

RL78G13

Renesas Starter Kit Tutorial Manual
(IAR Embedded Workbench)

RENESAS MCU
RL Family / G1X Series

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corporation without notice. Please review the latest information published by Renesas Electronics Corporation through various means, including the Renesas Electronics Corporation website (<http://www.renesas.com>).

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be 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, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

Disclaimer

By using this Renesas Starter Kit (RSK), the user accepts the following terms:

The RSK is not guaranteed to be error free, and the entire risk as to the results and performance of the RSK is assumed by the User. The RSK is provided by Renesas on an "as is" basis without warranty of any kind whether express or implied, including but not limited to the implied warranties of satisfactory quality, fitness for a particular purpose, title and non-infringement of intellectual property rights with regard to the RSK. Renesas expressly disclaims all such warranties. Renesas or its affiliates shall in no event be liable for any loss of profit, loss of data, loss of contract, loss of business, damage to reputation or goodwill, any economic loss, any reprogramming or recall costs (whether the foregoing losses are direct or indirect) nor shall Renesas or its affiliates be liable for any other direct or indirect special, incidental or consequential damages arising out of or in relation to the use of this RSK, even if Renesas or its affiliates have been advised of the possibility of such damages.

Precautions

The following precautions should be observed when operating any RSK product:

This Renesas Starter Kit is only intended for use in a laboratory environment under ambient temperature and humidity conditions. A safe separation distance should be used between this and any sensitive equipment. Its use outside the laboratory, classroom, study area or similar such area invalidates conformity with the protection requirements of the Electromagnetic Compatibility Directive and could lead to prosecution.

The product generates, uses, and can radiate radio frequency energy and may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment causes harmful interference to radio or television reception, which can be determined by turning the equipment off or on, you are encouraged to try to correct the interference by one or more of the following measures;

- ensure attached cables do not lie across the equipment
- reorient the receiving antenna
- increase the distance between the equipment and the receiver
- connect the equipment into an outlet on a circuit different from that which the receiver is connected
- power down the equipment when not in use
- consult the dealer or an experienced radio/TV technician for help NOTE: It is recommended that wherever possible shielded interface cables are used.

The product is potentially susceptible to certain EMC phenomena. To mitigate against them it is recommended that the following measures be undertaken;

- The user is advised that mobile phones should not be used within 10m of the product when in use.
- The user is advised to take ESD precautions when handling the equipment.

The Renesas Starter Kit does not represent an ideal reference design for an end product and does not fulfil the regulatory standards for an end product.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the RSK hardware functionality, and electrical characteristics. It is intended for users designing sample code on the RSK platform, using the many different incorporated peripheral devices.

The manual comprises of an overview of the capabilities of the RSK product, but does not intend to be a guide to embedded programming or hardware design.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the RL78G13 Group. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

| Document Type | Description | Document Title | Document No. |
|----------------------|---|---------------------------------|--------------|
| User's Manual | Describes the technical details of the RSK hardware. | RSKRL78G13 User's Manual | R20UT0776EG |
| Tutorial | Provides a guide to setting up RSK environment, running sample code and debugging programs. | RSKRL78G13 Tutorial Manual | R20UT0614EG |
| Quick Start Guide | Provides simple instructions to setup the RSK and run the first sample, on a single A4 sheet. | RSKRL78G13 Quick Start Guide | R20UT0615EG |
| Software Help Manual | Describes the functionality of the sample code. | RSKRL78G13 Software Help Manual | R20UT0747EG |
| Schematics | Full detail circuit schematics of the RSK. | RSKRL78G13 Schematics | R20UT0458EG |
| Hardware Manual | Provides technical details of the RL78G13 microcontroller. | RL78G13 Hardware Manual | R01UH0146EJ |

2. List of Abbreviations and Acronyms

| Abbreviation | Full Form |
|---------------------|-----------------------------------|
| ADC | Analogue to Digital Converter |
| API | Application Programming Interface |
| CD | Compact Disk |
| CPU | Central Processing Unit |
| E1 | E1 Emulator |
| LCD | Liquid Crystal Display |
| LED | Light Emitting Diode |
| ROM | Read-Only Memory |
| RSK | Renesas Starter Kit |
| USB | Universal Serial Bus |

Table of Contents

| | |
|---|----|
| Table of Contents | 6 |
| 1. Overview | 7 |
| 1.1 Purpose..... | 7 |
| 1.2 Features..... | 7 |
| 2. Introduction..... | 8 |
| 3. Introduction to Applilet..... | 9 |
| 3.1 What is Applilet | 9 |
| 3.2 Features | 9 |
| 3.3 Applilet Application Models..... | 9 |
| 3.4 Project Creation | 10 |
| 4. Tutorial Project Workspace | 12 |
| 5. Project Workspace..... | 13 |
| 5.1 Introduction..... | 13 |
| 5.2 Starting IAR Embedded Workbench..... | 13 |
| 5.3 Build Configuration | 16 |
| 6. Building the Tutorial Program | 17 |
| 6.1 Building Code | 17 |
| 6.2 Connecting the Debugger..... | 18 |
| 6.3 Connecting to the Target with the E1 Debugger | 18 |
| 7. Running the Tutorial | 20 |
| 7.1 Downloading the Program Code..... | 20 |
| 7.2 Running the Tutorial | 20 |
| 8. Reviewing the Tutorial Program | 21 |
| 8.1 Program Initialisation..... | 21 |
| 8.2 Main Functions | 22 |
| 9. Additional Information..... | 25 |

1. Overview

1.1 Purpose

This RSK is an evaluation tool for Renesas microcontrollers. This manual describes how to get the RSK tutorial started, and basic debugging operations.

