the appropriate chapters of the RL78/G16 User's Manual: Hardware.

### 3.6 Comparison of Interrupts and Events

#### 3.6.1 Interrupts and Exceptions

Both RL78/G16 and PIC16(L)F1 interrupt, register exception vectors, and map according to the available peripheral functions of the device. Table 3-12 shows an overview and comparison of interrupt vectors for each device family. An interruption or exception with higher priority takes precedence over one with lower priority; the highest priority is indicated by 1 and the lowest priority by 5. If an interrupt request with a priority higher than that of the interrupt currently being serviced is generated, it is acknowledged for multiple interrupt servicing.

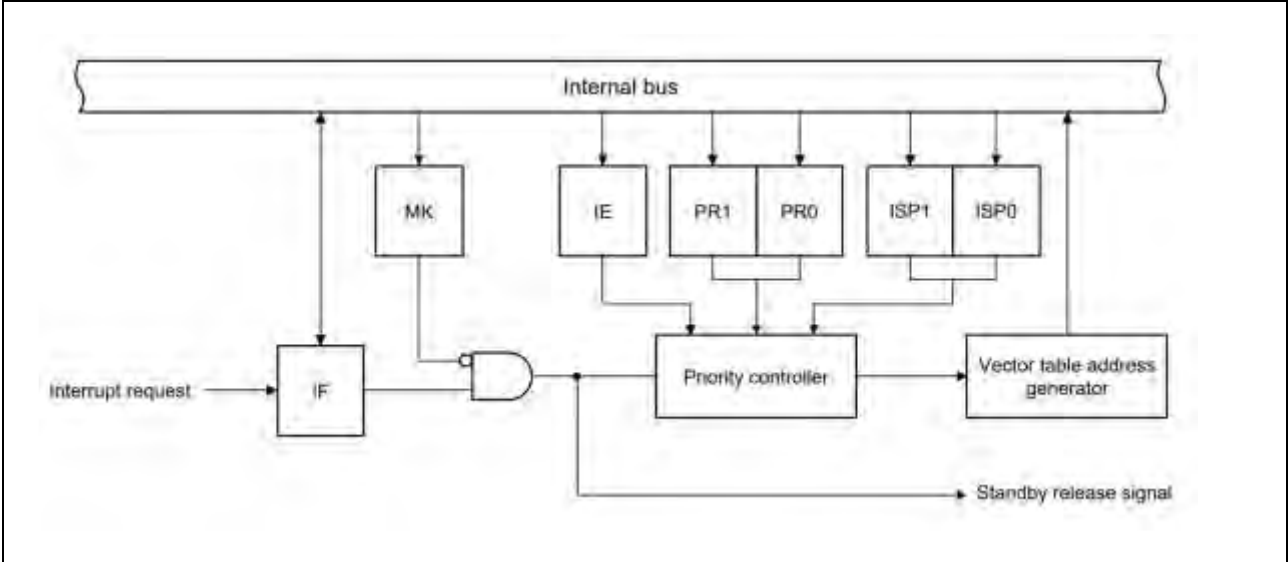
**Table 3-12** Comparison of Interrupts

Peripheral Function	PIC16(L)F183xx	RL78/G16
Interrupt types	Peripheral interrupts Internal interrupt request x 33 (max.) External interrupt request x 2	Maskable interrupts Internal interrupt request x 30 (max.) External interrupt request x 10 (max.)
	Reset Reset factors: MCLR pin input, watchdog timer, power-on reset, brown-out reset, reset instruction, stack overflow, programming mode exit	Reset Reset factors: RESET pin input, watchdog timer, selectable power-on reset, illegal instruction, data retention supply voltage, illegal memory access, RAM parity error  Software interrupts A software interrupt occurs when the BRK instruction is executed. Software interrupts cannot be disabled.
Priorities	N/A The software determines an interrupt request and controls the priority of the interrupt processing.	Default priority The default priority is used when multiple maskable interrupts with the same interrupt priority occur.
		Interrupt priority level The programmable priority level can be set between 0 to 3 for maskable interrupts.
Priority setting	No register available Priorities are controlled by the software.	PR0xy and PR1xy Registers PR0xy and PR1xy are used to set the maskable interrupt priority level.
Interrupt mask	PIEx The PIEx register is used to enable/disable corresponding peripheral interrupts.	MKxy The MKxy register is used to enable/disable corresponding maskable interrupts.

**3.6.1.1 Interrupt Control in RL78/G16**

For RL78/G16, the priority level for each interrupt condition is set with registers PR0xy and PR1xy, and the interrupt condition is enabled/disabled with the MKxy register.

