User's Manual

32

RX113 Group

Renesas Starter Kit Tutorial Manual
For CS+

RENESAS MCU
RX Family / RX100 Series

Renesas Electronics
www.renesas.com

Rev. 1.01 Mar 2015

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 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The recommended 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; and industrial robots etc.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anticrime systems; and safety equipment etc.

   Renesas Electronics products are neither intended nor 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 damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
6. 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.
7. 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 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. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

(2012.4)

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 how to use the CS+ IDE to develop and debug software for the RSK platform. It is intended for users designing sample code on the RSK platform, using the many different incorporated peripheral devices.

The manual comprises of step-by-step instructions to load and debug a project in CS+, but does not intend to be a complete guide to software development on the RSK platform. Further details regarding operating the RX113 microcontroller may be found in the Hardware Manual and within the provided sample code.

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 RX113 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.	RSKRX113 User's Manual	R20UT2756EG
Tutorial Manual	Provides a guide to setting up RSK environment, running sample code and debugging programs.	RSKRX113 Tutorial Manual	R20UT2757EG
Code Generator Tutorial	Provides a guide to code generation and importing into the CS+ IDE.	RSKRX113 Code Generator Tutorial Manual	R20UT3254EG
Quick Start Guide	Provides simple instructions to setup the RSK and run the first sample, on a single A4 sheet.	RSKRX113 Quick Start Guide	R20UT2758EG
Schematics	Full detail circuit schematics of the RSK.	RSKRX113 Schematics	R20UT2755EG
Hardware Manual	Provides technical details of the RX113 microcontroller.	RX113 Group, User's Manual: Hardware	R01UH0448EJ

2. List of Abbreviations and Acronyms

Abbreviation	Full Form
ADC	Analog-to-Digital Converter
CPU	Central Processing Unit
E1	Renesas On-chip Debugger
GDB	GNU Debugger
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MCU	Micro-controller Unit
ROM	Read Only Memory
RSK	Renesas Starter Kit
SCI	Serial Communications Interface
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

Table of Contents

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 three selectable build configurations:

- 'DefaultBuild' is a project with debug support and optimisation level set to two.
- 'Debug' is a project built with the debugger support included. Optimisation is set to zero.
- 'Release' is a project with optimised compile options, producing code suitable for release in a product. Optimisation is set to two.

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 CS+ debugger, compiler toolchains or the E1 emulator. Please refer to the relevant user manuals for more in-depth information.

2.1 Code Generator Plug in

The Code Generator plug in for the RX113 has been used to generate the sample code discussed in this document. Code Generator for CS+ is a plug in tool for generating template 'C' source code and project settings for the RX113. When using Code Generator, the engineer is able to configure various MCU features and operating parameters using intuitive GUI controls, thereby bypassing the need in most cases to refer to sections of the Hardware Manual.

Code Generator is not enabled by default during CS+ installation and is not required for this tutorial manual. For more information please refer to the 'Enabling Code Generator' section of the Code Generator Tutorial manual.

Once the engineer has configured the project, the 'Generate Code' function is used to generate three code modules for each specific MCU feature selected. These code modules are name 'r_cg_xxx.h', 'r_cg_xxx.c', and 'r_cg_xxx_user.c', where 'xxx' is a three letter acronym for the relevant MCU feature, for example 'adc'. Within these code modules, the engineer is then free to add custom code to meet their specific requirement. Custom code should be added, whenever possible, in between the following comment delimiters:

```
/* Start user code for adding. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
```

Code Generator will locate these comment delimiters, and preserve any custom code inside the delimiters on subsequent code generation operations. This is useful if, after adding custom code, the engineer needs to re-visit Code Generator to change any MCU operating parameters.

3. Tutorial Project Workspace

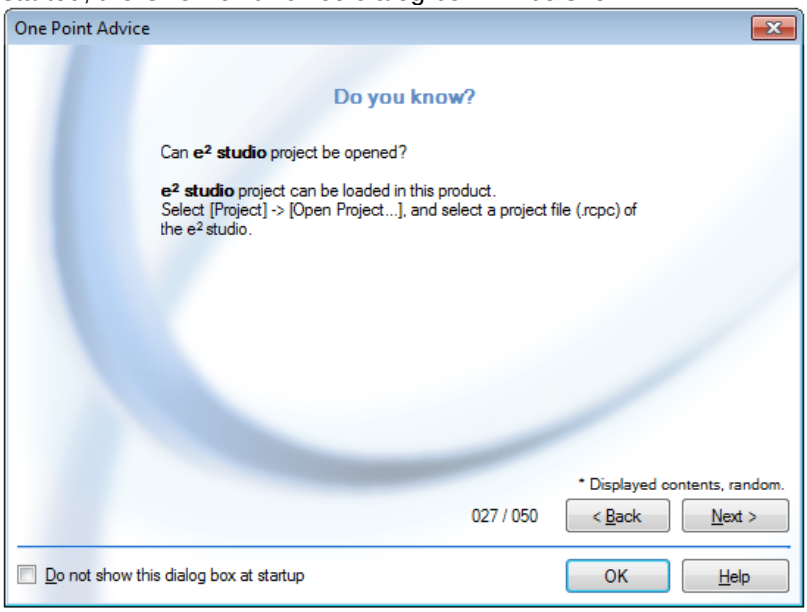
3.1 Introduction

CS+ is an integrated development tool that allows the user to write, compile, program and debug a software project on the RX, 78K, RL and V850 family of Renesas microcontrollers. CS+ 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 project.

