

RL78 Family

Arduino API Introduction Guide

Introduction

This application note explains the method for constructing an environment necessary for writing programs in an Arduino language or the like, taking as an example a case in which the RL78/14 Fast Prototyping Board (FPB) is used.

Target Device

RL78 Products

RL78/G14 Fast Prototyping Board

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

Contents

1.	RL78 Family Development Environment.....	3
1.1	Program execution environment	3
1.2	Installation of an integrated development environment.....	3
2.	Arduino Language Program in the Integrated Development Environment.....	4
2.1	Program execution environment	4
2.2	Project folder configuration.....	4
3.	Preparations for Project Startup.....	5
3.1	When using the CS+ IDE	5
3.2	When using the e2studio IDE.....	9
3.3	When using the IAR IDE	18
4.	Related Application Notes.....	22
	Revision History	23

1. RL78 Family Development Environment

There are three main integrated development environments specific to the RL78 family used with the FPB, which are CS+, e2studio, and IAR. Compilers that can be used include CC-RL, IAR, and GCC.

Here, the following integrated development environments are addressed.

- CS+ (CC-RL) integrated development environment
- e2studio (CC-RL) integrated development environment
- IAR integrated development environment

1.1 Program execution environment

In this application note, a program written in an Arduino language is executed in a development environment specific to the RL78 family. A conceptual diagram of the program execution environment is shown in Figure 1.1.

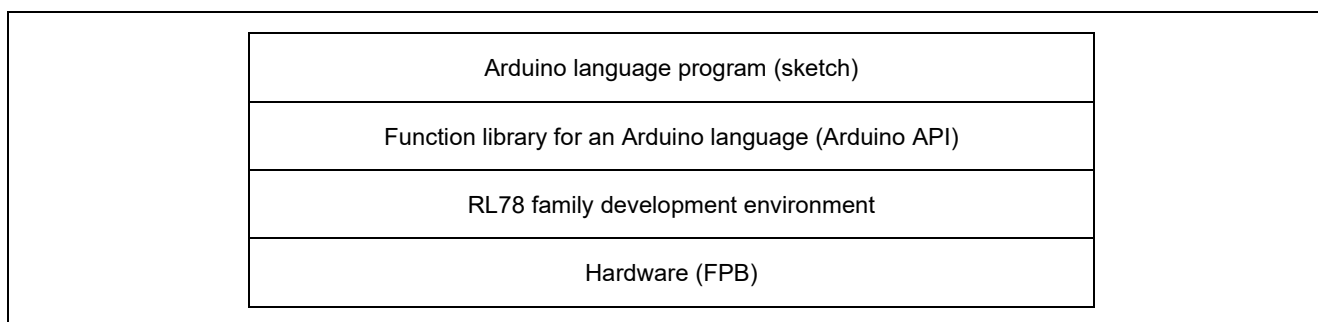


Figure 1.1 Program execution environment

1.2 Installation of an integrated development environment

In order to use the Arduino API, the RL78 family integrated development environment (IDE) must first be installed.

CS+ (CC-RL) and e2studio (CC-RL) can be obtained from the Renesas website; a My Renesas account is required. An account can be created free of charge.

For the installation procedure, please refer to the following manuals.

CS+ V8.02.00 Integrated Development Environment User's Manual: Installer (R20UT4525)

e2 studio Integrated Development Environment User's Manual: Getting Started Guide (R20UT4374)
Chapter 2

For the IAR environment, please refer to the website and manual provided by IAR Systems.

2. Arduino Language Program in the Integrated Development Environment

In this project that uses the FPB, a program (sketch) in an Arduino language consists only of the `setup()` function and `loop()` function in `AR_SKETCH.c`.

2.1 Program execution environment

In this application note, at the time the integrated development environment is started, the `loop()` function that begins the Arduino language program can be verified.

Ordinarily, a project in an RL78 family integrated development environment begins with a `main()` function. In a project that uses the Arduino API, a `main()` function or other functions execute the processing necessary to run an Arduino language program (sketch).

2.2 Project folder configuration

The configuration of folders in the project is shown in Figure 2.1.

The folder (workspace) in which the project is stored has three subfolders, one for each of the integrated development environments, which are CS+, e2studio, and IAR. However, the e2studio integrated development environment is archived in a zip file.

Files used in the RL78 family development environment are stored in the respective IDE folders.

The `AR_SKETCH.c`, which is the Arduino language program (sketch), is stored in the sketch subfolders. When viewing or modifying the sketch program, this `AR_SKETCH.c` file should be opened.

The Arduino API is stored in the `AR_LIB` subfolders.

Files used in the integrated development environments are stored in the other subfolders.

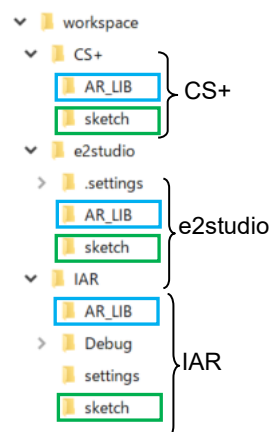


Figure 2.1 Folder configuration

3. Preparations for Project Startup

The procedures for startup preparation in the integrated development environments are as follows.

3.1 When using the CS+ IDE

(1) CS+ preparation tasks

The files in the CS+ folder are shown in Figure 3.1.

Before starting up the project, the name of the mtud file in the folder in which the project is stored (USER_NAME.mtud in project name.PC) must be changed.

In the case of Figure 3.1, "USER_NAME" should be changed to the username of the PC being used. The username can be confirmed in "User Accounts" in the Control Panel.

r01an5384_rl78g14fpb_dio1.USER_NAME.mtud

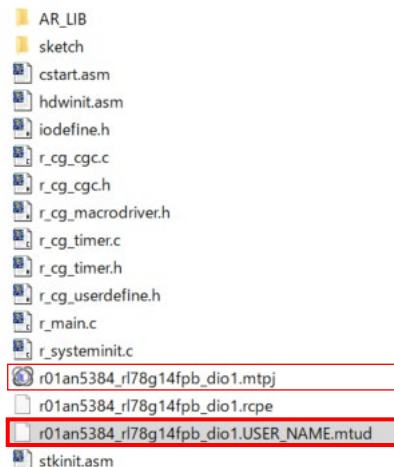


Figure 3.1 File configuration (CS+)

(2) Starting the CS+ Integrated Development Environment

By double-clicking the mtpj file (in the case of Figure 3.1, r01an5384_rl78g14fpb_dio1.mtpj), the CS+ integrated development environment is started.

When the username change in "(1) CS+ preparation tasks" is set correctly, as shown in Figure 3.2, "AR_SKETCH.c", which is the Arduino program (sketch) on project startup, is displayed at the bottom of the project tree, and the contents of "AR_SKETCH.c" are displayed.