1.2 Features

This RSK provides an evaluation of the following features:

- Renesas microcontroller programming
- User code debugging
- User circuitry such as switches, LEDs and a potentiometer
- Sample application
- Sample peripheral device initialisation code

The RSK board contains all the circuitry required for microcontroller operation.

2. Introduction

This manual is designed to answer, in tutorial form, the most common questions asked about using a Renesas Starter Kit (RSK). The tutorials help explain the following:

- How do I compile, link, download and run a simple program on the RSK?
- How do I build an embedded application?
- How do I use Renesas' tools?

The project generator will create a tutorial project with two selectable build configurations.

- 'Debug' is a project built with the debugger support include.
- 'Release' is a project with optimised compile options, producing code suitable for release in a product.

Files referred to in this manual are installed using the project generator as you work through the tutorials. The tutorial examples in this manual assume that installation procedures described in the RSK Quick Start Guide have been completed. Please refer to the quick start guide for details of preparing the configuration.

These tutorials are designed to show you how to use the RSK and are not intended as a comprehensive introduction to the IAR's debugger, compiler toolchains or the E1 emulator. Please refer to the relevant user manuals for more in-depth information.

3. Introduction to Applilet

3.1 What is Applilet

Applilet is a software which enables you to output the source code (device driver programs, C source files and header files) necessary to control the peripheral hardware functions provided by the microcontroller device (clock generator, port functions, etc.) by configuring various information using the GUI.

3.2 Features

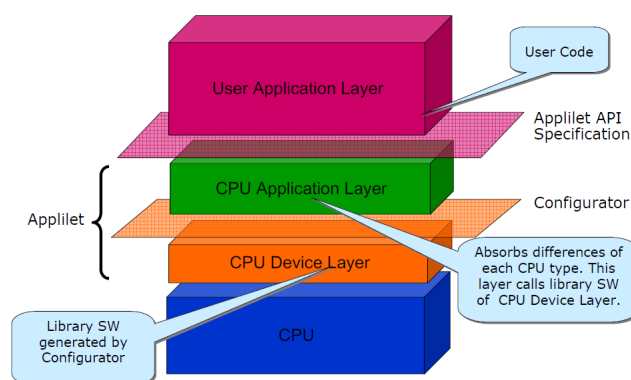
The Applilet3 has the following features:

- Reporting function
You can output configured information using the Applilet3 as files in various formats for use as design documents.
- Renaming function
The user can change default names assigned to the files output by the Applilet3 and the API functions contained in the source code.
- Code generating function
The Applilet3 can output not only device driver programs in accordance with the information configured using the GUI, but also a build environment such as sample programs containing main functions and link directive files.
- Project/workspace file generating function
The Applilet3 can output project and workspace files that can be used in application system integrated development environments such as IAR Embedded Workbench.

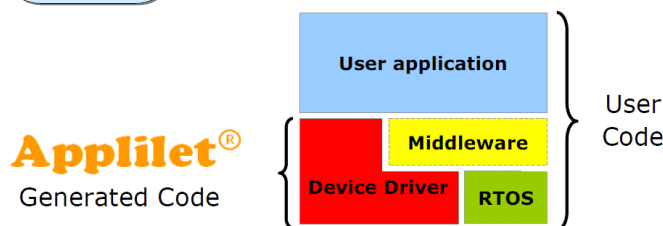
3.3 Applilet Application Models

The followings show the layer model and software model of Applilet:

- Application Layer Model.



- Application Software Model.



Applilet incorporates an easy to use GUI which aims to speed up the development process as well as improve code quality. The generated APIs are common to all family devices.

3.4 Project Creation

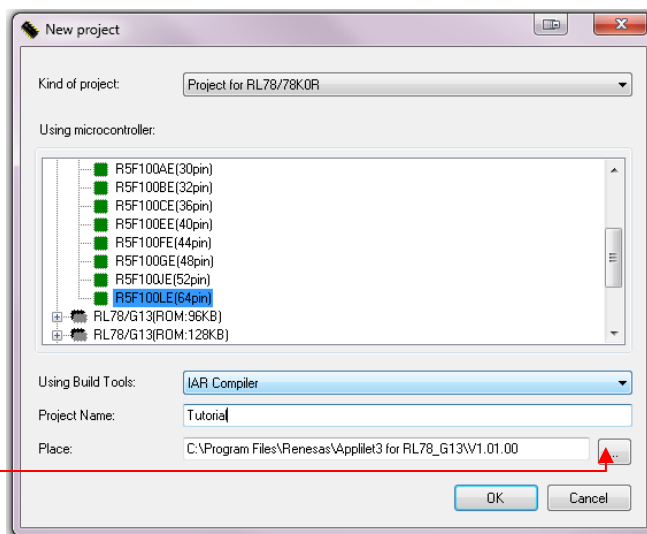
The integrated project wizard guides the user through the creation of a new project.

Launch Applilet3 for RL78G_13 V1.01.00 from the Windows Start Menu Start > All Programs > Applilet > Applilet3 for RL78_G13 > V1.0.01.00.

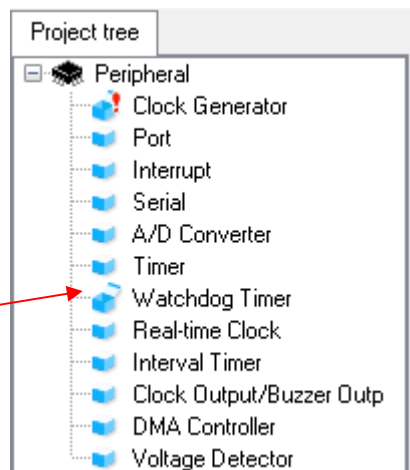
The steps required to create a new project and generate files are highlighted below:

- From Applilet3’s menu bar select File > New
This will open the ‘New Project’ window.
- Select the microcontroller device from the ‘Using Microcontroller:’ expandable list.
- Select ‘IAR Compiler’ as the build tool using the drop-down menu.
- Specify a ‘Project Name’.
- In the ‘Place’ field, specify a different folder to save the project.