3.2 Starting CS+

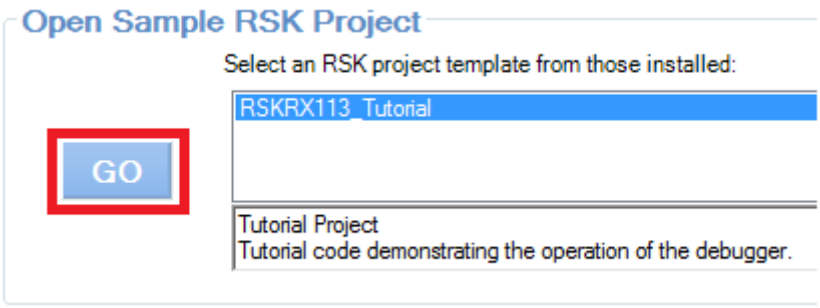
To use the program, start CS+:
Windows™ 7 & Vista: Start Menu (Start Menu > All Programs > Renesas Electronics CS+ > CS+ for CC (RX, RH850)
Windows™ 8: From Apps View, click 'CS+ for CC (RX, RH850)' icon

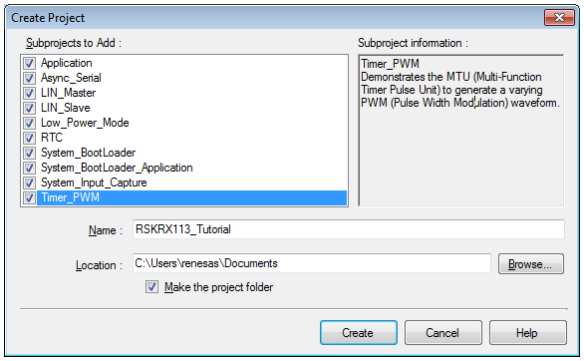
The first time CS+ is started, the One Point Advice dialog box will be shown:



The One Point Advice dialog box provides some useful tips when using CS+. Press 'OK' to skip the advice and close the One Point Advice dialog. The user will then be presented with the Start panel.

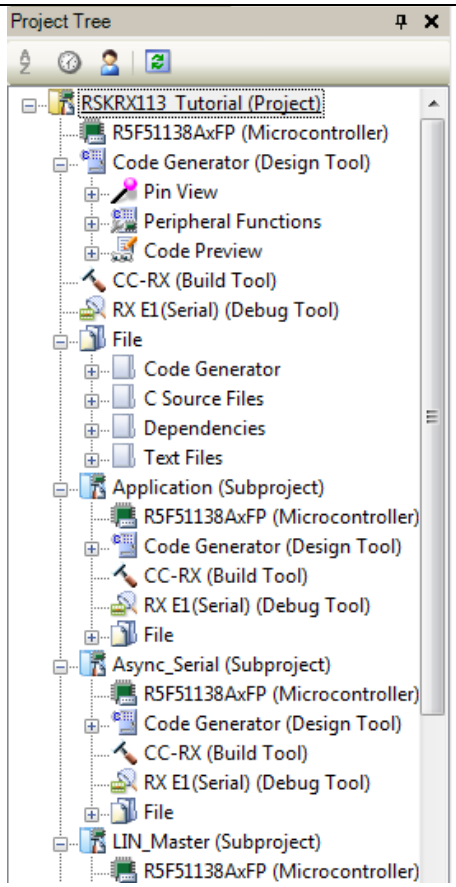
Under the 'Open Sample RSK Project', open a new Tutorial project by selecting the RSKRX113_Tutorial project template and click on 'Go' as shown below.



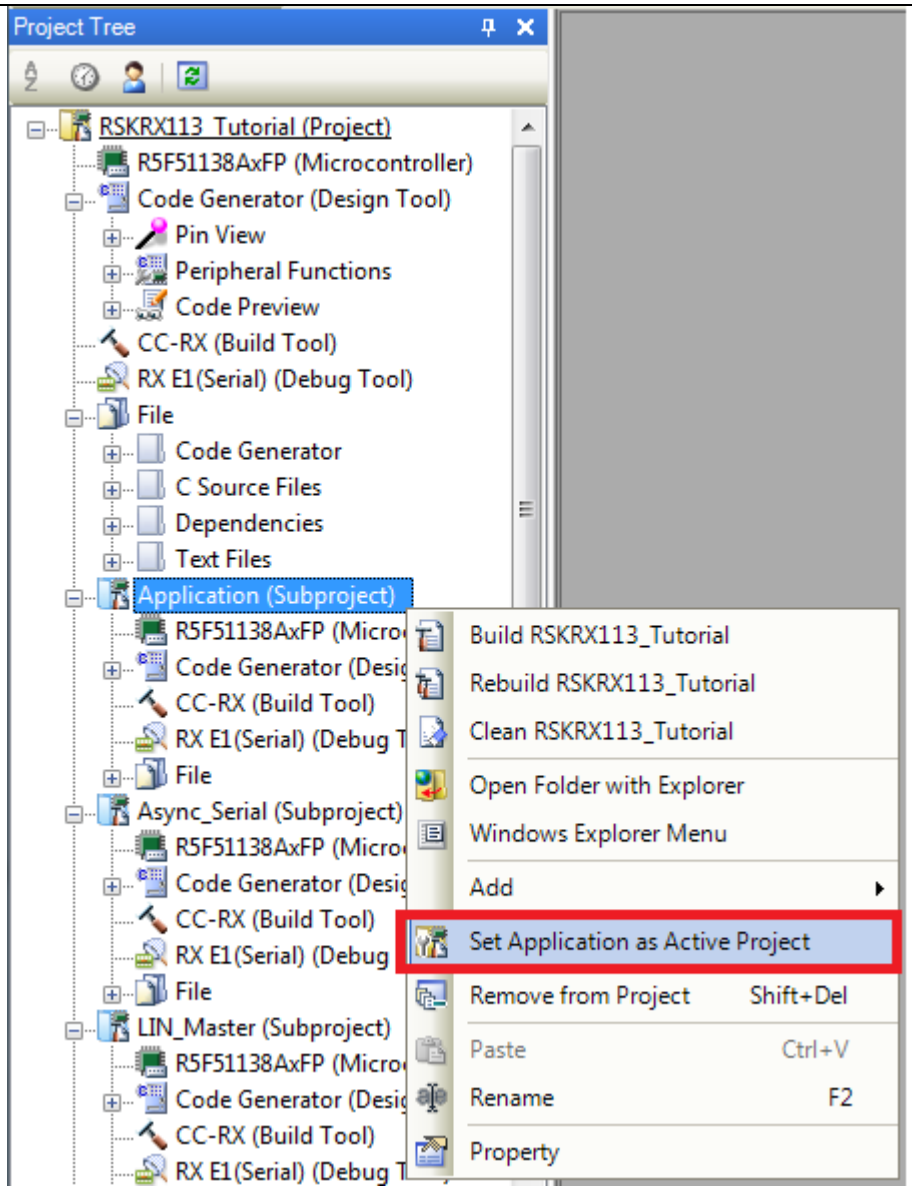
• will present a 'Create Project' dialog box. • Select all sub-projects by clicking on each checkbox and observe the information displayed under the 'Subproject information' heading as you select each project. • Specify a name and location for the new project and click on 'Create' • A dialog box will appear if the location specified does not exist; asking to create the folder specified. Click 'OK'.	