Figure 3-2 shows the configuration of RL78/G16 internal maskable interrupts. Interrupts need to be enabled by setting IE to “1” through the EI instruction before each interrupt is detected.



**Figure 3-2** Configuration of RL78/G16 Internal Maskable Interrupts

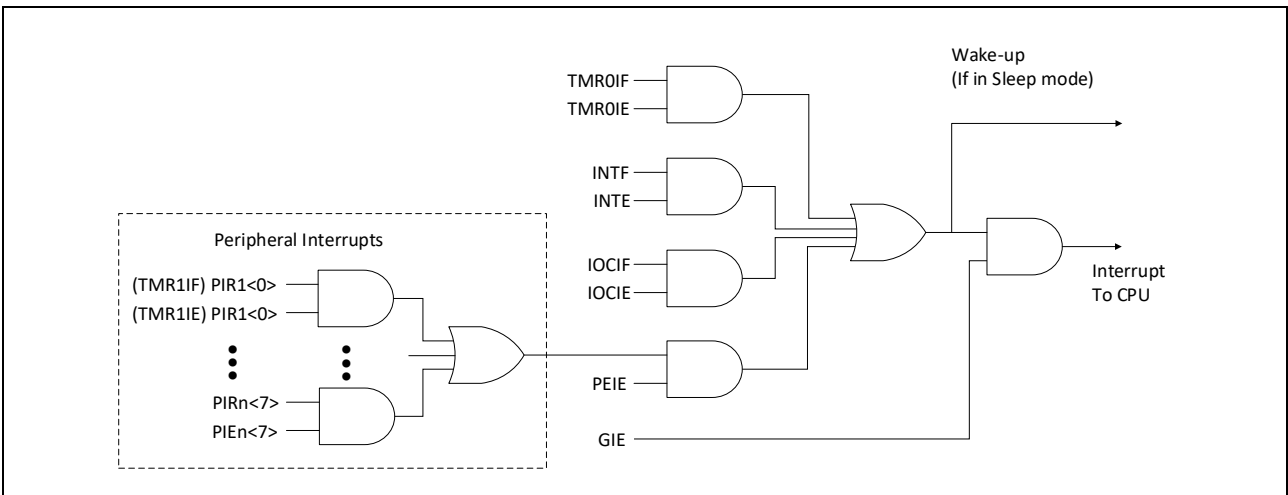
**3.6.1.2 Interrupt Control in PIC16(L)F183xx**

Figure 3-3 shows the configuration of PIC16(L)F183xx peripheral interrupts.

For PIC16(L)F183xx, the interrupt condition is enabled/disabled with the PIEx register. Interrupts need to be enabled by setting GIE to “1” in the INTCON register before each interrupt request flag is set.

Unlike RL78/G16, priorities are not handled by the device but are rather by the software that executes each interruption.

The software that executes an interrupt must determine the interrupt source by polling the interrupt request flag. Also, to avoid continued repetition of an interrupt processing, the interrupt request flag must be cleared before completing the interrupt processing.



**Figure 3-3** PIC16(L)F183xx Peripheral Interrupt Hierarchy<sup>[2]</sup>

### 3.7 Debugging and Programming Comparison

PIC16(L)F1 and RL78/G16 use different interfaces and functions when connecting the device for debugging or programming. Table 3-13 shows the comparison of debugging and programming in PIC16(L)F183xx and RL78/G16.

**Table 3-13** Debugging and Programming Comparison

Programming Function	PIC16(L)F183xx	RL78/G16
Type	Serial programming	Serial programming
Security	Code Protection Write Protection	Read protection from a debugger
Entry to the programming mode	High-Voltage Programming Entry Mode Low-Voltage Programming Entry Mode	After reset is released, hold the TOOL0 pin at low for 1ms, then set the pin to high and send 1-byte of the mode setting data within 100ms.
Command	Configuration, Address Setting, Erase Write, Read	Mode setting, CRC, programming after erase (status code can be received)
Interface	ICSP, High-speed/LVDS communication	Single-line UART, dedicated UART

#### 3.7.1 Debugging Comparison

##### 3.7.1.1 Debugging RL78/G16

E2 emulator or E2 emulator Lite is used for debugging RL78/G16 MCUs. These debuggers support standard debugging functions (step execution, breakpoint, memory/register check, etc.). For RL78/G16 programming, E2 emulator, E2 emulator Lite, or dedicated flash memory programmer PG-FP6 is used.

The on-chip debugging function is also supported, enabling debugging through a single-wire/two-wire serial interface.