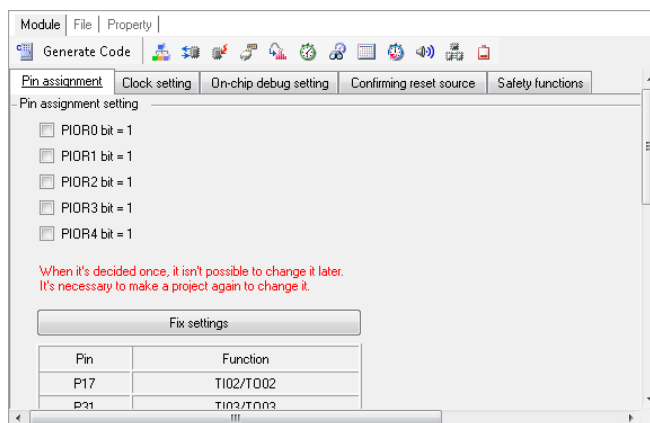
Note: Click here to create or locate a folder to save the project.



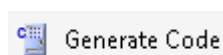
- The project tree shows all configurable peripherals of the selected device.
- The exclamation mark on the Clock Generator will disappear after the Pin Assignment configuration; shown in the screen-shot that follows the one opposite to this text.
- The Watchdog Timer module is configured and enabled by default; indicated by the open box.



- Click on the < Fix settings > button.
- The ‘Clock setting’ tab contains options for configuring the clocks.
- The ‘On-chip debug setting’ tab contains an option to enable/disable debugging and the use of a security ID. Ensure to set the ‘On-chip debug operation setting’ to ‘Used’ for debugging projects.



- The icons highlighted in the opposite screen-shot shows the available peripherals that can be configured. Click each icon to open the settings views and configure the desired peripherals.
- Click on ‘Generate Code’ after configuring all desired peripherals, to generate the project source and header files.
- The generated files can now be copied to the application system integrated development environments’ (IDE) project folder and added to the project; such as IAR Embedded Workbench.



Note: If files of the same name already exist in the IDE project folder, it may be necessary to back-up the old files before overwriting them.

Note: The Application sample project is an empty sample project with temporary files generated from Applilet. Users are encouraged to use Applilet to generate the source files containing API functions based on their application development’s requirements and use them in the Application sample project.

4. Tutorial Project Workspace

The workspace includes all the projects and files required to get started. The tutorial code is designed to show how code can be written, debugged and then downloaded.

Contents of common C files are controlled with defines set up in the build configuration options and `#ifdef` statements within the source files. Maintaining only one set of project files means that projects are more controllable.

5. Project Workspace

5.1 Introduction

IAR Embedded Workbench is an integrated development tool that allows the user to write, compile, program and debug a software project on the 78K, RL and V850 family of Renesas Microcontrollers. IAR will have been installed during the installation of the software support for the Renesas Starter Kit product. This manual will describe the stages required to create and debug the supplied tutorial code.

5.2 Starting IAR Embedded Workbench

To start the program, launch IAR Embedded Workbench from the Windows Start Menu.

When IAR is started, the IAR Information Centre will be shown:

IAR Information Center for Renesas RL78

Here you will find all the information you need to get started: tutorials, example projects, user and reference guides, support information, and release notes.

GETTING STARTED
Guidelines for setting up your project, adding files, compiling, linking, and debugging it.

USER GUIDES
Complete product documentation in PDF format gives you all the user and reference information you need.

EXAMPLE PROJECTS
Example applications that demonstrate hardware peripherals for specific devices and evaluation boards.

TUTORIALS
Tutorials to make you familiar with the IDE and the features of the IAR C-SPY debugger.

SUPPORT
For questions about how to use your IAR product, reporting a problem, or finding support resources.

RELEASE NOTES
All about the latest features, new device support, and program corrections.

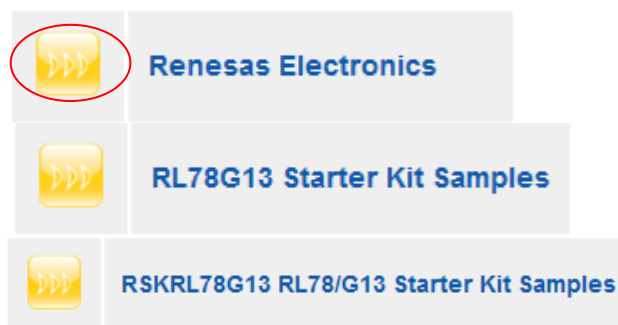
www.iar.com/resources
Read in-depth articles from our software engineers and product managers, see video presentations and check the webinar calendar or view the recordings.

www.iar.com/mypages
Create your user account and download product updates from MyPages. Here you can also manage your contact information and check your license and SUA status.



The IAR Information Centre provides all the information needed to get started; such as tutorials, example projects, user and reference guides, support information, and release notes.

To open the RSKRL78G13_Tutorial projects, click on the “Example Projects” section.