- CS+ will create and open the project showing the Project Tree as seen in the screenshot opposite.
- RSKRX113_Tutorial (Project) is the master project and includes the tools to modify, build and debug the code.
- The File folder seen in the screenshot belongs to the master project, RSKRX113_Tutorial.
- This folder contains and lists all project source and header files including text files arranged in separate folder structures.
- Folders containing the subprojects, indicated by "(Subproject)", are listed below the File folder.
- Each subproject folder, when expanded, reveals an identical tools and folder structure to that of the master project, RSKRX113_Tutorial.
- By default the RSKRX113_Tutorial project is set as the active project, indicated by the line under the project name.

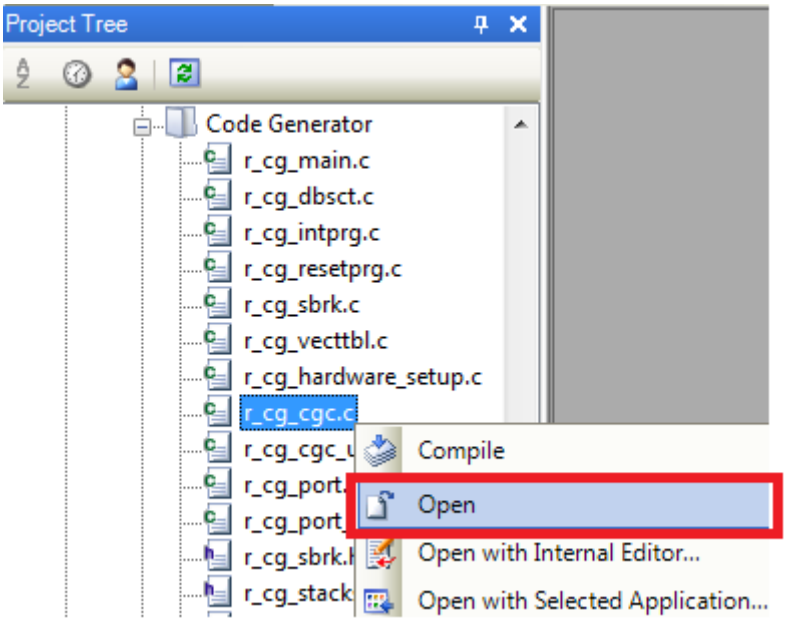
Note: 'Code Generator (Design Tool)' node is shown in the 'Project Tree' and indicates an optional plug-in has been enabled previously.



- To change the active project, right-click on the project/subproject name and select “Set x as Active Project” (x represents the project name).