##### 3.7.1.2 Debugging PIC16(L)F183xx

PIC16(L)F183xx is debugged using the MPLAB® ICE 4 in-circuit emulator, the MPLAB® ICD 5 in-circuit debugger, or the MPLAB® PICkit™ 5. PIC16(L)F183xx debugging and programming can be performed through the MPLAB IDE graphical interface. Connect these debug systems to the PC using the USB interface and to the target using a compatible connector. The MPLAB® ICE 4 in-circuit emulator has functions such as full-speed emulation, runtime variable watch, trace analyzation, and complicated breakpoint.

### 3.7.2 Programming Mode Comparison

#### 3.7.2.1 Serial Programming in RL78/G16

The RL78/G16 serial programming mode enables firmware updates using an external device (microcontroller or ASIC) connected to RL78/G16 and UART. Writing data to or deleting data from RL78/G16 using an external device is performed on-board. Off-board writing is not supported.

#### 3.7.2.2 Serial Programming in PIC16(L)F183xx

The in-circuit serial programming (ICSP) interface provides data for non-volatile memory (NVM) programming, enabling the programming of flash memory or EEPROM.

The following five pins are used for ICSP programming:

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VD
- VSS

In program/verify mode, program memory, EEPROM, user ID and configuration words are programmed via serial communication. The ICSPDAT pin is the bidirectional I/O used for serial data transfer and the ICSPCLK pin is the clock input pin.

Connection to the target device is typically made through an ICSP header. The connector commonly found on development tools is RJ-11 in a 6P6C (6 pin, 6 connector) configuration.

## 4. Digital Peripheral Comparison

### 4.1 General Purpose I/O (PORT, PIOR)

RL78/G16 is designed specifically for low-power consumption including I/O ports.

Also, RL78/G16 is equipped with a peripheral I/O redirect function that allows for switching the ports assigned to alternate functions. The alternate function pins pre-mapped by RL78/G16 can be switched to other dedicated functions by changing the setting value of the peripheral I/O redirection register (PIORx).

PIC16(L)F1 provides a Peripheral Pin Select (PPS) function. The PPS function allows you to assign a port only once or change the pin assignment dynamically during code execution.

**Table 4-1** I/O Port Function Comparison

Item	PIC16(L)F183xx	RL78/G16
Output mode	CMOS N-ch open drain	CMOS N-ch open drain
Input mode	CMOS or TTL On-chip pull-up	CMOS On-chip pull-up
Slew rate selection	Selectable with the SLRCONx register	N/A
Setting per port	Available	Available
Redirect function	Available with the PPS function	Available with the PIORx register
Wake-up	Available with external interrupts (INT pin or IOC pin)	Available with external interrupts (INTPx)
Input filtering	N/A	Available (filters a signal within 1us when an external interrupt (INTPx) is used)
User controllable input hysteresis	Controllable with the INLVLx register (CMOS or TTL)	N/A

I/O port code example:

For information regarding I/O port code examples, visit the Renesas Electronics website.

## 4.2 Universal Asynchronous Receiver-Transmitter (UART, SAU)

Both RL78/G16 and PIC16(L)F1 are equipped with UART functionality that enables asynchronous serial communication.

RL78/G16 is equipped with a UART function in the Serial Array Unit (SAU). The RL78/G16 UART enables asynchronous communication with two lines: serial data transmission (TxD) and serial data reception (RxD). These two communication lines are used to transmit/receive data to the target side asynchronously (using the internal baud rate). Each data frame consists of a start bit, data, parity bit, and stop bit. Full-duplex UART communication can be achieved by using two channels, the dedicated to transmission (even channel) and the dedicated to reception (odd channel).

**Table 4-2** UART Function Comparison

Item	PIC16(L)F183xx	RL78/G16
Data direction	LSB first	MSB first or LSB first
Data phase	Available	Available
Single-wire half-duplex communication	Available (synchronous)	N/A
Data length	8, 9 bits	7, 8, 9 bits
LIN hardware support	Available	N/A
Wakeup from standby	Available	Available
Auto baud rate detection	Available	N/A
Tx/Rx FIFO Depth	1	1

UART code example:

For information regarding UART code examples, visit the Renesas Electronics website.

### 4.3 Serial Peripheral Interface (SPI, SAU)

Both RL78/G16 and PIC16(L)F1 are equipped with a serial peripheral interface (SPI).

RL78/G16 is equipped with a simple SPI in the serial array unit (SAU). The RL78/G16 simple SPI transmits/receives data in synchronization with the serial clock (SCK) output from the master. The clock synchronous communication function achieves communication using three lines: 1 line for serial clock (SCK), and 2 lines for transmitting/receiving serial data (SO, SI).

