E1/E20 Emulator for the RH850/E1L and RH850/E1M-S Groups

R20UT3055EJ0400
Rev.4.00
Oct 01, 2015

Release Note (Restrictions on the Emulator when Used with CS+)

This document describes the items listed below. Also refer to E1/E20 Emulator Additional Document for the RH850/E1L and RH850/E1M-S groups for cautionary notes on using the emulator.

- Descriptions of restrictions applicable to the emulator but not to the target device
- Descriptions of restrictions applicable to both the target device and emulator but for which correction is only planned in the case of the emulator

For restrictions on the target device, refer to the following document.

- User's manual for the target MCU
- Document on restrictions for the target MCU

Contents

1. List of Restrictions and Added Specifications

No.	Restrictions or Changes/Additions to Specifications	CS+ Ver.	
		V2.02.00 V3.00.00	V3.01.00 V3.02.00
1	Rewriting of on-chip flash memory (clock monitor)	√	√
2	Downloading to and uploading from external flash memory	√	√
3	Display of download/upload status (progress dialog box)	√	√
4	Memory mapping	√	—
5	Memory function (CPU reset)	√	√
6	Display of memory in dedicated RAM areas used by peripheral modules	√	√
7	Access to memory by using the DMM facility	√	√
8	Display of I/O registers	√	√
9	Effective number of hardware breaks	√	√
10	Write-access and read/write-access break conditions	√	√
11	Breaks while clock settings are being made	√	√
12	Break triggered by reading data	√	√
13	Time measurement	√	—
14	Trace data on branch origins(*)	√	√
15	Out-of-range trace events (1)(*)	√	√
16	Out-of-range trace events (2)(*)	√	√
17	Abnormal exit when out-of-range trace events are set(*)	√	√
18	Using two trace modes at the same time(*)	√	√
19	Debug console	√	√
20	Boot loader projects for multi-core devices	√	—
21	Using the [IOR] panel while the peripheral guard facility is enabled	√	√
22	Data having priority (non-realtime) in tracing(*)	√	√
23	Not assigned (This item was removed because it does not apply to this product.)	—	—
24	Division of load modules	√	√
25	Trace function (trace settings)(*)	√	√
26	State when the size of the main window is maximized	√	√
27	Execution of Python scripts	√	√
28	Debugging information	√	√
29	Source files with the same name	√	√
30	Breakpoint settings for “for” statements and inline functions	√	√
31	Hot plug-in	√	—

√: Applicable, —: Not applicable

Note: When using a device with the trace function.

2. Details of Restrictions and Added Specifications

No. 1 Rewriting of on-chip flash memory (clock monitor)

[Description] The debugger changes settings of the PLL when the flash memory is rewritten*. Thus, rewriting the flash memory raises a possibility of the frequency becoming higher than that currently in use. If the frequency surpasses the upper limit which was set by the clock monitor (CLMA), this prevents rewriting of the flash memory.

[Changes in the PLL settings]

PLL0 is used without frequency division.

Note: Rewriting of flash memory proceeds in response to any of the operations below.

a. Downloading to on-chip flash memory
b. Changes in on-chip flash memory due to operations in the memory panel
c. Setting or cancellation of software breaks
d. Re-execution after a software break is encountered (including stepped execution)

[Resolution] If the change in the clock frequency due to the debugger is a problem, set [Change the clock to flash writing] in the properties panel to [No].

No. 2 Downloading to and uploading from external flash memory

[Description] This emulator does not support downloading to external flash memory.

[Resolution] None

No. 3 Display of download/upload status (progress dialog box)

[Description] If you click on the [Cancel] button on the [Progress Status] dialog box that appears during downloading, closing the dialog box takes some times because so does the cancellation of downloading. Wait until the dialog box is closed.

[Resolution] None

No. 4 Memory mapping

[Description] This emulator does not support mapping to external memory.

[Resolution] None

No. 5 Memory function (CPU reset)

[Description] When [CPU Reset after download] and [Execute to the specified symbol after CPU Reset] are set to [Yes], execution proceeds up to the position of the specified symbol after downloading. However, if the [Download] menu item is selected so that connection to the debugging tool and downloading proceed at the same time, memory values up to the position of the symbol after the reset are not displayed in pink (which indicates realtime RAM monitoring) in the [Memory] panel.

[Resolution] Execute "Connect" and "Download" separately. After execution is stopped and then resumed, memory values up to the position of the symbol are displayed in pink.

No. 6 Display of memory in dedicated RAM areas used by peripheral modules

[Description] The emulator does not support the display of memory in dedicated RAM areas used by peripheral modules.

[Resolution] Access can be gained by directly specifying an address in the watch panel.

Example: *((int*)0xFFC62230)

No. 7 Access to memory by using the DMM facility

[Description] The DMM facility (to modify data during a short break) can only be used for access to the local RAM and global RAM.

[Resolution] None

No. 8 Display of I/O registers

[Description] While display of the contents of I/O registers is supported, the display of individual bits is not.

[Resolution] The values of individual bits in I/O registers can be monitored by registering the bits in the watch panel.