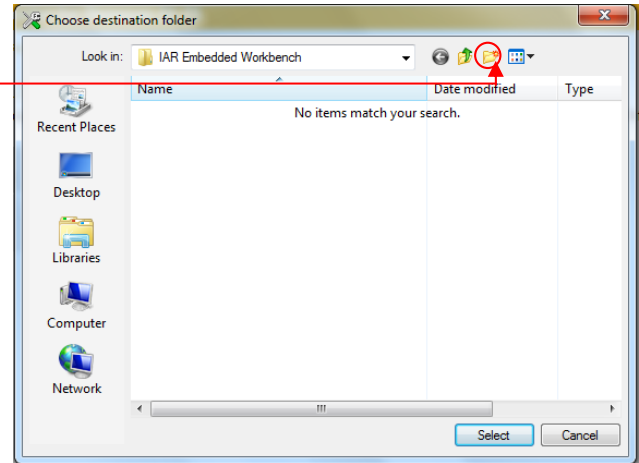
- Click the ‘link’ button next to “Renesas Electronics”.
- Click the ‘link’ button next to “RL78G13 Starter Kit Samples”.
- Click the ‘link’ next to “RSKRL78G13 RL78/G13 Starter Kit Samples” to open the RSKRL78G13_Tutorial workspace.



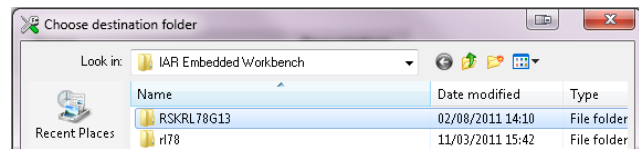
- Click the 'link' under 'Open project'.

| Info | Open project | Name | Description |
|---|---|-------------------|--------------|
|  |  | Project Workspace | Sample codes |

- When prompted to choose a destination folder:
- Click the 'New Folder' icon, highlighted by the circle.



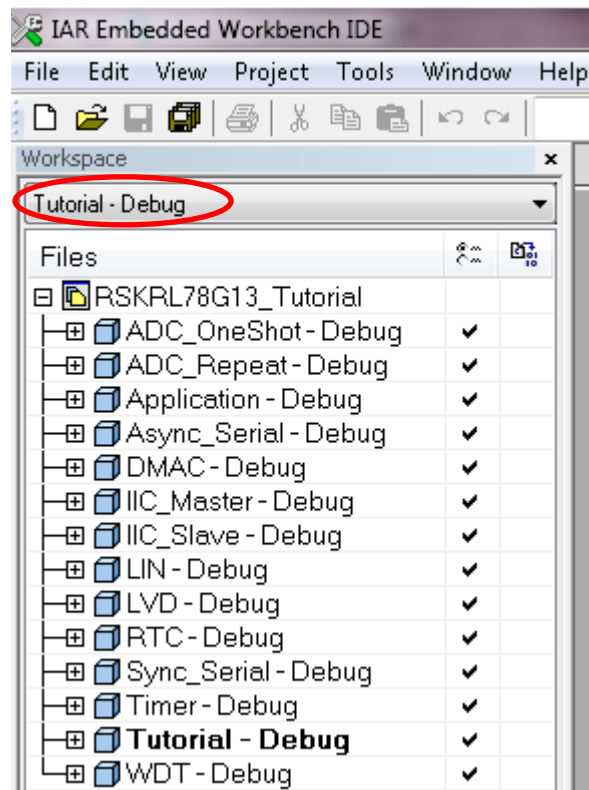
- Name the new folder 'RSKRL78G13'.
- Open the RSKRL78G13 folder.
- Click 'Select' to copy the project workspace.
- IAR will automatically open the workspace with the Information Centre displaying the available user manuals for the RL78G13.
- By default the RSKRL78G13_Tutorial workspace opens with the Tutorial sample project set as the active workspace.



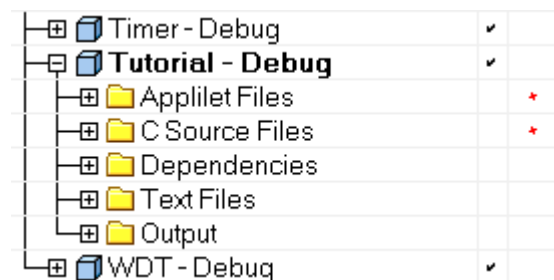
- The workspace as seen in IAR with all available projects.
- The active project’s name is shown on the “active project selection” drop-down list as circled in red. The bold letters within the Files tree gives an indication of which project is currently active.

Note:

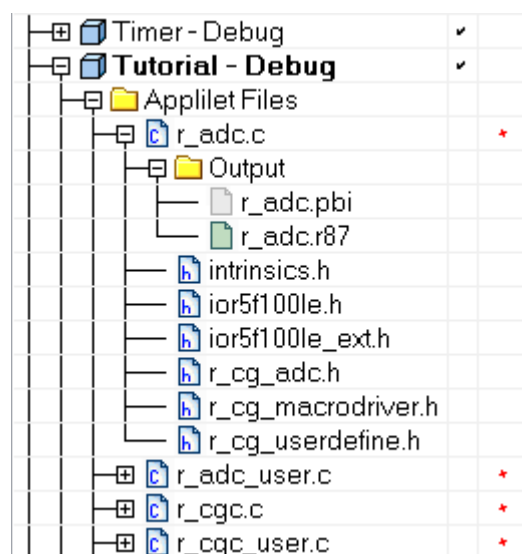
The active project can be changed by clicking on the “active project selection” drop-down list and selecting the desired project.



- Expanding the Tutorial project within the Files project tree reveals the files folder structure, which is common to all projects.
- Each folder, when expanded, reveals the files contained in it.



- All source files have a list of header files included in sources. The example opposite shows the source file “r_adc.c” with all included header files.
- IAR automatically includes the intrinsics.h, ior5f100le.h and ior5f100le_ext.h header files, hence. These files do not need to be specified in the source files. IAR selects these header files depending on the device specified when the projects were created.
- The Output folder contains debug-related files.
- The red next to source files indicate a change in the file since it was last saved.

