CS+ RH850 Compiler CC-RH V1.02.00 Release Note

R20UT3315EJ0102
Rev.1.02
June. 30, 2015

Thank you for using the CS+ integrated development environment.

This document describes the restrictions and points for caution. Read this document before using the product.

Contents

Chapter 1. Target Devices 2

Chapter 2. User's Manuals 3

Chapter 3. Keywords When Uninstalling the Product 4

Chapter 4. Changes 5

4.1 Changes to the CC-RH compiler 5

4.1.1 Support for the G3MH core 5

4.1.2 Added an option to designate the precision of double-type variables 5

4.1.3 Added an option to select the rounding mode for floating point types 6

4.1.4 Added an option to change a default allocation section 6

4.1.5 Added an option to designate aligning the addresses where functions start with four-byte boundaries 7

4.1.6 Output of the addresses of structure members to a link map file 7

4.1.7 Added predefined macros 7

4.1.8 Removal of restrictions 8

4.1.9 Changes to other specification of compiler 8

4.1.10 Changes to other specifications of assembler 8

Chapter 5. Points for Caution 11

5.1 FE level exceptions 11

5.2 STARTOF, SIZEOF operators 11

Chapter 1. Target Devices

The target devices supported by the CC-RL compiler are listed on the Website.

Please see the URL below.

CS+ Product Page:

http://www.renesas.com/cs+

Chapter 2. User's Manuals

Please read the following user's manuals along with this document.

Manual Name	Document Number
CC-RH V1.02.00 RH850 Family C Compiler Package User's Manual: Compiler	R20UT3280EJ0100
CS+ Integrated Development Environment User's Manual: CC-RH Build Tool Operation	R20UT3283EJ0100

Chapter 3. Keywords When Uninstalling the Product

There are two ways to uninstall this product.

- Use the integrated uninstaller (uninstalls all CS+ components)
- Use the Windows uninstaller (only uninstalls this product)

To use the Windows uninstaller, select “CS+ CC-RH V1.02.00” from “Programs and Features” of the control panel.

Chapter 4. Changes

This chapter describes changes to the CC-RH compiler.

4.1 Changes to the CC-RH compiler

This section describes changes to the CC-RH compiler from V1.01.00 to V1.02.00.

4.1.1 Support for the G3MH core

The specifications have been changed so that designating *g3mh* as the argument of -Xcpu option is now possible.

When -Xcpu=*g3mh* option is designated, the instructions at the machine language level are rearranged for the G3MH core. This might enhance the performance in execution for target devices incorporating a G3MH core.

The specification of interrupts for the #pragma interrupt directive has also been changed to support the G3MH core. The interrupts that are specifiable differs with the -Xcpu option.

- *FPINT* may be designated for *priority* of #pragma interrupt when the -Xcpu=*g3mh* option has been designated.
- An error (E0523005) occurs if *FPP* or *FPI* is designated for *priority* of #pragma interrupt when the -Xcpu=*g3mh* option has been designated.
- An error (E0523005) occurs if *FPINT* is designated for *priority* of #pragma interrupt when -Xcpu=*g3mh* option has not designated.

4.1.2 Added an option to designate the precision of double-type variables

A -Xdbl_size option for designating the precision of variables of the double and long-double types has been added. The available values are *4* and *8*. An error occurs if another value is designated.

When -Xdbl_size =*4* is designated, variables of the double and long-double types are handled as single-precision floating point (4-byte). On the other hand, when -Xdbl_size =*8* is designated, they are handled as double-precision floating point (8-byte). The default behavior is handling as double-precision floating point (8-byte).

4.1.3 Added an option to select the rounding mode for floating point types

A -Xround option for designating the rounding mode for floating point constants has been added. The available values are *nearest* and *zero*. An error occurs if another value is designated.

When -Xround=*nearest* is designated, floating point constants are rounded to the nearest representable values. On the other hand, when -Xround=*zero* is designated, floating point constants are rounded in the direction of zero. When this option is omitted, floating point constants are rounded to the nearest representable values.

4.1.4 Added an option to change a default allocation section

A -Xsection option for changing the allocation of sections from the default allocation has been added. The default sections for the allocation of variables are as follows.

- Variables without initial values: .bss
- Variables with initial values: .data
- const variables: .const

In V1.01.00 and earlier versions, changing the default sections for allocation required explicit #pragma section directives in the source code. However, this can be changed by designating the -Xsection option in V1.02.00 and subsequent versions. Designate this in the format, -Xsection=*string*=*value* [,*string*=*value*]. The relations between strings and values are as follows.

string	*value*	Sections after the change		
		Variables without initial values	Variables with initial values	const variables
data	*r0_disp16*	.zbss	.zdata	-
	r0_disp23	.zbss23	.zdata23	-
	ep_disp16	.ebss	.edata	-
	ep_disp23	.ebss23	.edata23	-
	gp_disp16	.sbss	.sdata	-
	gp_disp23	.sbss23	.sdata23	-
const	*zconst*	-	-	.zconst
	zconst23	-	-	.zconst23

When a section is also changed by a #pragma section directive, the #pragma section directive takes priority. When *ep_disp16* or *ep_disp23* is designated for *value*, compilation proceeds as if -Xep=*fix* was designated. Using -Xsection at the same time as -Omap or -Osmap leads to an error.