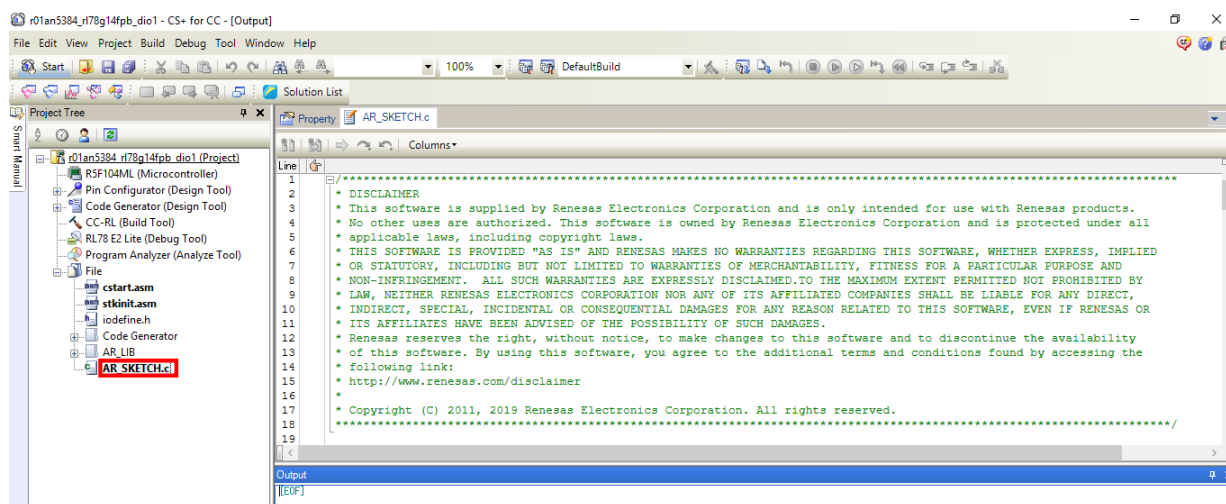


Figure 3.2 Screen after project startup (CS+)

If the contents of "AR_SKETCH.c" are not displayed, click on "AR_SKETCH.c" in the project tree.

(3) Procedure for project execution in CS+

Click on "Build (B)" on the menu bar and select "Build Project (B)" on the pulldown menu. The build is executed.

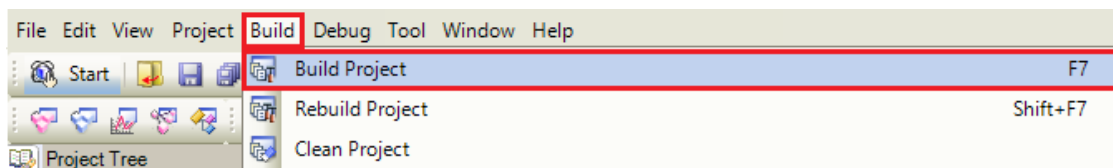


Figure 3.3 Build (CS+)

When the build ends normally, confirm that the PC and the FPB are properly connected via USB.

Next, click on "Debug (D)" on the menu bar, and select "Download to debug tool (D)" on the pulldown menu.

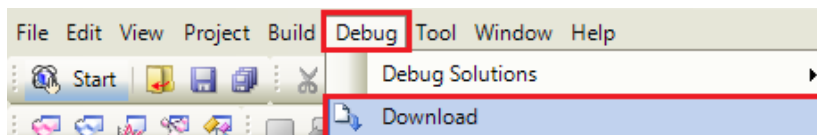


Figure 3.4 Downloading to the FPB (CS+)

If the PC and the FPB are connected via USB, the created (built) program (sketch) is downloaded to the FPB.

When the debugger is started, the first line of the loop function is highlighted, indicating that the program is executed from this point.

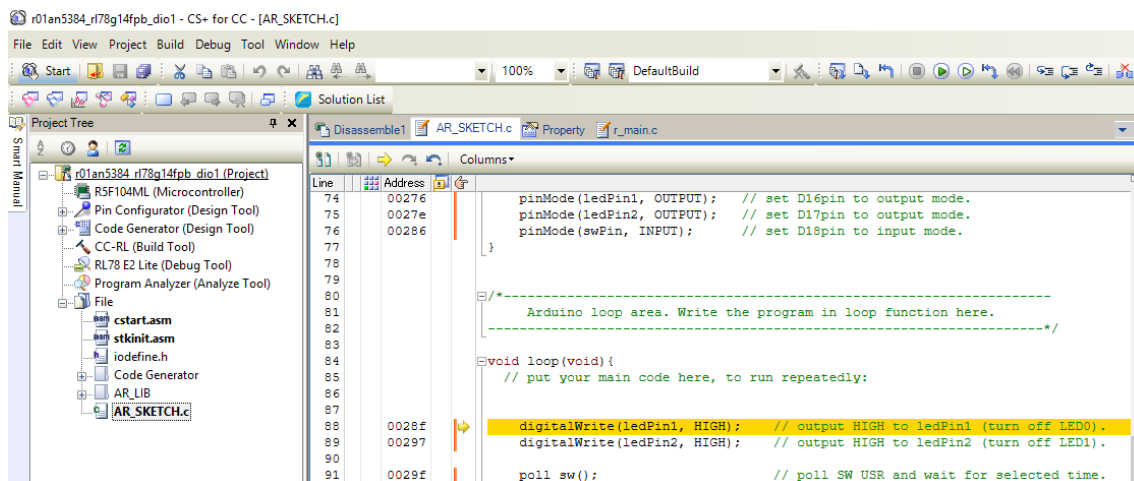


Figure 3.5 Screen after downloading (CS+)

If behavior is not as described above, it is possible that the RL78 family development environment has not been installed correctly. Please start over from installation of the development environment.

If the download ends normally, click on "Debug (D)" on the menu bar, and select "Go (G)" on the pulldown menu.

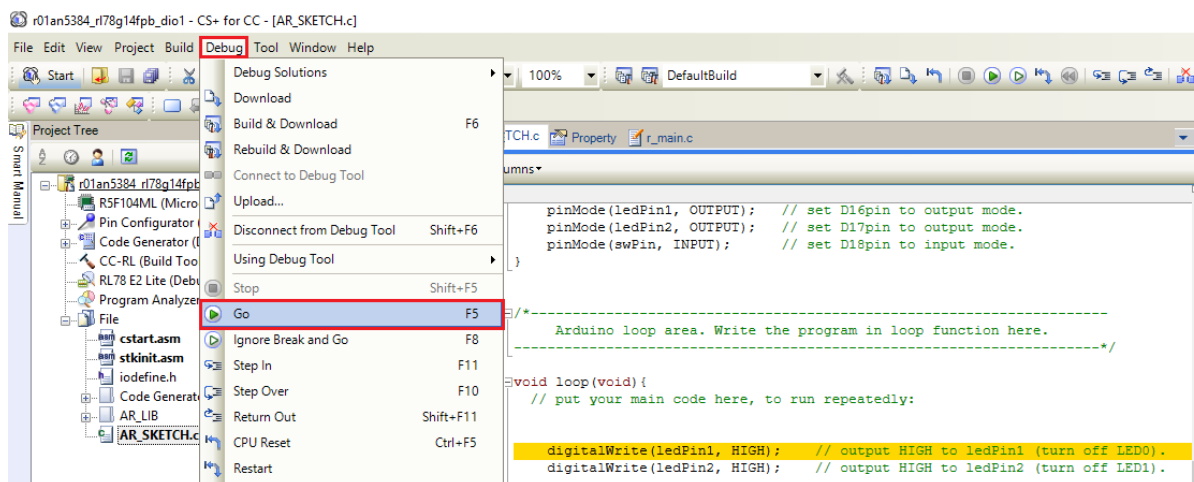


Figure 3.6 Starting execution of program (sketch) (CS+)

For details of debugging procedures using CS+, please refer to the following manual.