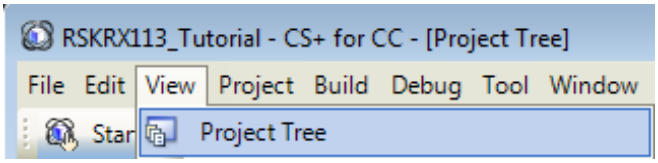
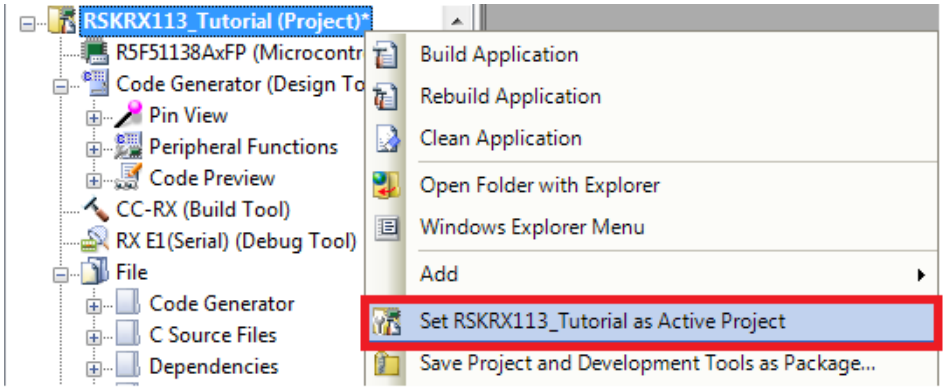
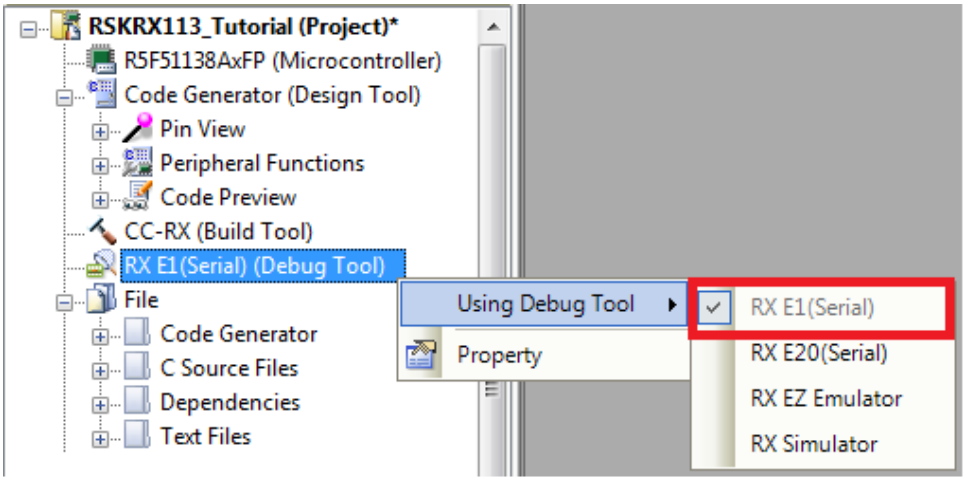


- The File folder contains four subfolders. This structure is common to all projects.
- Some of the source files were generated by Code Generator, which are grouped under the 'Code Generator' folder which itself is listed under the File folder in the Project Tree. These files are prefixed with 'r_cg' to indicate that they were generated by a code generator. All other user-generated source files are contained in the 'C Source Files' folder.
- To open a file for viewing, right-click on the file and select 'Open'. Alternatively, double-click on the file.

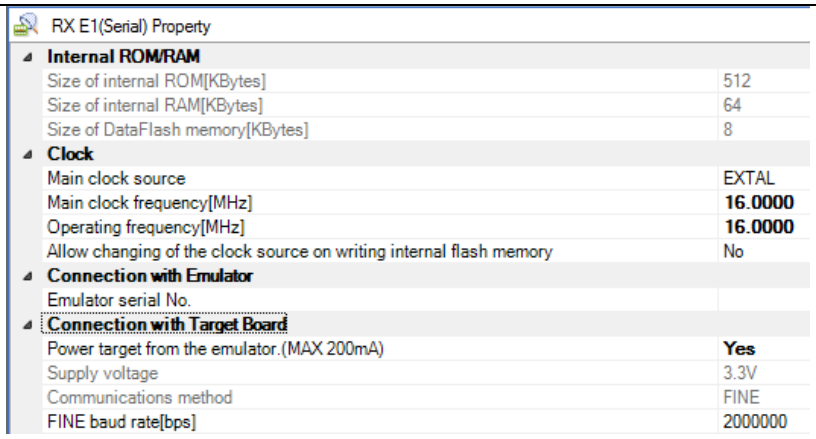


3.3 Configuring the Debug Tool (E1)

Note: The Tutorial sample project's settings are pre-configured. This section is intended to familiarise the user with the debug tool settings for when they create their own project.

• The Project Tree will be displayed on the left-hand pane of CS+. • This can also be invoked from the menu bar [View > Project Tree].	
• If not already active, right-click on the RSKRX113_Tutorial name and select: "Set RSKRX113_Tutorial as Active Project"	
• The opposite screen-shot indicates that the Debug Tool is E1, which is correct for this tutorial.	

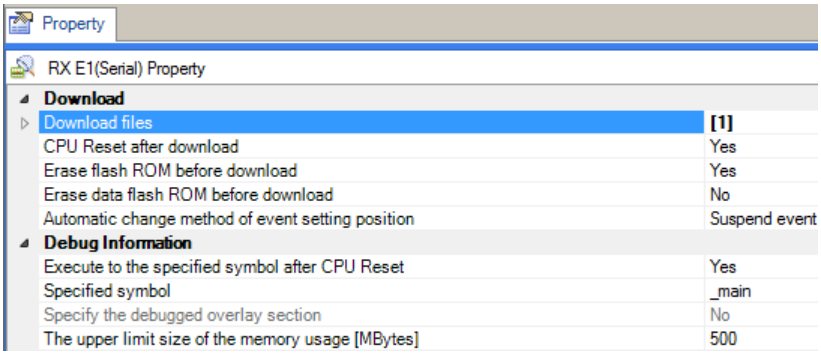
- Right click on RX E1(Serial) (DebugTool).
- Click on Property.
- View the Connect Settings.
- Verify that the settings match the opposite screen-shot.



The project is configured to halt code execution on the first instruction of the main function after programming the micro-controller. To specify another function as the entry point:

- View the Download File Settings of the RX E1's property.
- Change the 'specified symbol' to another available function.
- Ensure to prefix the function name with an underscore ("_").