No. 9 Effective number of hardware breaks

[Description] Only a total of up to 12 hardware breaks can be set for the combination of CPU1 and PCU.

[Resolution] None

No. 10 Write-access and read/write-access break conditions

[Description] In cases where a write-access or read/write-access break condition is set, the write cycle of read-modify-write (RMW) generated by a bit-manipulation instruction does not lead to a break.

[Resolution] None

No. 11 Breaks while clock settings are being made

[Description] The flash memory cannot be programmed if a break occurs while clock settings are being made.

[Resolution] When performing operations below in a break state while the clock was being set, set [Change the clock to flash writing] in the properties panel to [No].

a. Any operation that involves programming of the flash memory (e.g. re-downloading)

b. Setting or deleting software breakpoints

Also, do not set software breakpoints within the clock-setting routine.

No. 12 Break triggered by reading data

[Description] Read-access break conditions cannot be set for SYSCALL instructions.

[Resolution] None

No. 13 Time measurement

[Description] Point-to-point time measurement is not possible.

[Resolution] Measure execution time up to the point of stopping the program.

No. 14 Trace data on branch origins

[Description] Branch trace information being displayed consists of trace data on the branch origins and destinations, complemented by the source code between the origins and destinations. When you use trace start and trace end events, trace data may not be acquired because branch information that is necessary for complementing the source code is not acquired. This depends on the locations of the trace start and trace end events.

[Resolution] None.

Trace start and trace end events must be set on lines that include branch instructions.

No. 15 Out-of-range trace events (1)

[Description] For an out-of-range trace event, be sure to specify the start and end addresses.

[Resolution] None

No. 16 Out-of-range trace events (2)

[Description] Only one section is specifiable for an out-of-range trace event.

[Resolution] None

No. 17 Abnormal exit when out-of-range trace events are set

[Description] If connection to the debugging tool is lost for some reason (e.g. by disconnecting the USB) when an out-of-range trace event is set, re-connecting the emulator does not restore that out-of-range trace event. It is also not possible to set new out-of-range trace events in this state.

[Resolution] After connection to the debugging tool has been lost, close the project and open it again, then reconnect the debugging tool and download the program. You can now set new out-of-range trace events.

No. 18 Using two trace modes at the same time

[Description] “Section trace” and “Range Out trace” cannot be used at the same time.

[Resolution] None

No. 19 Debug console

[Description] This emulator does not support the debug console.

[Resolution] None

No. 20 Boot loader projects for multi-core devices

[Description] Boot loader projects are not supported for multi-core devices.

[Resolution] None

No. 21 Using the [IOR] panel while the peripheral guard facility is enabled

[Description] Opening the [IOR] panel while the peripheral guard facility is enabled leads to an error because the [IOR] panel also displays registers that have been protected by the peripheral guard facility. Do not open the [IOR] panel while the peripheral guard facility is enabled.

[Resolution] To view data on registers other than those protected by the peripheral guard facility, use the [Watch] panel after adding the registers to the panel.

No. 22 Data having priority (non-realtime) in tracing

[Description] When priority in tracing is given to data, the function to stop tracing when the trace memory becomes full (trace-full stop function) is not usable.

[Resolution] To use the trace-full stop function, give priority in tracing to speed (realtime).

No. 24 Division of load modules

[Description] The restrictions below apply when the CC-RH compiler is used to generate split load modules from a program.

a. Source-level debugging becomes impossible

b. The second and subsequent output files are not automatically registered with the debugging tool.

[Resolution] Do not divide the program up into separate load modules.

No. 25 Trace function (trace settings)

[Description] Even though trace settings are displayed in the properties area under the tabbed page for debug tool settings, the trace function cannot be used if the device does not include its own trace function. Do not change the related items in the properties area.

[Resolution] None

No. 26 State when the size of the main window is maximized

[Description] When the list in the combo box used for switching between cores on the status bar is being displayed while the size of the main window is maximized, part of the list is hidden behind the task bar and thus cannot be selected.

[Resolution] Set the task bar to “Hide automatically” or set the location of the task bar as [Right], [Left], or [Upper].

No. 27 Execution of Python scripts

[Description] Commands other than those listed below cannot be used when a Python script is executed with a hook process setting.

debugger.Register.GetValue
debugger.Register.SetValue
debugger.Memory.GetValue
debugger.Memory.SetValue

[Resolution] Use a hook facility that can be registered by the hook function in the python console.

No. 28 Debugging information

Some restrictions apply to the debugging information generated by the CC-RH compiler.

Problems that arise due to these restrictions are listed below.

Note, however, that the reason for these problems is a difference between the debugging information generated by the compiler and the actual code. The result of executing the code generated by the CC-RH compiler is correct.

No. 28-1 Source-level stepping

[Description 1] During source-level stepping, the debugger may appear to be executing instructions that are not supposed to be executed.

[Example 1-1] In the example below, execution stops at the position marked (*2) after completing the for loop starting at (*1) and branches to the next line depending on the value of "i".

