RX66T Group

Renesas Starter Kit Sample Code (CS+ for CC-RX)

Introduction

Renesas Starter Kits (RSK) are supplied as complete development systems for the selected microcontroller. The kit includes an evaluation board, portable On-Chip Debugger, and a set of peripheral sample code.

Target Device

RX66T Group

Development Environment

IDE: CS+ V8.12.00

Compiler: CC-RX v3.06.00

Hardware: Renesas Starter Kit for RX66T

Contents <contents header>

1. Installation

The CS+ IDE should already be installed on the user's personal computer (PC). It is assumed that the following software and versions are installed:

·   Renesas CS+ Version V8.12.00

·   CC-RX Tool Chain Version v3.06.00

Create a new folder, for example 'C:\Renesas\Workspace\RSK\RSKRX66T'. Copy the application note zip package 'an-r01an4590ej0200-rx66t-rsk.zip' downloaded from the website to this folder and extract it here.

2. Opening the sample projects

Open CS+ from the Windows Start button by selecting 'All Programs > Renesas Electronics CS+ > CS+ for CC(RL78,RX,RH850)'.
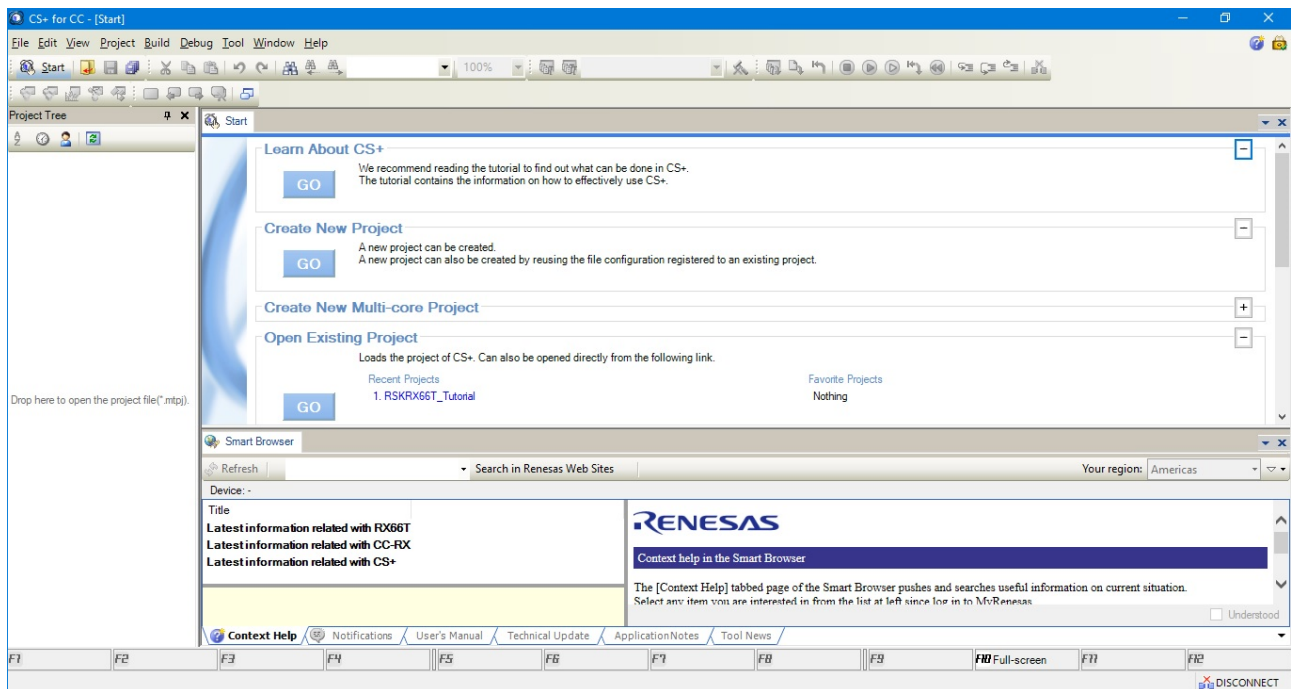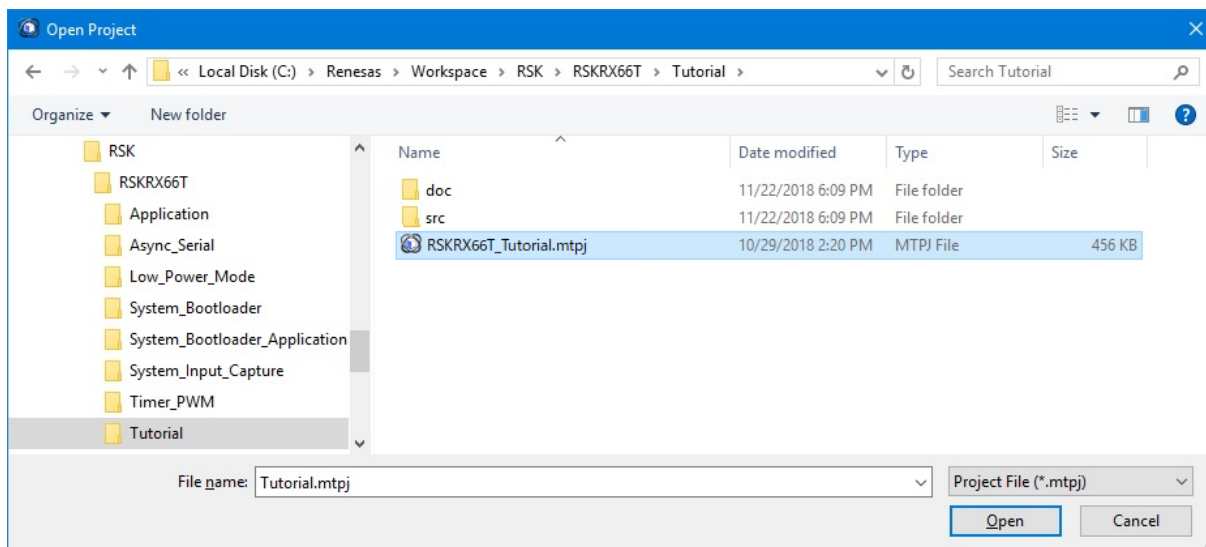