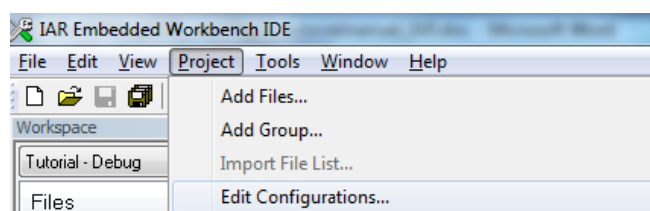


5.3 Build Configuration

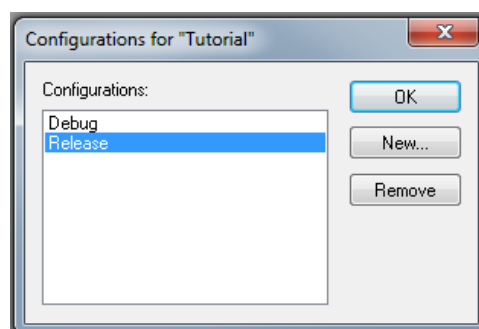
The build configurations are selected from the build tool's Property panel. The options available are Debug and Release. The Debug build is configured for use with the debugger. The Release build is configured for the final ROM-programmable code.

A common difference between the two build configurations is the optimisation setting and the addition of debug information. With optimisation turned on, the debugger may seem to execute code in an unexpected order. To assist in debugging it is often helpful to turn optimisation off on the code being debugged.

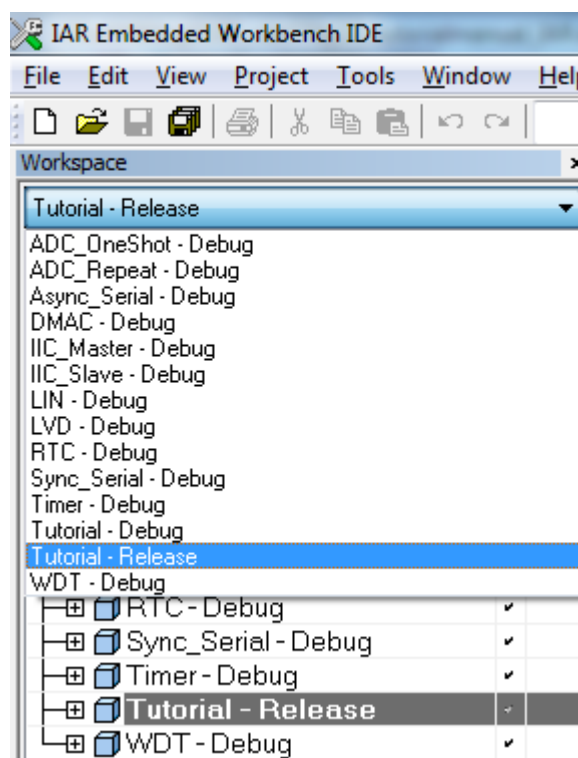
- To change build configuration, from the menu bar select [Project > Edit Configurations...].



- The "Configurations for" dialog will open.
- Select the desired configuration and press < OK >



- Alternatively, the build configuration can be changed from the "Project Selection" drop-down menu list within the Workspace window.



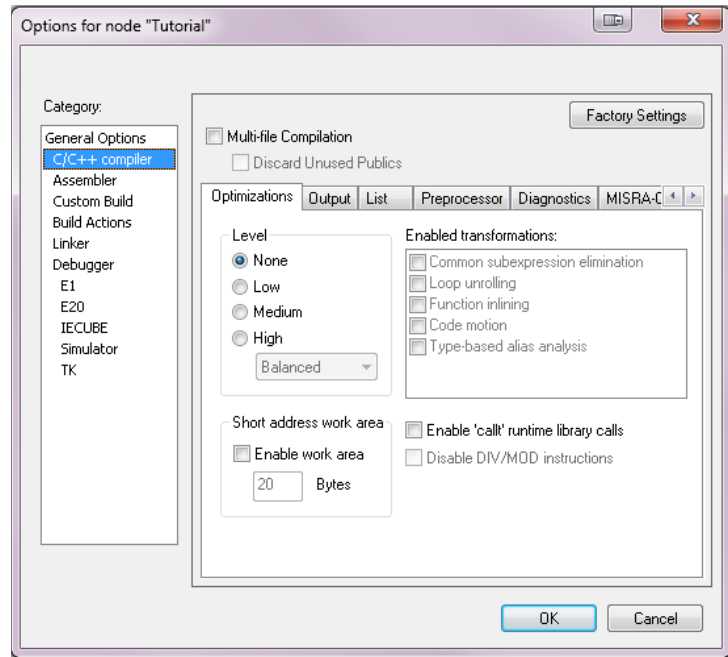
6. Building the Tutorial Program

The tutorial project build settings have been pre-configured in the node options. To view the node options select the 'Project' menu item and click 'Options'.

The Category pane on the left hand side will exist on all project node options. It is important when changing any setting to be aware of the current configuration that is being modified.

For the purpose of this tutorial, the user is not expected to alter any options.

- Review the options on each of the tabs and drop down lists to be aware of the options available. For the purposes of the tutorial, leave all options at default.
- When complete close the dialog box by clicking [OK].



6.1 Building Code

There is a choice of two shortcuts available for building the project.

- Selecting the 'Make' toolbar button will build the active project.
- The same action can be set from the menu bar [Project > Make].