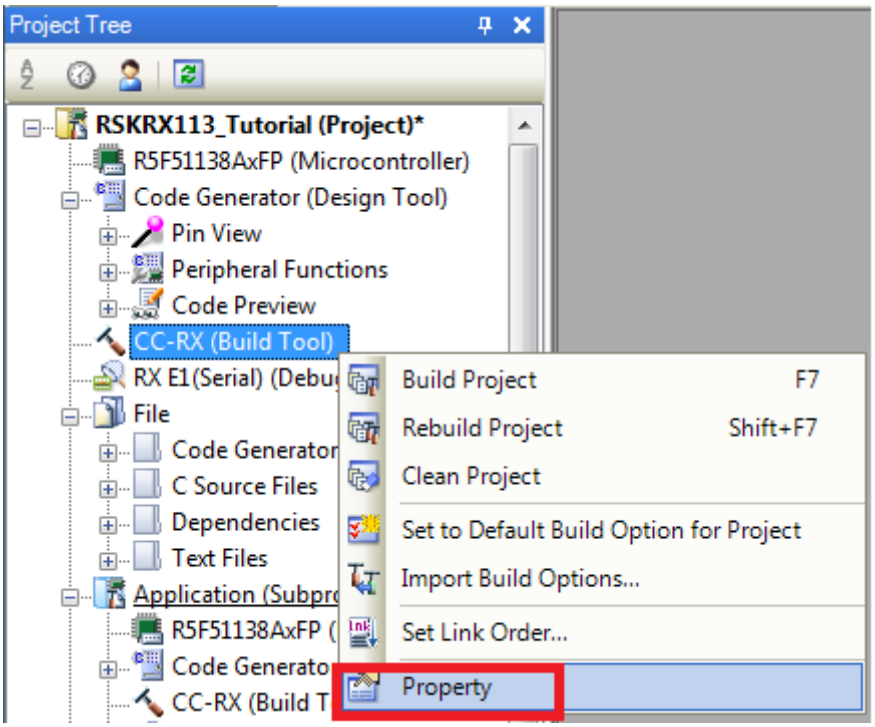
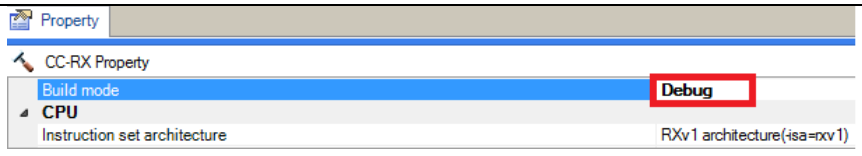
Note: Do not specify an interrupt handler as the entry point.



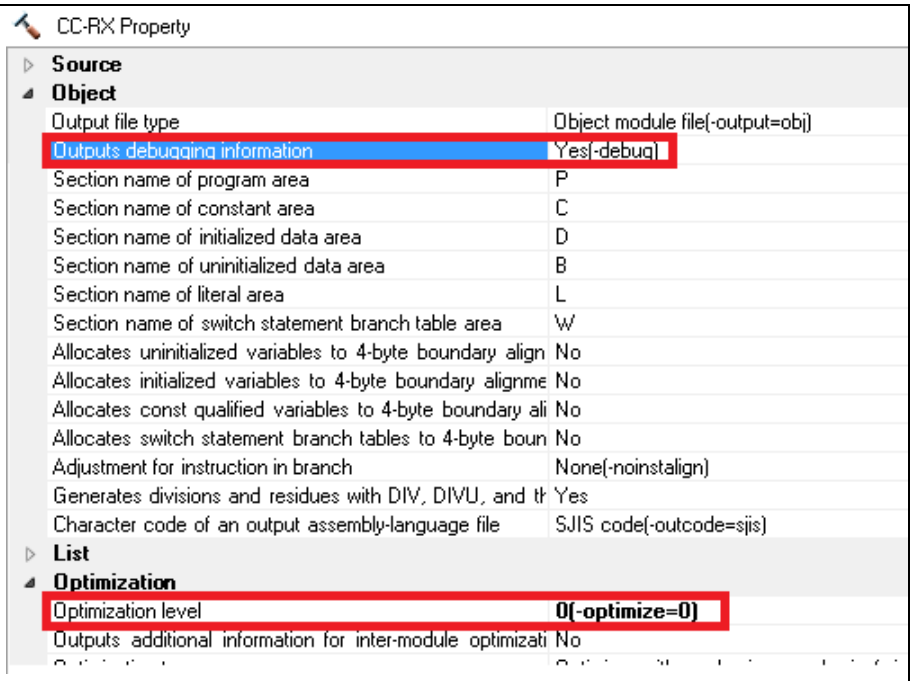
3.4 Build Configuration

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

A common difference between the three builds 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.

• Right-click on CC-RX (Build Tool) from the Project Tree. • Select 'Property'.	
• The Common Options sheet will open by default. • Verify that the Build Mode is set to Debug.	
• Click on the Compile Options sheet to view compiler options.	

- Ensure the 'Outputs debug information' entry is set to 'Yes(-debug)'.
- Ensure the 'Optimization' entry is set to '0(-optimize=0).



4. Building the Tutorial Program

The tutorial project build settings have been pre-configured in the toolchain options. To view the toolchain options double-click on CC-RX(Build Tool) from the Project Tree and select the available tabs. It is important when changing settings to be aware of the current configuration before modifying the settings.

• Review the options on each of the tabs to be aware of the options available. For the purposes of the tutorial, leave all options at default. • When complete, the Property panel can be closed by clicking [x] on the right-hand corner of the Property window.	