CS+ V8.03.00 Integrated Development Environment User Manual RL78 Debugging Tool Edition (R20UT4587), Chapter 2

To stop execution of the program (sketch), click on "Debug (D)" on the menu bar, and select "Stop (S)" on the pulldown menu.

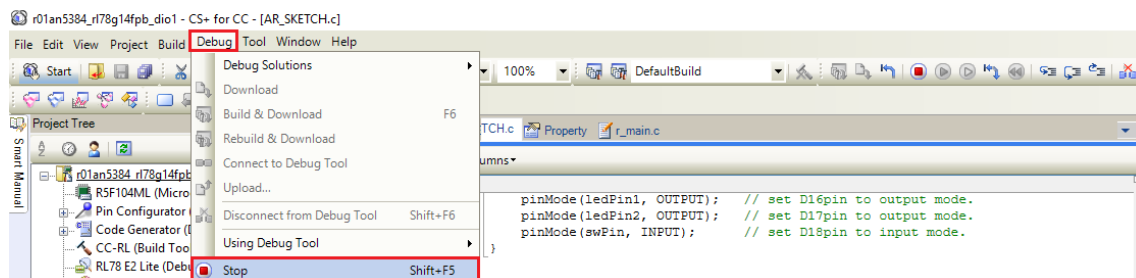


Figure 3.7 Stopping execution of program (sketch) (CS+)

When the program execution stops, a screen like that in Figure 3.8 appears. When program execution stops, the contents of the program executed immediately before stopping are displayed. If there is no need to view these contents, click on "x" on the right side to close the tab.

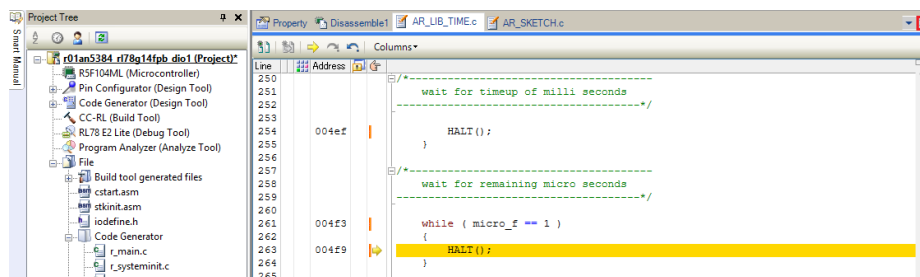


Figure 3.8 Screen example after stopping execution (CS+)

(4) Close the CS+ project

To close the project, with the program (sketch) execution in the stopped state, click on "Debug (D)" on the menu bar and select "Disconnect from Debug Tool (N)" on the pulldown menu.

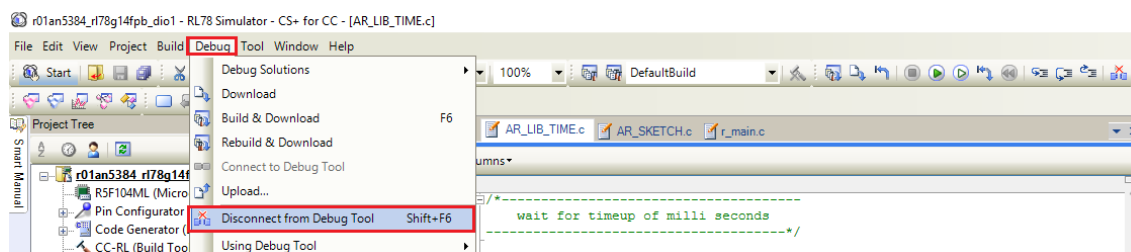


Figure 3.9 Close the project (CS+)

Finally, to close CS+, click on "File (F)" on the menu bar, and select "Exit (X)" on the pulldown menu.

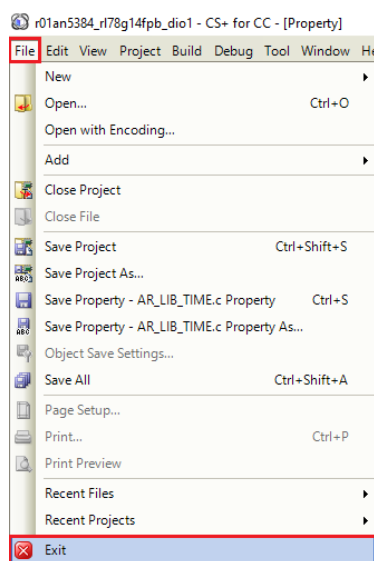


Figure 3.10 Exit CS+

3.2 When using the e2studio IDE

(1) e2studio preparation tasks

Upon starting the Eclipse launcher, the screen for selection of the workspace folder is displayed; click the "Browse (B)" button and specify the folder to use for the e2studio project (in the case of Figure 3.11, e2_work). After specifying the folder, click the "Launch (L)" button to start e2studio.

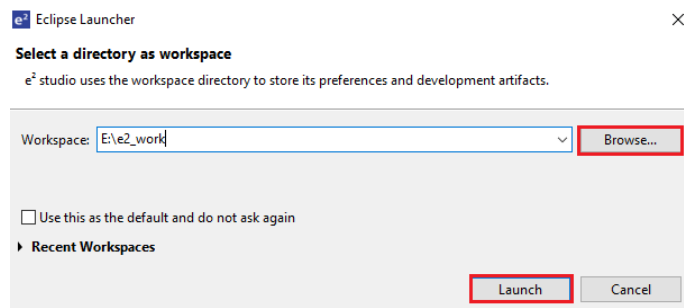


Figure 3.11 Specifying the e2studio work folder

At the "Logging/Reporting" display screen, to permit data collection, check the checkbox and then click the "Apply" button.

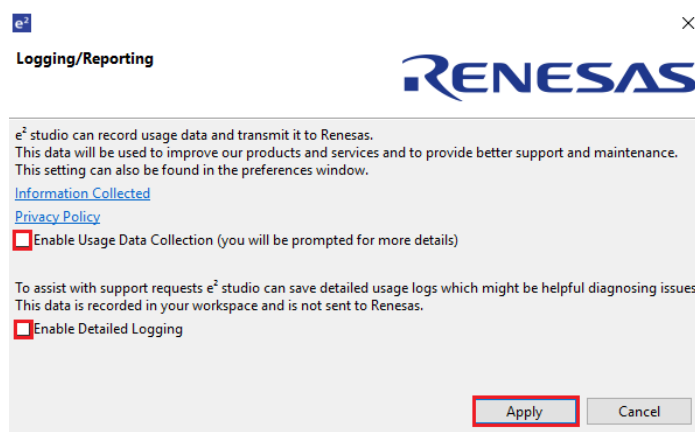


Figure 3.12 Logging/ data collection

Click the right-pointing triangle on the upper right to jump to the "Workbench".