- Pressing [F7]. This is equivalent to pressing the 'Make' toolbar button.
- Selecting 'Rebuild All' from the menu bar [Project > Rebuild All] will rebuild the active project.



Build the project now by pressing [F7] or pressing one of the build icons as shown above. During the build each stage will be reported in the Output Window. The build will complete with an indication of any errors and warnings encountered during the build.

6.2 Connecting the Debugger

For this tutorial it is not necessary to provide an external power supply to the board. The power will be obtained from the USB port. Please be aware that if you have too many devices connected to your USB port it may be shut down by Windows. If this happens remove some devices and try again. Alternatively provide an external power source taking care to ensure the correct polarity and voltage.

Other sample code supplied with this RSK will require a variable power supply; in which case an external 5V variable power supply should be used. Refer to the RSKRL78G13 User Manual for further details.

The Quick Start Guide provided with the Renesas Starter Kit board gives detailed instructions on how to connect the E1 to the host computer. The following assumes that the steps in the Quick Start Guide have been followed and the E1 drivers have been installed.

- Fit the LCD module to LCD connector on the board, via the header marked 'LCD'. Ensure all the pins of the connector are correctly inserted in the socket.
- Connect the E1 Debugger to a free USB port on your computer.
- Connect the E1 Debugger to the target hardware ensuring that it is plugged into the connector marked 'E1'.
- If supplying external power to the board please refer to Section 5.3 to turn off the option of supplying power from the E1 before turning on the external power supply.

6.3 Connecting to the Target with the E1 Debugger

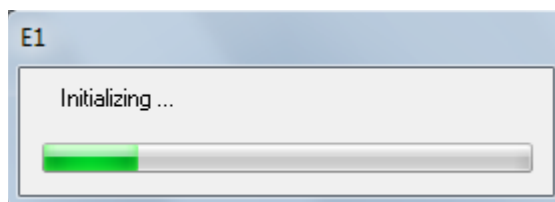
This section will take you through the process of connecting to the device, programming the Flash and executing the code.

Please note that the "Emulator Mode" wizard shown here will only appear the FIRST time you connect to the target within a project. On subsequent connections the "Emulator Setting" dialog will appear please choose the same options to connect.

- Click the 'Download & Debug' button to connect the target with the E1 debugger and download the code to the target.



- The user will see the dialog shown opposite.



- Open the hardware setup settings from the menu bar [Emulator > Hardware Setup...]

Alternatively, click the 'Hardware setup' icon from the menu bar.



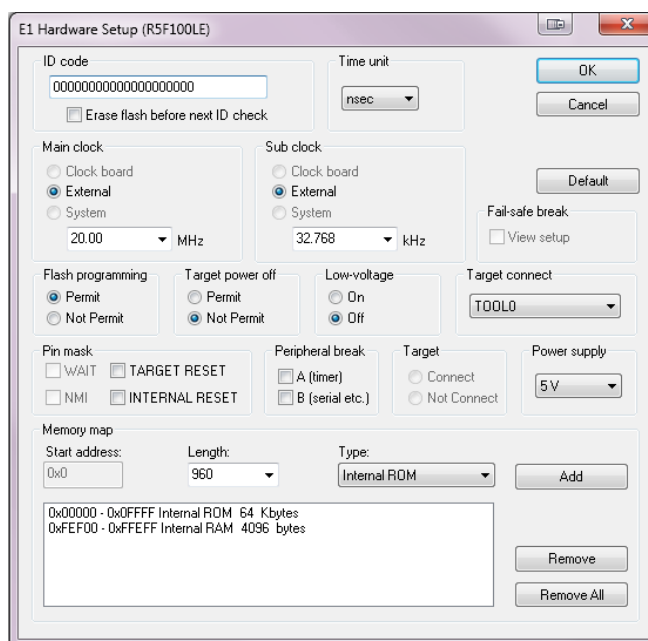
- On the E1 Hardware Setup (R5F100LE) dialog, verify that the main clock and sub clock values are set as;

Main clock frequency 20.00 [MHz]

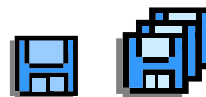
Sub clock frequency 32.768 [kHz]

- Ensure the power supply is set to 5V to power the target from the E1 debugger.
- Click < Cancel > to close the dialog.

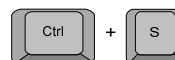
Note: The drop-down menus are used to select from the pre-defined frequencies on the list. Any changes to these settings will take effect after disconnecting then reconnecting the debugger. When supplying external power to the RSK, set the 'Power Supply' option as 'Target'.



You can also save files by clicking the 'Save' or 'Save All' buttons from the IAR toolbar.



You can also save files using the following keyboard shortcut:



7. Running the Tutorial

7.1 Downloading the Program Code

Now that the code has been built and downloaded to the RSK IAR, the code needs to be executed.

- On completion of program download, the debugger and code are ready to be executed. This should have occurred during connection between the target and the E1 debugger. The program counter indicator will point to the main function; this is the program's entry point.

```
*/  
void main(void)  
{  
    /* Start user code. Do not edit comment generated here */  
  
    /* Initialise the LCD module. */  
    InitialiseDisplay();  
  
    /* Display information on the debug LCD. */  
    DisplayString(LCD_LINE1, "Renesas");  
    DisplayString(LCD_LINE2, NICKNAME);  
}
```

7.2 Running the Tutorial

Once the program has been downloaded onto the RSK device, the program can be executed. Click the 'Go' button to begin the program from the current program counter position. It is recommended that you run through the program once first, and then continue to the review section.



8. Reviewing the Tutorial Program

This section will look at each section of the tutorial code, how it works, and how it could be altered to be implemented into more complex code.