4.1 Building the Code

There is a choice of three shortcuts available for building the project:

• Selecting the ‘Build Project’ toolbar button will build all projects listed in the project tree.	
• Pressing [F7]. This is equivalent to pressing the ‘Build Project’ toolbar button.	
• Selecting the ‘Rebuild Project’ toolbar button will rebuild all project files.	
• Selecting the ‘Build & Download’ toolbar button will only build the active project and download the code to the target device after a successful build.	
• Pressing [F6]. This is equivalent to pressing the ‘Build & Download’ toolbar button.	

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.

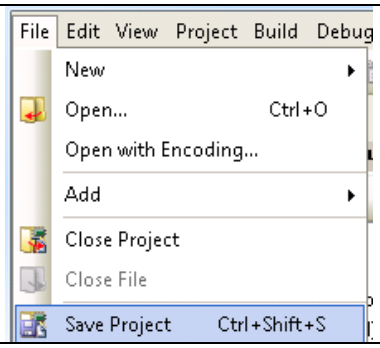
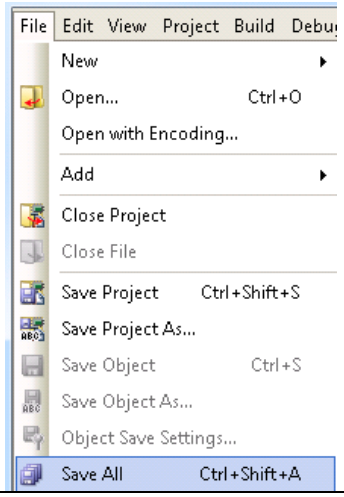
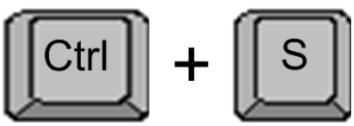
4.2 Connecting the Debugger

For this tutorial the E1 debugger will provide power to the RSK, no external power supply is required.

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 RSK LCD APP V2 display to the RSK board. 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 RSK ensuring that it is plugged into the connector marked ‘E1’.

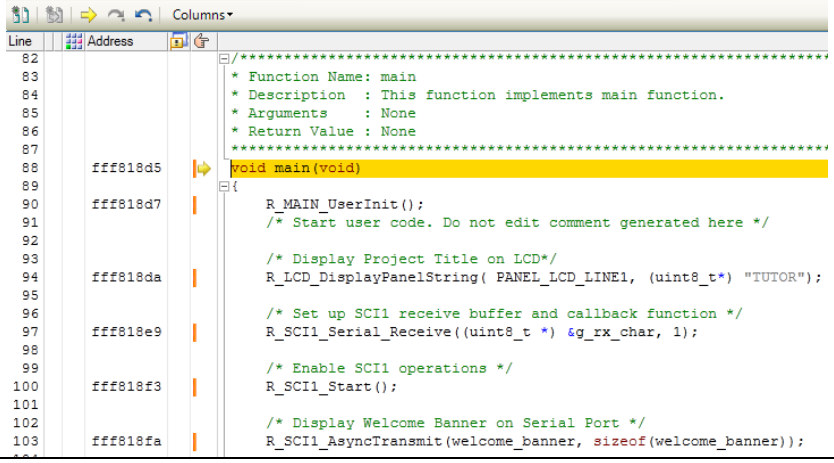
4.3 Saving Project Settings

If you have changed any project settings this is a good time to save the project. • Select ‘File’ | ‘Save Project’.	
If you make any changes to files in CS+ and want to preserve these change, you can save them by: • Select ‘File’ | ‘Save All’.	
You can also save files by clicking the ‘Save’ or ‘Save All’ buttons from the CS+ toolbar.	
In addition files can be saved using the keyboard shortcut [Ctrl + S]:	

5. Downloading and Running the Tutorial

5.1 Downloading the Program Code

Now that the code has been built in CS+ it needs to be downloaded to the RSK.

• Click on the program download button. Alternatively, select Debug from the Menu bar and click on Download.	
• On completion of program download, the debugger and code are ready to be executed. The program counter indicator will point to first line of code inside the main function; this is the program's entry point.	

```
Columns
Line  Address
 82                /*****************************************************************
 83                * Function Name: main
 84                * Description  : This function implements main function.
 85                * Arguments    : None
 86                * Return Value : None
 87                *****************************************************************
 88      fff818d5  void main(void)
 89                {
 90      fff818d7      R_MAIN_UserInit();
 91                    /* Start user code. Do not edit comment generated here */
 92
 93                    /* Display Project Title on LCD*/
 94      fff818da      R_LCD_DisplayPanelString( PANEL_LCD_LINE1, (uint8_t*) "TUTOR");
 95
 96                    /* Set up SCI1 receive buffer and callback function */
 97      fff818e9      R_SCI1_Serial_Receive((uint8_t *) &g_rx_char, 1);
 98
 99                    /* Enable SCI1 operations */
100      fff818f3      R_SCI1_Start();
101
102                    /* Display Welcome Banner on Serial Port */
103      fff818fa      R_SCI1_AsyncTransmit(welcome_banner, sizeof(welcome_banner));
```

Before proceeding, it is necessary to connect to the RSK RL78G1C-USB port to a PC using a USB Type A to mini B cable. The first time this port is connected to the PC an 'Installing Device Driver Software' pop-up will appear and the device driver will be automatically installed. Open Device Manager, the virtual COM port will appear under 'Port (COM & LPT)' as 'RSK USB Serial Port (COMx)', where x is a number. Open a terminal; emulation program, such as HyperTerminal, with the settings 19200, 8, N, 1 on the virtual COM port.