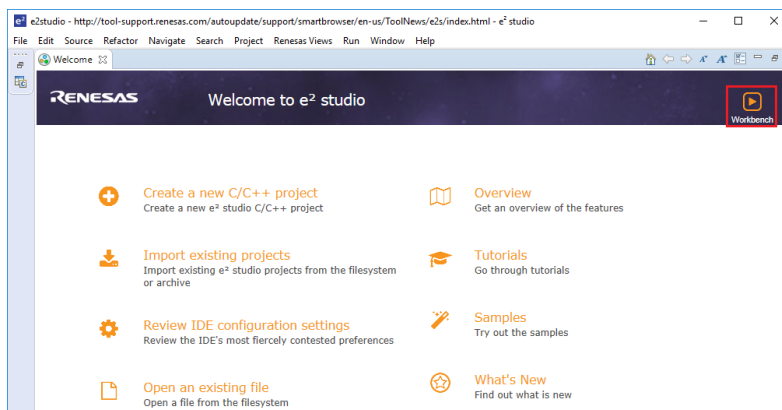


Figure 3.13 "Welcome to e2studio" screen

Click on "x" to close the "Notifications in the Smart Browser".

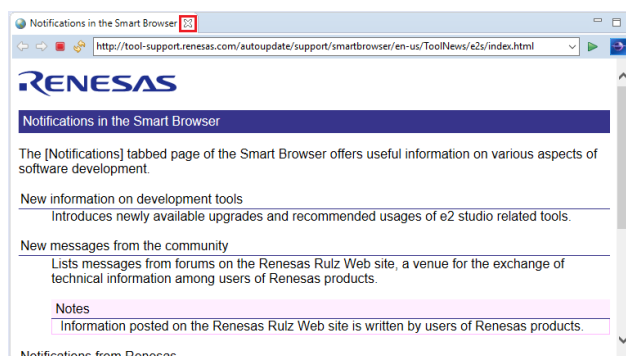


Figure 3.14 "Notifications in the Smart Browser" screen

The regular e2studio screen is displayed.

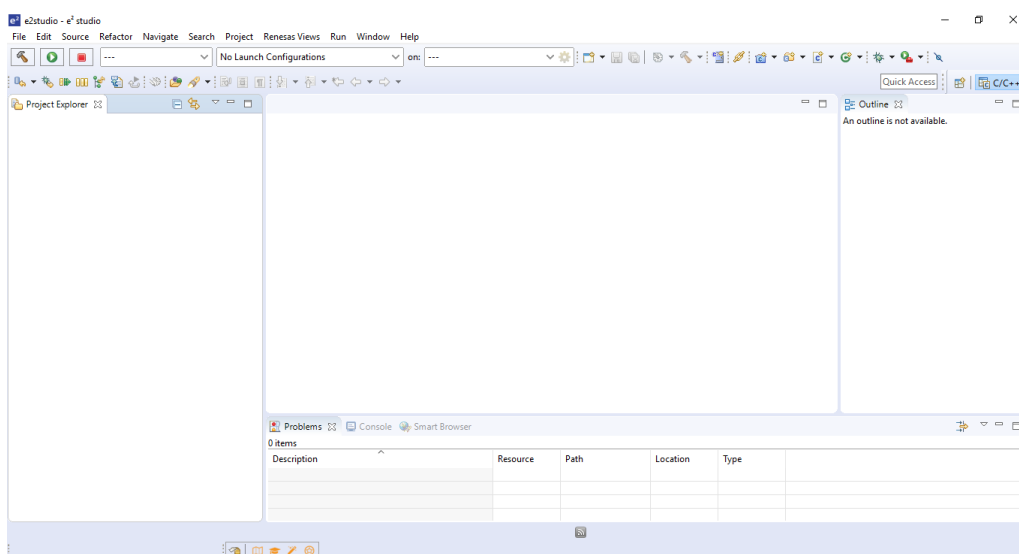


Figure 3.15 Regular e2studio screen

(2) Importing an existing project into e2studio

An existing project (archive file) is imported into the workspace. In this application note, the existing project is archived in the workspace with the name "e2studio.zip".

Click on "File (F)" on the menu bar and click on "Import (I)" on the pulldown menu to open the "Import" screen.

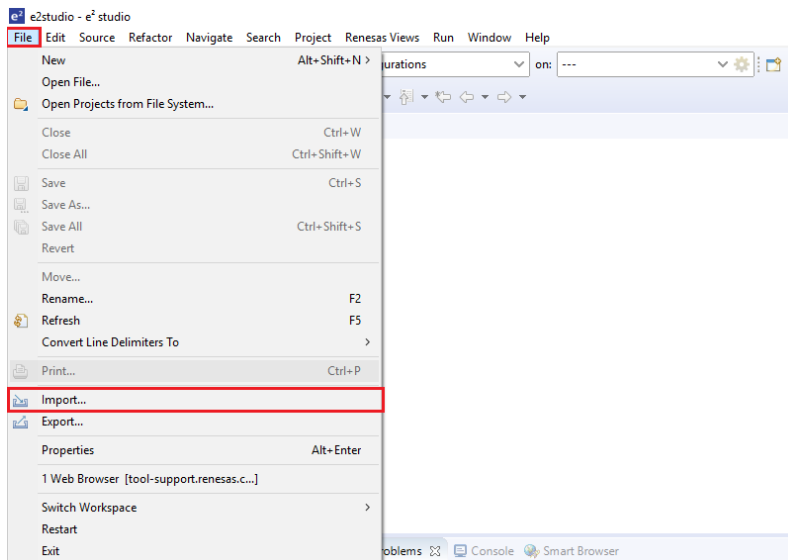


Figure 3.16 Starting the "Import" screen

Click on ">" to the left of "General" under "type filter text" on the "Import" screen to open items in "General". Next, select "Existing Projects into Workspace", and click the "Next (N)" button.

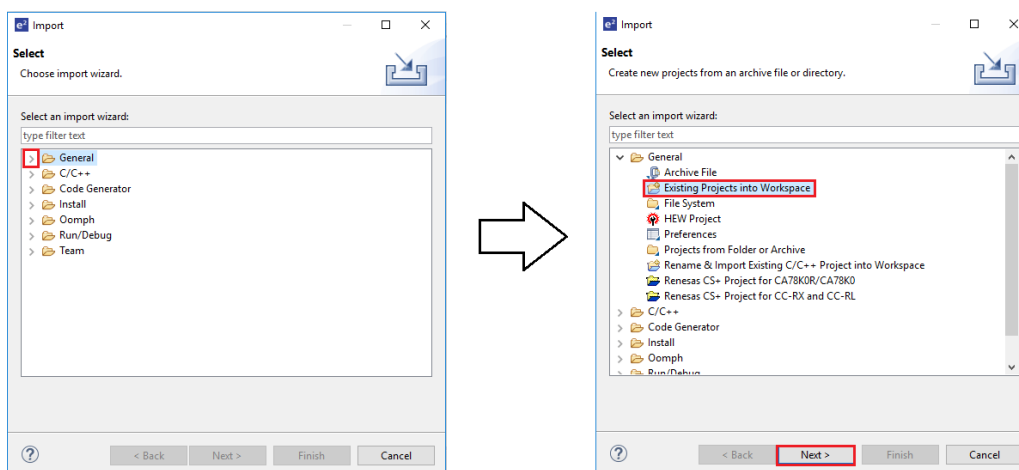


Figure 3.17 Settings in "Import"

In "Import Projects", select "Select archive file (A)", and click the "Browse (R)" button.