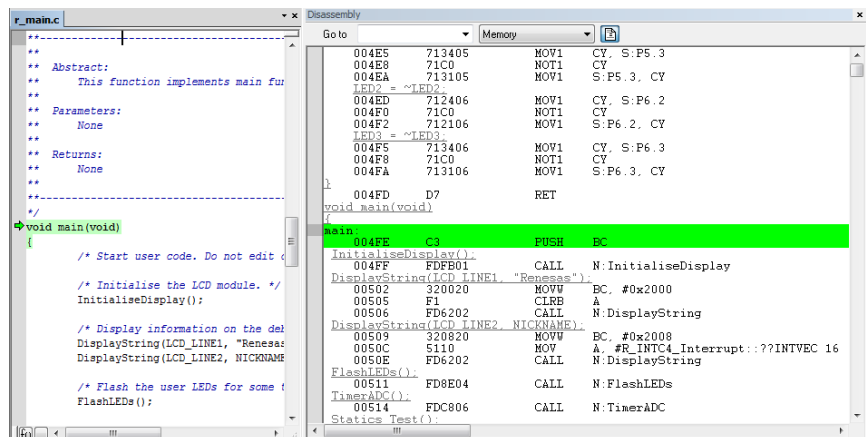
8.1 Program Initialisation

Before the main program can run, the microcontroller must be configured. Due to the debugger configuration used for the Tutorial project and the rest of the sample projects, the user will not be able to step through the hardware initialisation code as a result. Please refer to Section 4.3 to change the entry point after programming the microcontroller. Specify 'R_Systeminit' as the function name if viewing of hardware initialisation is desired. The initialisation code is executed every time the device is reset via the reset switch or from a power reboot. The user is advised not to use the 'step' feature of the debugger to exit the R_Systeminit function.

Ensuring the Tutorial program has been downloaded onto the RL78G13; press the 'Reset CPU' button on the Debug Toolbar.



- From the Menu bar select View > Disassemble > Disassemble to view the source and disassembly files side by side.

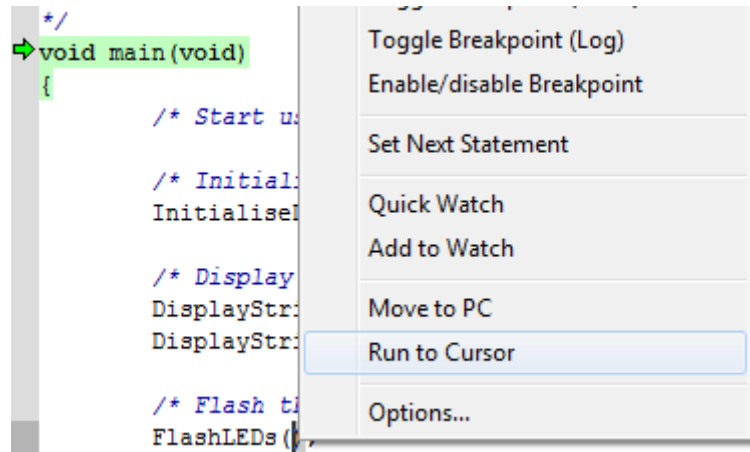


Revert back to the source by clicking on the file containing the function pointed to by the program counter indicator.

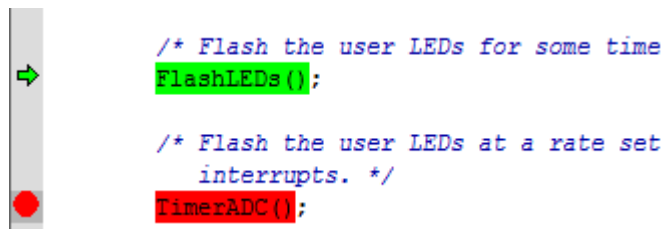
8.2 Main Functions

This section will look at the program code called from with the main() function, and how it works.

- Right click the 'FlashLEDs()' function call and select 'Run to Cursor' to execute the program up to this line. The function 'InitialiseDisplay()' call initialises the debug LCD and the 'DisplayString()' function calls will write the strings "Renesas " and "RL78G13 " on line 1 and 2, respectively.



- Set a break-point on the 'TimerADC()' function call by double-clicking on the Breakpoint column as shown on the opposite.
- Click the 'Step In' button to step into the 'FlashLEDs()' function. Alternatively, press F11.



- The program counter should now move into the FlashLEDs function definition. This function enables the external interrupts connected to the user switches and flashes the user LEDs 200 times or until a user switch has been pressed. On exiting the loop, the function disables the switch interrupts and returns to the main function.
- Execute the program by pressing the 'Go' button. Alternatively, press F5.



Press any user switch to exit the function before the user LEDs flash 200 times.

```
void FlashLEDs(void)
{
    /* Declare a loop iteration count variable */
    uint32_t ulLed_Delay;
    uint8_t ucLEDcount;

    /* Enable SW1 interrupts */
    R_INTC1_Start();

    /* Enable SW2 interrupts */
    R_INTC2_Start();

    /* Enable SW3 interrupts */
    R_INTC4_Start();

    /* Flash the LEDs for 200 times or until a user switch is pressed. */
    for(ucLEDcount=0; ucLEDcount<200; ucLEDcount++)
    {
        /* If a user switch was pressed, exit the loop. */
        if( gSwitchFlag > 0x00)
        {
            break;
        }
    }
}
```

- The program counter should come to a halt at the TimerADC function.
- Step over the function by clicking the ‘Step Over’ button. Alternatively, press F10.