**Table 4-3** SPI Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Control pins	SCK, SDO, SDI, SS	SCK, SI, SO
Master/Slave selection	Available	Available
Data bit width	8 bits	7, 8 bits
Maximum speed	Master communication: 8 MHz ( $f_{CLK}/4$ ) Slave communication: 4 MHz ( $f_{OSC}/8$ )	Master communication: 4 MHz ( $f_{CLK}/4$ ) Slave communication: 2.67 MHz ( $f_{MCK}/6$ )
Simplex communication	Available	Available
Chip select management	Available (by Hardware)	Available (by Software)
I/O clock phase control	Available	Available
Data direction	MSB first	MSB first or LSB first
SPI format support	Motorola	Motorola
Tx/Rx FIFO Depth	1	1

SPI code example:

For information regarding the SPI code examples, visit the Renesas Electronics website.

#### 4.4 Inter-Integrated Circuit (IICA, SAU)

Both RL78/G16 and PIC16(L)F1 are equipped with inter-integrated circuit (I<sup>2</sup>C) functionality.

The RL78/G16 is equipped with a SAU (Serial Array Unit) that has simple I<sup>2</sup>C functions, and an IICA that has more advanced functions. Clock synchronous communication can be carried out for multiple devices using both serial clock (SCL) and serial data (SDA) communication lines. However, the simple I<sup>2</sup>C functions as master only, since its purpose is single communication with devices such as flash memory and A/D converter.

**Table 4-4** I<sup>2</sup>C Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Master/Slave selection	Available	Available (IICA)
Multi-master support	Available	Available (IICA)
Maximum transfer speed	400 kHz	400 kHz
Addressing mode	7, 10 bits	7 bits
Clock stretch	Available	Available (IICA)
Wakeup function	Available	Available (IICA)
Software reset	Available	Available (IICA)
FIFO/Buffer	1 byte SSPxSR	1 byte IICAn (IICA) SDRmn (SAU)
Programmable noise filter	N/A	Available (IICA)

I<sup>2</sup>C code example:

For information regarding I<sup>2</sup>C code examples, visit the Renesas Electronics website.

### 4.5 Timer (TAU, 12-Bit Interval Timer)

Both RL78/G16 and PIC16(L)F1 are equipped with multi-function timers.

RL78/G16 is equipped with the timer array unit (TAU) comprising eight 16-bit timers. Each timer can be used independently, or multiple timers can be combined to use as an advanced timer.

Additionally, advanced timers such as Timer KB, Timer RD, and Timer RG, which facilitate forced PWM output shutdown and high-resolution three-phase complementary PWM output, are integrated into the RL78/G14, G1F, and G24.

**Table 4-5** Timer Names

PIC16(L)F183xx		RL78/G16	
Timer Name	Abbr.	Timer Name	Abbr.
Timer0 module	Timer0	Timer array unit	TAU
Timer1/3/5 module with gate control	Timer1/3/5	12-bit interval timer	-
Timer2/4/6 module	Timer2/4/6		

**Table 4-6** Timer Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Resolution	8, 16 bits	8, 16 bits
PWM	Available <sup>Note</sup>	Available
Capture	Available <sup>Note</sup>	Available
Compare	Available <sup>Note</sup>	Available
One-shot	Available <sup>Note</sup>	Available
Up/down count function	N/A	N/A
Conjunctive operation with standby function	Available	Available
Programmable prescaler clock frequency	Available	Available
Event/interrupt	Available	Available
Auto reload function	Available	Available

Note: The function operates in conjunction with the CCP module.

**Table 4-7** Timer Use Case Comparison

Timer Use	PIC16(L)F183xx	RL78/G16
PWM	Timer2/4/6 <sup>Note</sup>	TAU
Capture	Timer1/3/5 <sup>Note</sup>	TAU
Compare	Timer1/3/5 <sup>Note</sup>	TAU
One-shot	Timer1/3/5 <sup>Note</sup>	TAU
Synchronization	-	TAU
Interval timer	Timer0 Timer1/3/5 <sup>Note</sup>	TAU 12-bit interval timer
Prescaler	4-bit prescaler	4-bit prescaler (TAU)

Note: The function operates in conjunction with the CCP module.

Timer code example:

For information regarding timer code examples, visit the Renesas Electronics website.

## 4.6 Watchdog Timer (WDT)

Both RL78/G16 and PIC16(L)F1 are equipped with a watchdog timer (WDT).