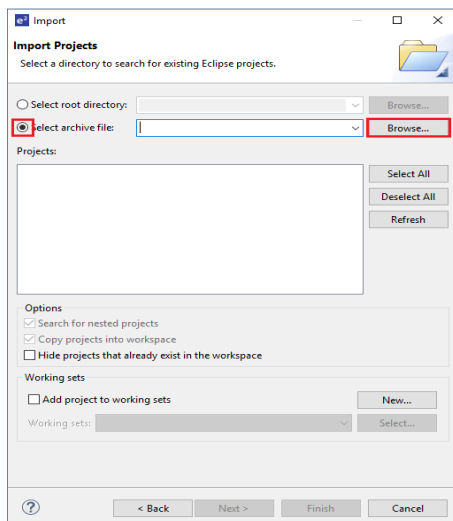


Figure 3.18 Settings in "Import Projects"

To select the file, open the workspace folder in which the existing project is stored, select e2studio.zip in the workspace folder, and click the "Open (O)" button.

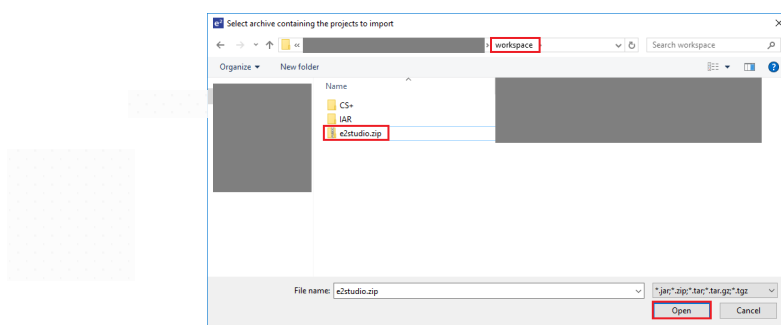


Figure 3.19 Selecting the existing project

Confirm that e2studio.zip is selected, and click the "Finish (F)" button.

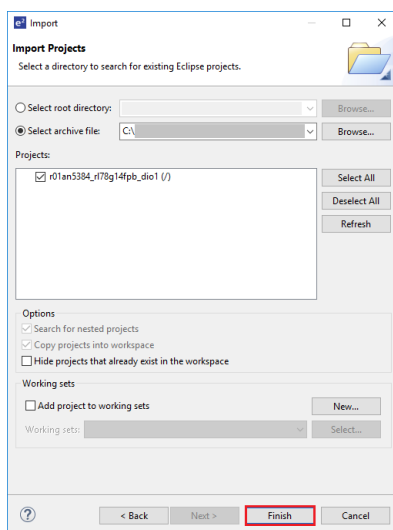


Figure 3.20 End of importing the existing project

(3) Project settings in e2studio

Click on "Project (P)" and, from the pulldown menus, select "Build Configurations", "Set Active", and "DefaultBuild (Debug on hardware)".

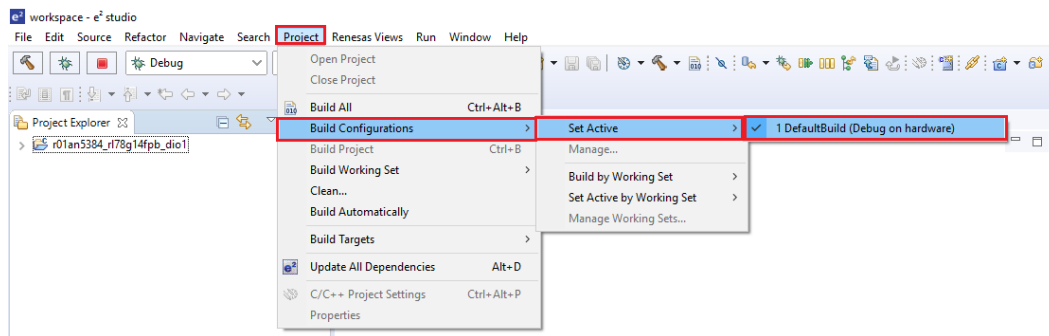


Figure 3.21 Project settings (e2studio)

(4) Building the project in e2studio

Click on "Project (P)" and select "Build All (A)" on the pulldown menu. The build is executed.

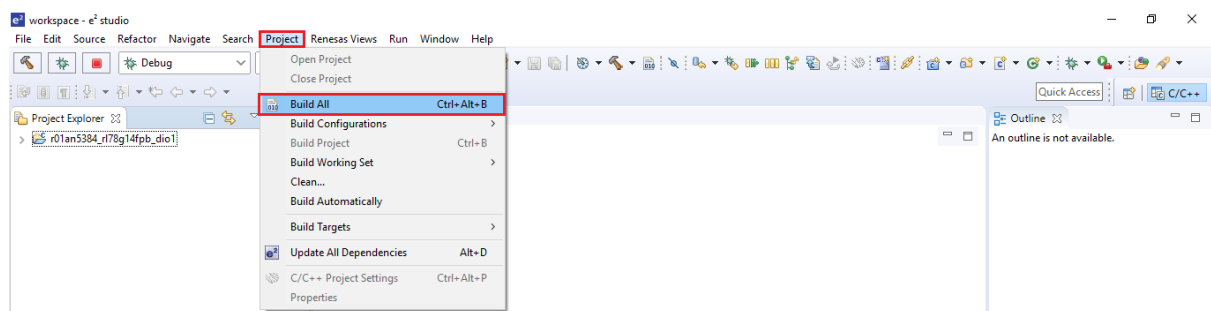


Figure 3.22 Project build (e2studio)

(5) Preparations for project debugging

When the build ends normally, prepare for debugging.

Click on "Window (W)" on the menu bar and, from the pulldown menus, select "Perspective (R)", "Open Perspective (O)", and "Debug".

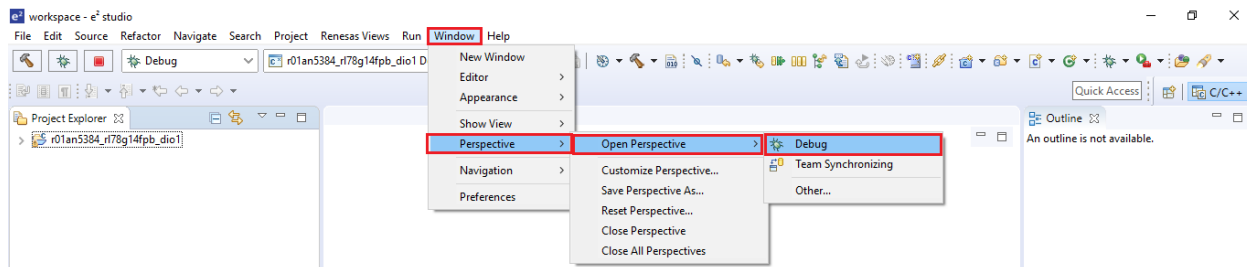


Figure 3.23 Debugging preparations (e2studio)

Click on "Run (R)" on the menu bar, and then click on "Debug Configurations (B)".