The TimerADC function starts a continuous A/D conversion and a periodic timer whose period is updated with the ADC result.

```

/* Flash the user LEDs for some time or until a push button is
FlashLEDs();

/* Flash the user LEDs at a rate set by the user potentiometer
interrupts. */
TimerADC();

/* Demonstration of initialised variables. Use this function with
debugger.*/
Statics_Test();

/* Halt program in an infinite while loop */
while (1U)
{
    ;
}
    
```

- Open the r_timer_user.c file. Locate the first line of code inside the R_TAU0_Channel1_Interrupt interrupt handler:
LED0 = ~LED0;

```

#pragma vector = INTIM01_vect
__interrupt void R_TAU0_Channel1_Interrupt(void)
{
    /* Start user code. Do not edit comment generated here */

    /* Toggle user LEDs */
    LED0 = ~LED0;
    LED1 = ~LED1;
    LED2 = ~LED2;
    LED3 = ~LED3;

    /* Store the gTimerADCperiod value of the variable */

    /* Ensure that the value is below 0x75 */
    if(gTimerADCperiod > 0x75)
    {
        gTimerADCperiod = 0x75;
    }

    /* Update the gTimerADCperiod value */
}
    
```

- Right click the instruction and select ‘Toggle Breakpoint (Code)’. Alternatively, set the cursor on the instruction and click on the Toggle Breakpoint’ button.



- Execute the program by pressing the ‘Go’ button.

- The program will halt at the breakpoint due to the timer's period elapsing.

```
#pragma vector = INTIM01_vect
__interrupt void R_TAU0_Channell1_Interrupt(void)
{
    /* Start user code. Do not edit comment generated here */

    /* Toggle user LEDs */
    LED0 = ~LED0;
    LED1 = ~LED1;
    LED2 = ~LED2;
    LED3 = ~LED3;
}
```

- The cursor should automatically be set on the instruction.
- Remove the breakpoint by clicking on the 'Toggle Breakpoint' button once.
- Press F5 to resume program execution and observe the string on the debug LCD's second line change one character at a time from 'STATIC' to 'TESTTEST' as the Statics_Test function is executed.
- After all characters have been changed, the debug LCD's second line will display "RL78G13".

```
void Statics_Test(void)
{
    /* Declare loop count variable */
    uint8_t uiCount;
    /* Declare string variable to hold the string to be copied */
    uint8_t ucStr[] = "STATIC \0";
    /* Declare variable buffer to store the copied string */
    uint8_t ucReplace[] = "TESTTEST\0";

    /* Declare a delay count variable */
    uint32_t ulDelay;

    /* Write ucStr variable, "STATIC" to LCD */
    DisplayString(LCD_LINE2, (int8_t*)ucStr);

    /* Begin for loop which writes one letter of ucReplace to the LCD at a time
    The nested while loops generate the delay between each letter change */
    for (uiCount=0; uiCount<8; uiCount++)
    {
        /* Replace letter number uiCount of ucStr from ucReplace */
        ucStr[uiCount] = ucReplace[uiCount];

        /* Display the character on the debug LCD */
        DisplayString(LCD_LINE2, (int8_t*)ucStr);

        /* LED Flashing Delay */
        for(ulDelay=0; ulDelay<700000; ulDelay++)
        {
            /* Delay */
        }
    }
}
```

- This is the extent of the tutorial code.

For further details regarding hardware configuration, please refer to the RSKRL78G13 User's Manual and the RL78G13 Hardware Manual.

The E1 emulator features advanced logic-based event point trigger system, and full instruction on its use is outside the scope of this tutorial. For further details, please refer to the E1 Emulator User's Manual

9. Additional Information

Technical Support

For more details on how to use IAR Embedded Workbench, refer to [Help > Content] from the menu bar within the IAR Embedded Workbench.

For information about the RL78G13 series microcontrollers refer to the RL78G13 Group hardware manual.

For information about the RL78G13 assembly language, refer to the RL78 Series Software Manual.

Online technical support and information is available at: <http://www.renesas.com/rskrl78g13>

Technical Contact Details

America: techsupport.america@renesas.com

Europe: tools.support.eu@renesas.com

Japan: csc@renesas.com

General information on Renesas Microcontrollers can be found on the Renesas website at:

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

Trademarks

All brand or product names used in this manual are trademarks or registered trademarks of their respective companies or organisations.

Copyright

This document may be, wholly or partially, subject to change without notice. All rights reserved. Duplication of this document, either in whole or part is prohibited without the written permission of Renesas Electronics Europe Limited.

© 2011 Renesas Electronics Europe Limited. All rights reserved.

© 2011 Renesas Electronics Corporation. All rights reserved.

© 2011 Renesas Solutions Corp. All rights reserved.

| | |
|------------------|----------------------------|
| REVISION HISTORY | RSKRL78G13 Tutorial Manual |
|------------------|----------------------------|

| Rev. | Date | Description | |
|------|--------------|-------------|----------------------|
| | | Page | Summary |
| 1.00 | Jun 21, 2011 | — | First Edition issued |
| | | | |

Renesas Starter Kit Tutorial Manual

Publication Date: Rev.1.00 Jun 21, 2011

Published by: Renesas Electronics Corporation



Renesas Electronics Corporation

<http://www.renesas.com>

SALES OFFICES

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

Renesas Electronics America Inc.

2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited

1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH

Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.

7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.

Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited

Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.

7F, No. 363 Fu Shing North Road Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.

1 harbourFront Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.

11F, Samik Laviéd' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

RL78G13 Group



Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Renesas Electronics:

[YR0K50104PS000BE](#) [YR0K50563NS010BE](#) [YR0K50564MS000BE](#) [YR0K50571MS000BE](#) [YR0K505111S000BE](#)
[YR0K505231S000BE](#) [YR0K50100LS000BE](#)