To open the project select 'Project > Open Project…' from the menu bar.
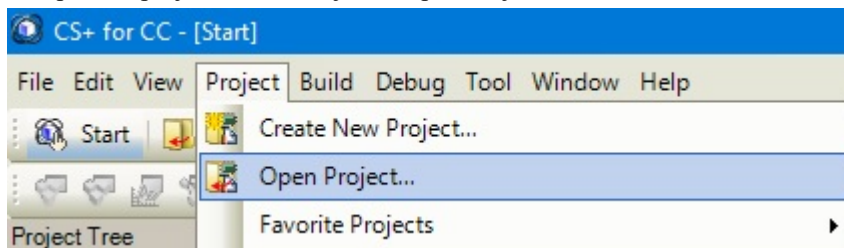
Select Tutorial project as an example.

Navigate to 'C:\Renesas\Workspace\RSK\RSKRX66T\Tutorial' folder and select the file 'RSKRX66T_Tutorial.mtpj'. Click <Open>.

3. Opening the Sample Code and Source Files

Once the Tutorial project is open, the source code and all dependent files can be opened in the editor by expanding the folders in the 'Project Tree' and double clicking the files listed. Files are grouped according to type. A number of separate projects are provided. Each project contains sample code for a specific peripheral.
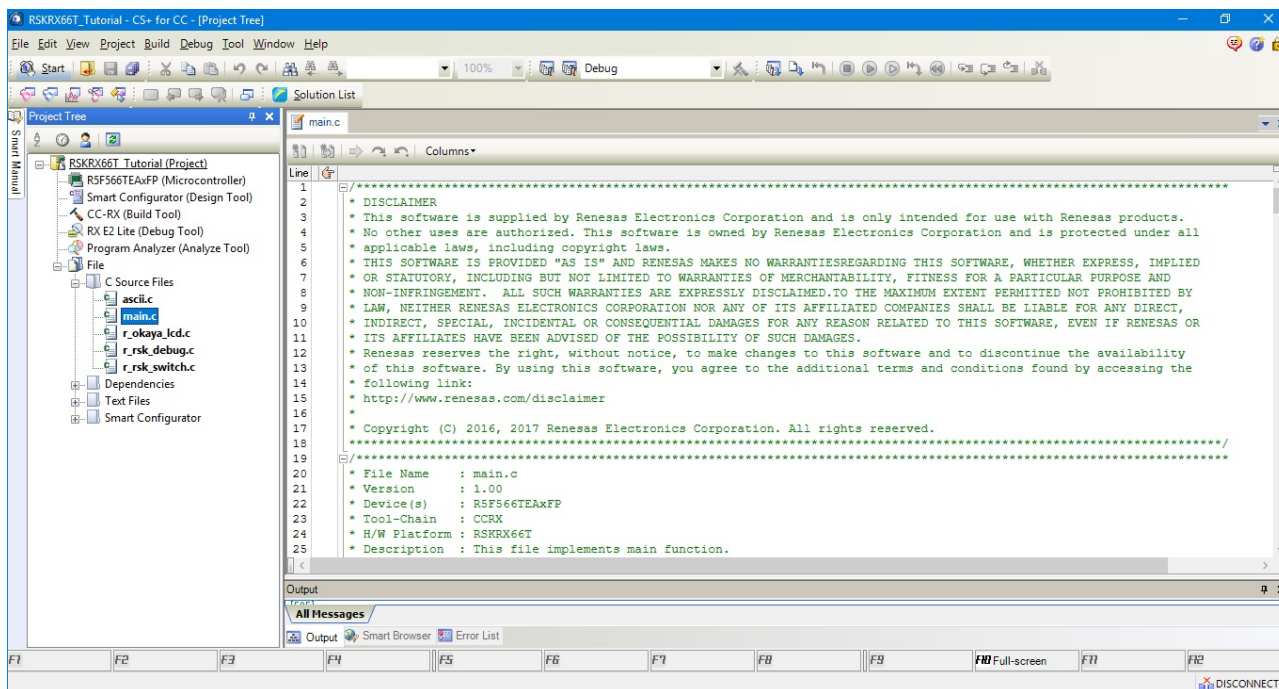
4. Source Code Functionality

Each project is specifically written to run on the appropriate RSK. However, this source code can be useful as an example of peripheral initialization even without the RSK.

Each project includes a description.txt file that describes the functionality of that sample. There will also be a C source file that includes "main" in the name, for example "main.c". This source file includes the C function main().

Website and Support

Renesas Electronics Website
    http://www.renesas.com/

Inquiries
    http://www.renesas.com/contact/

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Nov 20. 2018	-	First edition issued
2.00	Oct.28.2024	-	Fix the "System_BootLoader" sample project. Update development environment. -IDE: CS+ V8.12.00 -Compiler: CC-RX v3.06.00

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. 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the 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.