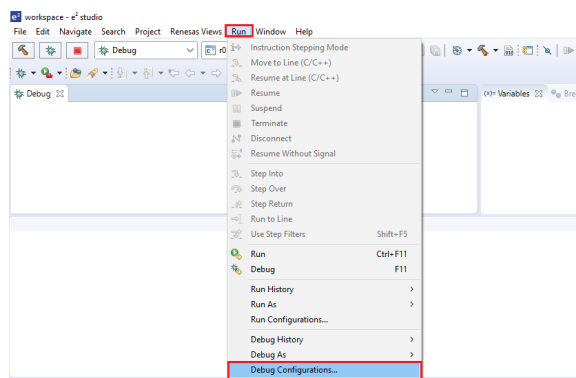


Figure 3.24 Debug configurations

On the "Debug Configurations" screen, select "project name" + "DefaultBuild" under "Renesas GDB Hardware Debugging" (in the case of Figure 3.25, select r01an5384_rl78g14fpb_dio1 DefaultBuild).

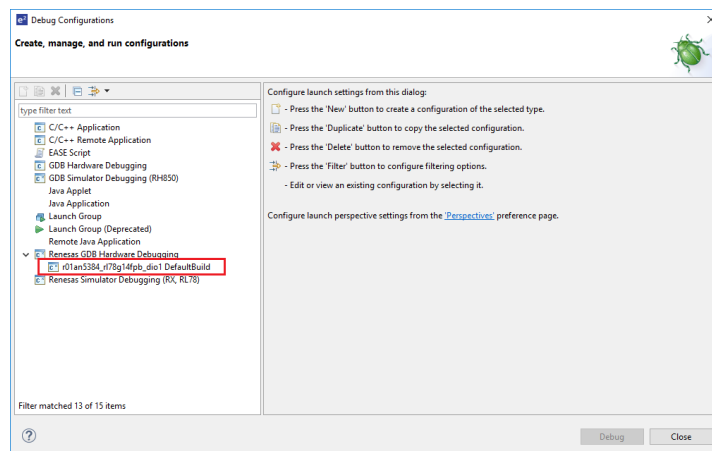


Figure 3.25 Debug configurations settings

Next, confirm that on the "Main" tab, the "Project (P)" is set to the project name (in the case of Figure 3.26, "r01an5384_rl78g14fpb_dio1"), and confirm that the "C/C++ Application" is set to "project name" + ".x".

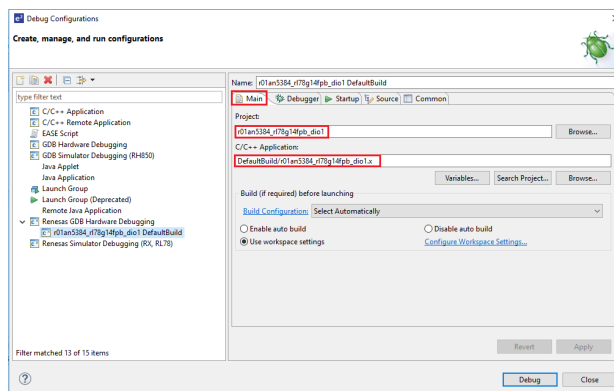


Figure 3.26 "Main" tab settings

Next, select the "Debugger" tab, and confirm that the "Debug hardware" is set to "E2 Lite (RL78)" and that the "Target Device" is the device being used (in the case of Figure 3.27, "R5F104ML").



Figure 3.27 "Debugger" tab settings

Next, on the "Connection Settings" tab within the "Debugger" tab, make the settings shown in Figure 3.28.

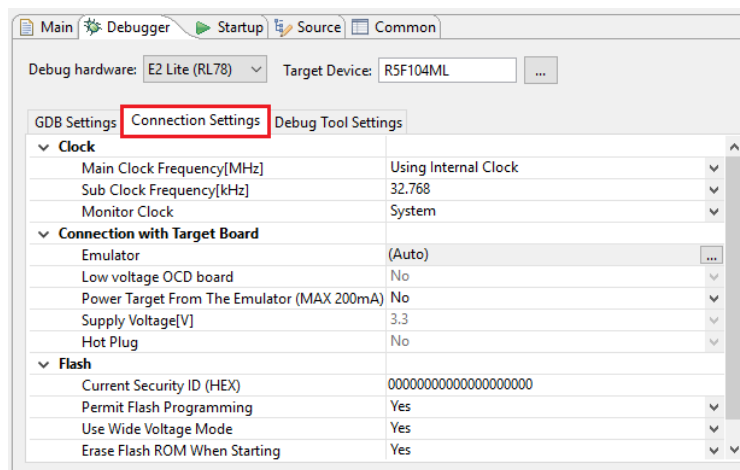


Figure 3.28 "Connection Settings" tab settings

Next, on the "Debug Tool Settings" tab, set the values shown in Figure 3.29. Be sure to set "Stop emulation of timer group when stopping" to "Yes".

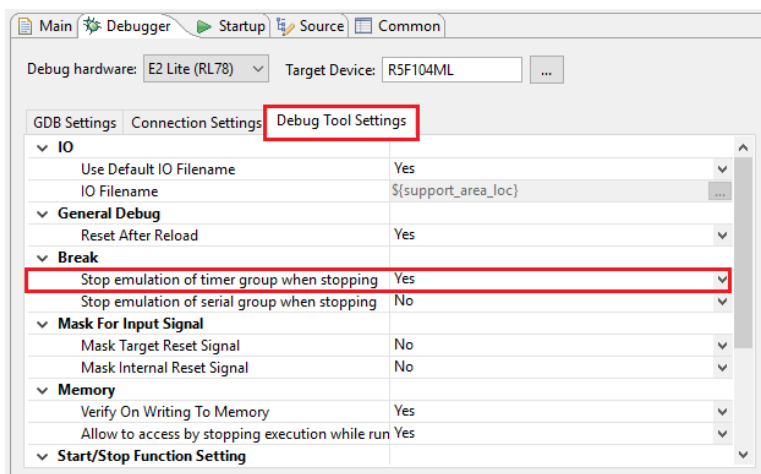


Figure 3.29 "Debug Tool Settings" tab settings

Next, confirm that on the "Startup" tab, the "Set breakpoint at" checkbox is checked and is set to "loop".

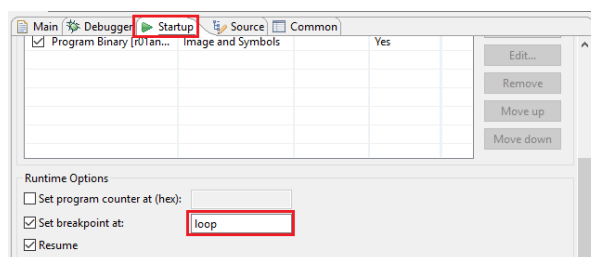


Figure 3.30 "Startup" tab settings

After confirming that settings have been made correctly, click the "Apply (Y)" button.

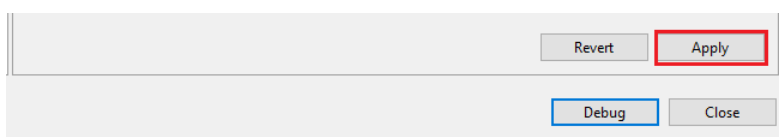


Figure 3.31 Applying the debug configurations

(6) Project debugging

Confirm that the PC and the FPB are properly connected via USB.

With the PC and the FPB properly connected by USB, upon clicking the "Debug (D)" button (see Figure 3.31), the program (sketch) is downloaded to the FPB, and execution is possible.