Under some conditions, however, the PC (indicated by an arrow) will appear to have moved to the position marked (*3) regardless of the value of "i" at (*2).

```
void main()
{
    int i = 0;
    int j = 0;
    for (j = 0; j < 10; j++) {  <-(*1)
        i = atoi("100");
    }
    if (i != 100) {            <-(*2)
        i = atoi("100");       <-(*3)
    }
    return;
}
```

Conditions:

a. There is a conditional or loop control statement (e.g. if, for, switch) at the position marked (*2).

b. The statement immediately before (*2) is any of the following.

- A conditional or loop control statement
- A label, goto, or return statement
- A statement including a ternary, logical, or NOT operator

[Resolution] None

[Description 2] During source-level stepping, the debugger may appear to be executing instructions that are not supposed to be executed.

[Example 1-2] In the example below, execution branches to the assignment for the third case statement (*2) from the position marked (*1) when function GetCount() is called for the first time. Under some conditions, however, the PC (indicated by an arrow) will appear to have moved to the position marked (*3), the statement for the case immediately before (*2).

```
  enum Count { ZERO, ONE, TWO };

  enum Count GetCount()
  {
     static enum Count value = ZERO;

     switch (value) {      <-(*1)
case TWO:
   value = ZERO;
   break;
case ONE:
   value = TWO;   <-(*3)
   break;
case ZERO:
default:
   value = ONE;   <-(*2)
   break;
     }
     return value;
  }
```

Conditions:

The instruction at the branch destination

a. acquires the value of a constant, or

b. determines the address of a variable.

[Resolution] None

No. 28-2 Display of information on variables

[Description] If two or more variables defined in a function have the same name, the values of variables that can be viewed when the program has stopped may differ from the expected values.

Whether this phenomenon arises depends on the optimization level* selected during the process of compilation.

Note: The optimization level can be set via [Build Tool] - [Common Options] - [Frequently Used Options (Link)].

[Example] In the example below, char-type variable "a" is in the innermost scope at (*1) and int-type variable "a" is in the innermost scope at (*2). Under some conditions, however, only the value of one of the variables will be visible at (*1) and (*2).

```
void main()
{
    int a = 100;
    {
        char a = 'A';
        a++;          <-(*1)
    }
    a++;              <-(*2)
}
```

- Display of (*1) in the [Watch] panel

```
     "a"       'A' (0x41) "signed char" "0xfefb1004" // Expected value
  or "a"  100 (0x00000064) "int"      "0xfefb1000"
```

- Display of (*2) in the [Watch] panel

```
     "a"       'B' (0x42) "signed char" "0xfefb1004"
  or "a"  100 (0x00000064) "int"      "0xfefb1000" // Expected value
```

Condition:

Optimization other than for debugging at the time of compilation.

[Resolution] Select [Optimize for Debugging] as the optimization level before compilation.

No. 29 Source files with the same name

[Description] When two or more files with the same name exist in a load module being debugged, line addresses are not displayed correctly in the editor. Setting of events also does not work correctly.

Example:

C:¥Work¥CS+¥ProjA¥ProjA.mtpj¥Src¥main.c -> A.abs

C:¥Work¥CS+¥ProjB¥ProjB.mtpj¥Src¥main.c -> B.abs

This is a case where the above two load modules are being debugged simultaneously.

Note: Although multiple load modules are used in the above example, this restriction is also applicable to cases where a single load module is in use.

Condition: The relative paths to the files from the compilation directory are the same (including the filenames).

Building by CS+

Project file directory (*.mtpj) = compilation directory

Note: The filename extension for files in subprojects is *.mtsp.

Building by using a makefile

Current directory = compilation directory

[Resolution] Source files with the same name can be distinguished in either of the following ways.

a. Change the configuration of the folders so that the relative paths to the files from the compilation directory differ.

Before:	ProjA¥Src¥main.c
	ProjB¥Src¥main.c
After:	ProjA¥SrcA¥main.c
	ProjB¥SrcB¥main.c

With this change, the relative paths will be as follows.

"SrcA¥main.c"
"SrcB¥main.c"

b. Change the names of the source files so that all of the files to be debugged have unique names.

Before:	ProjA¥Src¥main.c
	ProjB¥Src¥main.c
After:	ProjA¥Src¥mainA.c
	ProjB¥Src¥mainB.c

No. 30 Breakpoint settings for “for” statements and inline functions

[Description] When the C source code for a program includes statements of the types listed below, the instructions corresponding to a single line of the source code will be at multiple points. However, the editor only indicates the address of one of the instructions. When a breakpoint is set for a line of a listed type, the break will only be generated at the address indicated by the editor.

a. Inline functions*

b. Template functions

c. The first lines of for and do-while statements

Note: This includes functions which are inline-expanded by optimization.

[Resolution] None

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; anti-crime 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.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

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

California Eastern Laboratories, Inc.
4590 Patrick Henry Drive, Santa Clara, California 95054-1817, U.S.A.
Tel: +1-408-919-2500, Fax: +1-408-988-0279

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 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

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 1207, 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 India Pvt. Ltd.
No.777C, 100 Feet Road, HALII Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 5.0