5.2 Running the Tutorial

Once the program has been downloaded onto the RSK device, the program can be executed. Click the 'Go' button or press F5 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. Operating instructions for the program can be found in the file 'description.txt', under the 'Text Files' folder in the CS+ Project Tree.	

6. Reviewing the Tutorial Program

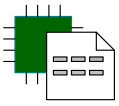
This section will look at each section of the tutorial code and basic debugging functionality in CS+.

6.1 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. Please refer to Section 3.3 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 RX113; press the 'CPU Reset' button on the Debug Toolbar.	

- From the Menu bar select View > Disassemble > Disassemble1. Alternatively, use the Display Disassemble button to open and view the 'source and disassembly'.
- To make the Display Disassemble button available on the toolbar, right-click on the toolbar and select 'View Panels'.



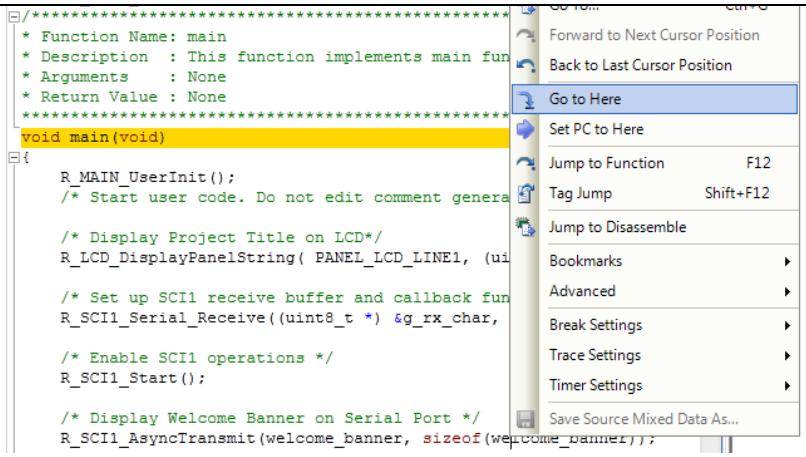
Revert back to the source by clicking on the file containing the function pointed to by the program counter indicator. Alternatively, right click in the Disassemble1 window and click "Jump to Source"

```
88:     void main(void)
        _main:
fff818d5        6040            SUB             #4H,R0
90:         R_MAIN_UserInit();
fff818d7        39dd00          BSR.W           _R_MAIN_UserInit
94:         R_LCD_DisplayPanelString( PANEL_LCD_LINE1, (uint8_t*) "TUTOR");
fff818da        fb221405f8ff    MOV.L           #-0007FAECH,R2
fff818e0        754133          MOV.L           #33H,R1
fff818e3        057cf8ff        BSR.A           _R_LCD_DisplayPanelString
fff818e7        6612            MOV.L           #1H,R2
97:         R_SCI1_Serial_Receive((uint8_t *) &g_rx_char, 1);
fff818e9        fb1200140000    MOV.L           #00001400H,R1
fff818ef        05540500        BSR.A           _R_SCI1_Serial_Receive
100:        R_SCI1_Start();
fff818f3        05d40400        BSR.A           _R_SCI1_Start
fff818f7        75423e          MOV.L           #3EH,R2
```

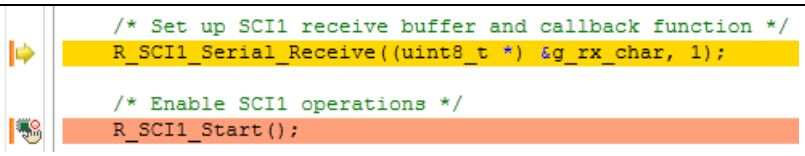
6.2 Main Functions

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

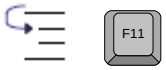
- Right click the 'R_SCI1_Serial_Receive' function call and select 'Go to Here' to execute the program up to this line.
  The 'R_LCD_DisplayPanelString' will write "TUTOR" onto the bottom line of the LCD



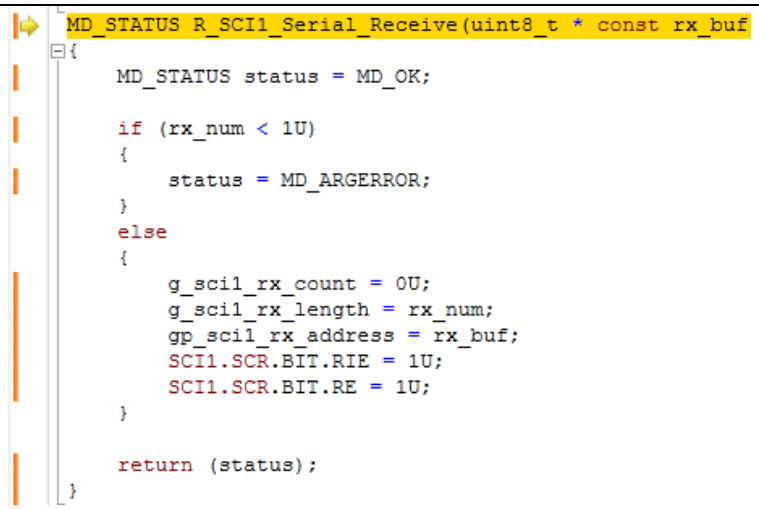
- Set a breakpoint on the 'R_SCI1_Start' function call by clicking on the On-Chip Breakpoint column to the left of the number column.



- Click the 'Step In' button to step into the 'R_SCI1_Serial_Receive' function. Alternatively, press [F11].