Each time thereafter, when debugging, click on "Run (R)" on the menu bar and then click on "Debug Configurations (B)". Then, click the "Debug (D)" button on the "Debug Configurations" screen (see Figure 3.31).

(7) Program execution

Click on "Run (R)" on the menu bar and then click on "Resume (M)".

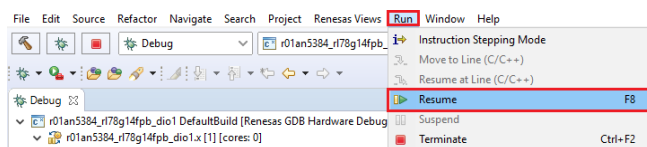


Figure 3.32 Start of program (sketch) execution (e2studio)

To suspend execution of the program (sketch), click on "Run (R)" on the menu bar, and then click on "Suspend (S)". Similarly, to end execution of the program (sketch), click on "Terminate (T)".

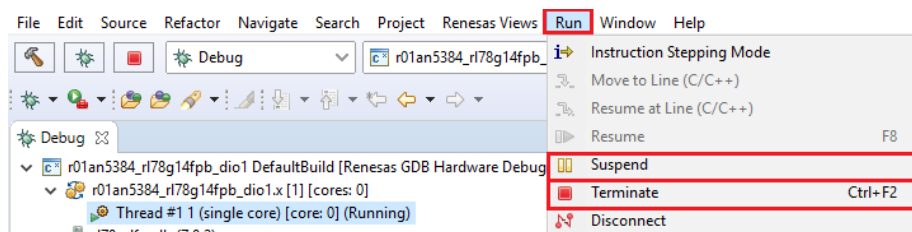


Figure 3.33 Suspending/terminating the program (sketch) (e2studio)

For details of debugging functions, refer to Chapter 5 of the e2studio Integrated Development Environment User's Manual and Introductory Guide (R20UT4374).

3.3 When using the IAR IDE

(1) Starting the IAR Integrated Development Environment

The files in the IAR folder are shown in Figure 3.34.

By double-clicking the eww file (in the case of Figure 3.34, r01an5384_rl78g14fpb_dio11.eww), the IAR integrated development environment is started.

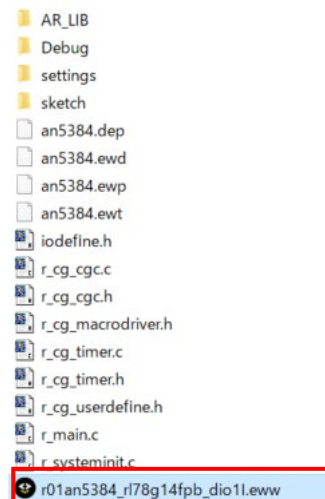


Figure 3.34 File configuration (IAR)

Upon starting the IAR integrated development environment, the screen shown in Figure 3.35 appears.

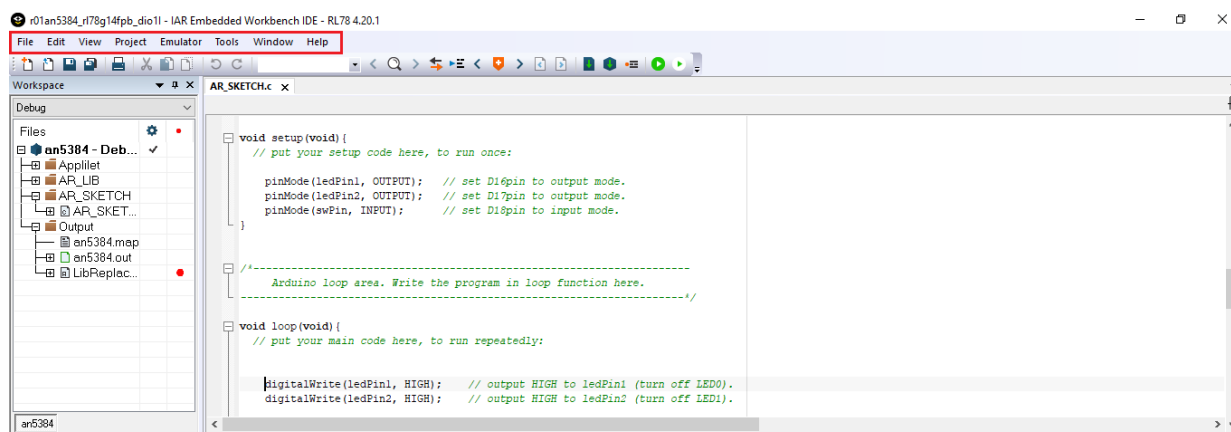


Figure 3.35 Screen after project startup (IAR)

(2) Procedure for project execution in IAR

Click on "Project (P)" on the menu bar and select "Make (M)" on the pulldown menu. Build processing is performed.

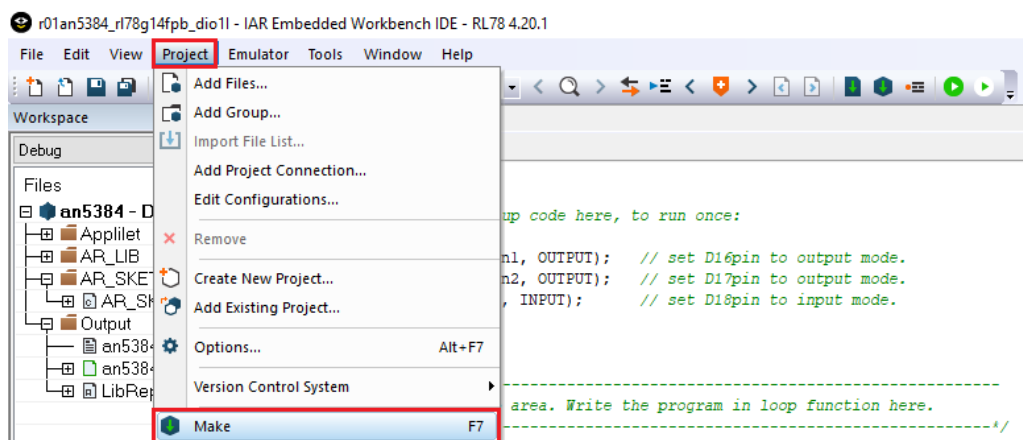


Figure 3.36 Build (IAR)

When the build processing ends normally, confirm that the PC and the FPB are properly connected via USB.

Next, click on "Project (P)" on the menu bar, and select "Download and Debug (D)" on the pulldown menu.