RL78/G16 is equipped with a watchdog timer that uses a low-speed on-chip oscillator to detect program runaway. An internal reset signal is generated when the timer detects a program run away.

**Table 4-8** WDT Name

PIC16(L)F183xx		RL78/G16	
Timer Name	Abbr.	Timer Name	Abbr.
Watchdog timer	WDT	Watchdog timer	WDT

**Table 4-9** WDT Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Window mode	N/A	N/A
Interval timer mode	Available	Available
Clock source	LFINTOSC	f <sub>IL</sub>
Interrupt	N/A	Available
Counter resolution	23 bits	17 bits
Clock divider	N/A	N/A

WDT code example:

For information regarding WDT code examples, visit the Renesas Electronics website.

## 5. Analog Peripheral Comparison

### 5.1 Analog Digital Converter (ADC)

Both RL78/G16 and PIC16(L)F1 are equipped with an A/D converter that converts analog signals to digital signals.

The RL78/G16 ADC configuration enables control of up to 11 analog input channels, allowing selection of internal reference voltage, temperature sensor output voltage, and TSCAP voltage for the capacitive touch sensor. The ADC resolution can be set to either 10-bit and 8-bit.

**Table 5-1** ADC Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Resolution (bit)	10 bits	12, 10 bits
Conversion rate (Msps)	16	0.235
FIFO	N/A	N/A
ADC reference (V)	Internal: $V_{DD}$ , FVR <sup>Note 2</sup>	Internal: $V_{DD}$
	External: $V_{REF+}$	External: —
Operation mode	Normal operation, DOZE, IDLE, SLEEP	Normal operation, HALT
Auto power down	N/A	N/A
External input channel <sup>Note 1</sup>	Up to 17	Up to 11
Internal input channel	Temperature sensor output voltage, DAC output voltage, FVR <sup>Note 2</sup>	Temperature sensor output voltage, internal reference voltage, touch sensor capacitance voltage
Number of ADCs	1	1

Note 1: The number of external input channels differs depending on the device.

Note 2: Fixed Voltage Reference

ADC code example:

For information regarding ADC code examples, visit the Renesas Electronics website.

## 5.2 Comparator (COMP)

Both RL78/G16 and PIC16(L)F1 are equipped with comparators.

RL78/G16 is equipped with comparators that provide basic functions. The comparators allow selection of response speed and external/internal reference voltage, and generate an interruption when detecting a valid edge of the comparator's output.

**Table 5-2** Comparator Function Comparison

Function	PIC16(L)F183xx	RL78/G16
Available comparators	Up to 2	Up to 2
Comparison results output pin	CxOUT	VCOUTx
+ input	CxIN0+	External analog pin (IVCMPx)
	DAC output	
	FVR <sup>Note 1</sup>	
	V <sub>SS</sub> (ground)	
- input	CxINy- pin	External analog pin (IVREFx)
	FVR <sup>Note 1</sup>	Internal reference voltage 0.815V (typ.)
	Analog ground	
Programmable hysteresis	Available	N/A
Operation mode	Normal operation, DOZE, IDLE, SLEEP	Normal operation, HALT, STOP
Output filtering	N/A	Digital filter <sup>Note 2</sup> (3 samplings)
Output polarity control	Available	Available
Interrupt	Rising edge	Rising edge
	Falling edge	Falling edge
	Both edge	Both edge

Note 1: Fixed Voltage Reference

Note 2: When using a comparator during the STOP mode, set it to disable the digital filter.

CMP code example:

For information regarding CMP code examples, visit the Renesas Electronics website.

## 5.3 Internal Reference Voltage (AV<sub>REFP</sub>)

Both RL78/G16 and PIC16(L)F1 are equipped with internal reference voltage that can provide reference voltage to the internal peripherals.

**Table 5-3** Internal Reference Voltage Comparison

Function	PIC16(L)F183xx	RL78/G16
Internal reference voltage (V)	1.024, 2.048, 4.096	0.815
Internal reference output	N/A	N/A
Internal connection to ADC	Available	Available
Internal connection to COMP	Available	Available

## Reference

[1] Microchip, PIC16F18056/76 datasheet, DS40002325C

[2] Microchip, PIC16(F)F18326/18346 datasheet, DS40001839E

[3] Microchip, PIC16(L)F18426/46 datasheet, DS40001985C

**Revision History**

Rev.	Date	Description	
		Page	Summary
1.00	Dec.01.2025	-	First edition issued
1.10	Feb.26.2026	P4, P10, P43, P55	Modified format and contents of Table 1-1, Table 2-3, Table 3-6, and Table4-3

## 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](http://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/](http://www.renesas.com/contact/).