- The program counter should now move into the R_SCI1_Serial_Receive function definition. This function is an API function provided by the Code Generator. It sets up the SCI1 interrupt handler code to receive a specified number of bytes into a receive buffer. Once the specified number of bytes has been received, the interrupt handler code calls a callback function as shown later on in this section.
- For full details on how to configure a project using Code Generator refer to the Code Generator Tutorial Manual.
- Press the  button to resume program execution.



- The program counter should come to a halt at the R_SCI1_Start function.
- Step over the function by clicking the 'Step Over' button. Alternatively, press F10.

The R_SCI1_Start function enables the SCI interrupts. The program then proceeds to transmit a welcome banner and enters the main while() loop. The code inside the loop waits for user input from either SCI reception or switch SW3, and then performs an A/D conversion.

```
/* Set up SCI1 receive buffer and callback function */
R_SCI1_Serial_Receive((uint8_t *) &g_rx_char, 1);

/* Enable SCI1 operations */
R_SCI1_Start();

/* Display Welcome Banner on Serial Port */
R_SCI1_AsyncTransmit(welcome_banner, sizeof(welcome_banner));
```

- Locate the function call to 'lcd_display_adc' inside the while loop.
- Set a breakpoint on the 'lcd_display_adc' function call by on the On-Chip Breakpoint column to the left of the number column.

```
while (1U)
{
    uint16_t adc_result;

    /* If the user has requested ADC sample via the serial port
    if (TRUE == g_adc_trigger)
    {
        /* Call the function to perform an ADC conversion */
        adc_result = get_adc();

        /* Display the result on the LCD */
        lcd_display_adc(adc_result);
```

- In the Project Tree pane, locate the file 'r_cg_sci_user.c' and double-click to open the source file. Scroll down to the function r_sci1_callback_receiveend().
- Set a breakpoint on the line of code inside the r_sci1_callback_receiveend function as shown opposite.
- Continue to execute the program by pressing the

button.

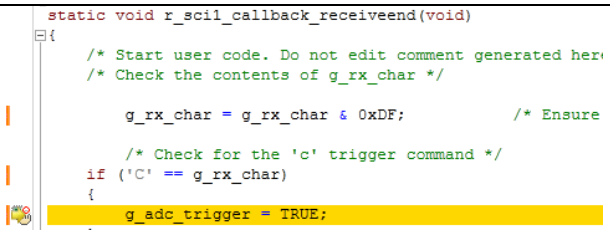
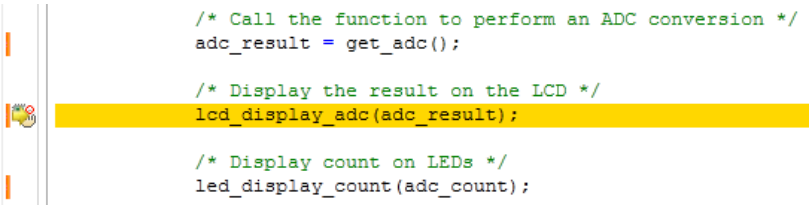
```
static void r_sci1_callback_receiveend(void)
{
    /* Start user code. Do not edit comment generated here */
    /* Check the contents of g_rx_char */

        g_rx_char = g_rx_char & 0xDF;           /* Ensure ASCII

        /* Check for the 'c' trigger command */
    if ('C' == g_rx_char)
    {
        g_adc_trigger = TRUE;
    }

    /* Set up SCI1 receive buffer and callback function again */
    R_SCI1_Serial_Receive((uint8_t *)&g_rx_char, 1);


    /* End user code. Do not edit comment generated here */
}
```

• In the terminal; emulation window, press the 'c' button on the keyboard. • The program will halt at the breakpoint in the r_sci1_callback_receiveend function as shown opposite. Remove the breakpoint by clicking on the breakpoint column. • Continue to execute the program by pressing the  button.	
• The program will halt at the breakpoint in the main while loop. • Remove the breakpoint by clicking on the breakpoint column. Continue to execute the program by pressing the  button.	

The program proceeds to display the result of the A/D conversion on the LCD and in the terminal window. In addition, the running count of A/D conversions performed is displayed in binary form using LEDs 0-3 on the RSK. Adjust the potentiometer and press SW3 on the RSK and an additional A/D conversion will be performed.

• Press the 'Stop' button to halt program execution. • This is the extent of the tutorial code.	

For further details regarding hardware configuration, please refer to the RX Family Software Manual and the RX113 Group 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

7. Additional Information

Technical Support
For details on how to use CS+, refer to the manual available on the DVD or from the web site.

Parts of the sample code provided with the RSKRX113 can be reproduced using the Code Generator plug in tool. Source files and functions generated by Code Generator are prefixed with ‘r_’ and ‘R_’, respectively.

For information about the RX113 Group microcontrollers refer to the RX113 Group Hardware Manual.

For information about the RX assembly language, refer to the RX Family Software Manual.

Technical Contact Details

Please refer to the contact details listed in section 8 of the “*Quick Start Guide*”

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

REVISION HISTORY	RSKRX113 Tutorial Manual

Rev.	Date	Description	
		Page	Summary
1.00	Dec 17, 2014	—	First Edition issued
1.01	Mar 03, 2015	9 - 21	Added a frame

Renesas Starter Kit Manual: Tutorial Manual

Publication Date: Rev. 1.01 Mar 03, 2015

Published by: Renesas Electronics Corporation

Renesas Electronics Corporation

SALES OFFICES http://www.renesas.com

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

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, 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: +49-211-6503-0, Fax: +49-211-6503-1327
Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679
Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999
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: +852-2265-6688, Fax: +852 2886-9022/9044
Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670
Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300
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.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

RX113 Group

Renesas Electronics Corporation

R20UT2757EG0101