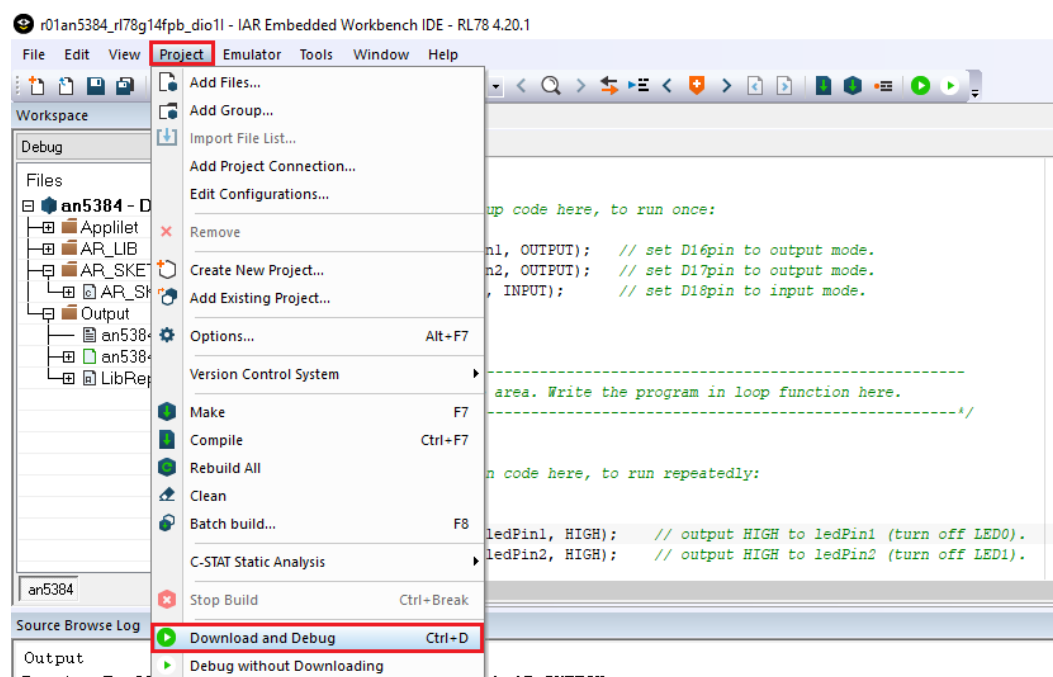


Figure 3.37 Downloading to the FPB (IAR)

If the PC and the FPB are properly connected via USB, the created (built) program (sketch) is downloaded to the FPB.

When the debugger is started, the first line of the loop function is highlighted, indicating that the program is executed from this point.

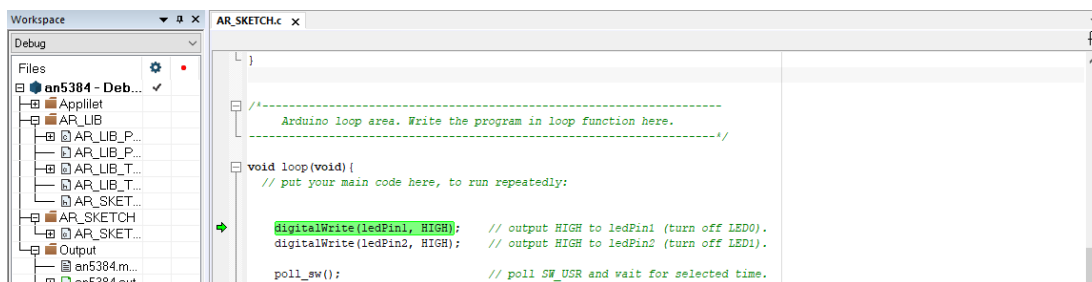


Figure 3.38 Screen after downloading (IAR)

To execute the program downloaded to the FPB, click on "Debug (D)" on the menu bar, and select "Go (G)" on the pulldown menu (see Figure 3.39).

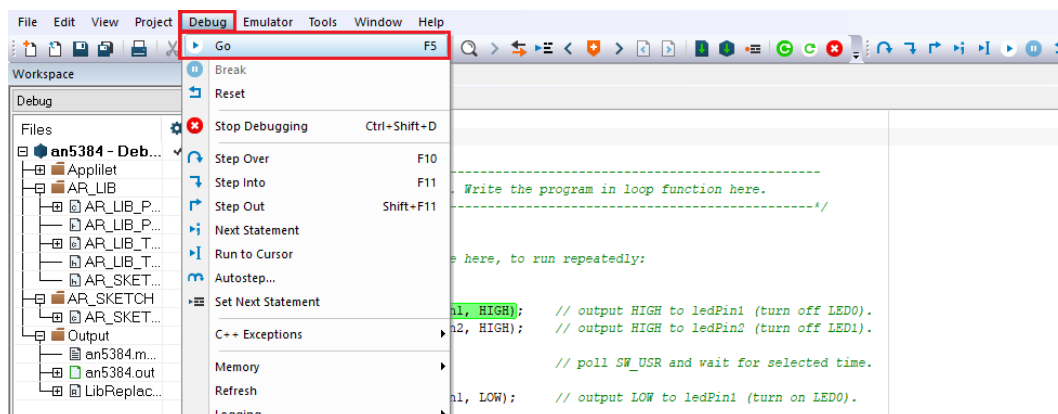


Figure 3.39 Starting execution of program (sketch) (IAR)

For details of the procedure for program debugging, please refer to the website and manual provided by IAR Systems.

To temporarily halt execution of the program (sketch), click on "Debug (D)" on the menu bar, and select "Break (B)" on the pulldown menu.

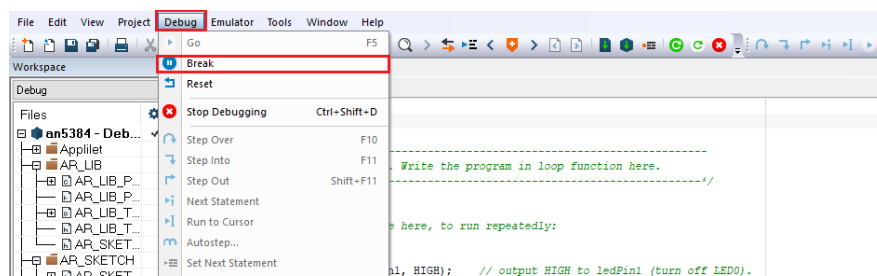


Figure 3.40 Temporarily halting program (sketch) (IAR)

To stop execution of the program (sketch), click on "Debug (D)" on the menu bar, and select "Stop Debugging (S)" on the pulldown menu.

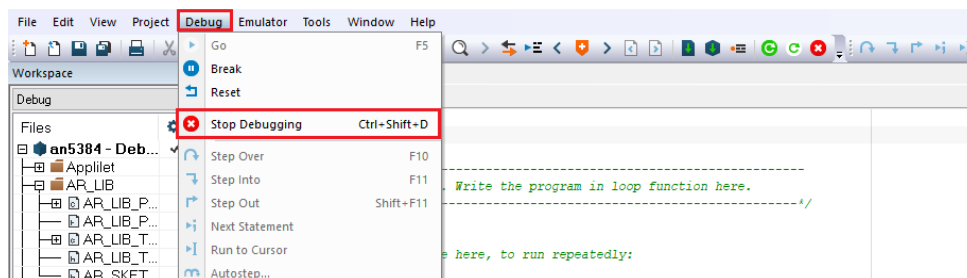


Figure 3.41 Stopping execution of program (sketch) (IAR)

4. Related Application Notes

Application notes related to this application note are indicated below.

They should be referred to as well.

RL78/G14 Onboard LED Flashing Control (Arduino API) (R01AN5384)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	June.30.20	—	First Edition

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

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

1. Precaution against Electrostatic Discharge (ESD)

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

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

2. Processing at power-on

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

3. Input of signal during power-off state

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

4. Handling of unused pins

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

5. Clock signals

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

6. Voltage application waveform at input pin

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

7. Prohibition of access to reserved addresses

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

8. Differences between products

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

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

6. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(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.4.0-1 November 2017)

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/