4.1.5 Added an option to designate aligning the addresses where functions start with four-byte boundaries

An -Xalign4 option for designating the alignment of the addresses where functions start with four-byte boundaries has been added.

When -Xalign4 option is designated, the addresses where functions start are aligned with four-byte boundaries. This option may enhance performance in execution. When this option is omitted, the addresses where functions start are aligned with two-byte boundaries.

4.1.6 Output of the addresses of structure members to a link map file

struct has been added as a value for the linker's -show option.

When the -show=*struct* option is designated, the "Symbol List" of the link map file indicates the addresses of the members of structures and unions.

4.1.7 Added predefined macros

The predefined macros mentioned below have been newly added. They are usable in compiling, such as for conditional compilation with the use of #ifdef text.

Macro name	Definition
__DBL4	This value is not set. (defined when *4* is specified by the -Xdbl_size option).
__DOUBLE_IS_32BITS__	This value is not set. (defined when *4* is specified by the -Xdbl_size option).
_RON	This value is not set. (defined when *nearest* is specified by the -Xround option).
_ROZ	This value is not set. (defined when *zero* is specified by the -Xround option).

The following defined macros are changed their definition, since -Xdbl_size option is added. The changed definitions are written by red font.

Macro name	Definition
__DBL8	This value is not set. (defined only when *8* is specified by the -Xdbl_size option or -Xdbl_size option is not specified.)
__DOUBLE_IS_64BITS__	This value is not set. (defined only when *8* is specified by the -Xdbl_size option or -Xdbl_size option is not specified.)

The following predefined macros are also supported for use in assembling.

Macro name	Definition
__RENESAS__	This value is defined as 1.
__ASRH__	This value is defined as 1.
__ASRH	This value is defined as 1.
__RH850__	This value is defined as 1.
__RH850	This value is defined as 1.

4.1.8 Removal of restrictions

One restriction below has been removed.

- Point to note regarding the use of both judgment of a match and greater or less than for variables (No.1)

4.1.9 Changes to other specification of compiler

(a) Change to the specification of -Xasm_option with the *-Xasm_far_jump* option

The specification has been changed so that when -Xasm_option=*Xasm_far_jump* option is designated, the *-Xasm_far_jump* assembler option for assembly source files which were generated by the compiler is invalidated.

4.1.10 Changes to other specifications of assembler

(a) Change to the specification when a relative label is designated as the operand for a JMP instruction

The specification has been changed so that designating a relative label as the operand for a JMP instruction in an assembly source file leads to an error.

(b) Change to the specification of Macro directives

The specification of Macro directives (.macro, .local, .rept, .irp) has been changed partially.

- .macro

When the macro definitions are nested by the .macro directive, it will cause an error in V1.02.00. And the plural same names as formal parameter in macro body are specified , it will also cause an error.

- .local

When the label is defined by .local directive in any field except for macro body, .rept-.endm block,.irp-.endm block, or inline_asm function, it will cause an error in V1.02.00.

- .rept, .irp

When the macro is defined within .rept-.endm block or .irp-.endm block, it will cause an error in V1.02.00.

(c) Change to the specification of Macro operator

When the concatenation "~" concatenates except for the alphanumeric characters and special characters to another within a macro body, it will cause an error in V1.02.00.

And when a dollar symbol ($) is specified as an actual argument for a macro call, it will also cause an error.

(d) Change to the specification of Area reservation directive

The specification of .ds directive has been changed. In V1.01.00 and earlier versions, if a value in the operand is not parenthesized, the assembler assumes that an initial value is specified. In V1.02.00, it assumes that a size is specified .

[example]

V1.01.00:

```
 .ds   3          ; Reserve 1-byte area initialized by 3.
```

V1.02.00:

```
 .ds   3          ; Reserve 3-bytes area initialized by 0.
```

(e) Change to the specification of Data definition directives (.db, .db2, .db4, .db8, .dhw, .dw, .ddw)

The specification of data definition directives(.db, .db2, .db4, .db8, .dhw, .dw, .ddw) has been changed. In V1.01.00 and earlier versions, if a value in the operand is parenthesized, the assembler assumes that a size is specified. In V1.02.00, it assumes that an initialize value is specified.

[example]

V1.01.00:

```
 .db   (3)        ; Initialize 3-byte area initialized by 0.
```

V1.02.00:

```
 .db   (3)        ; Initialize 1-byte area initialized by 3.
```

(f) Change to the specification of Data definition directives (.db8, .ddw)

The specification of data definition directives(.db8, .ddw) has been changed. When the floating-point number is specified in the operand field, it will cause an error in V1.02.00.

[example]

```
 .ddw   0.1       ; It will cause E0551230 error in V1.02.00.
```

(g) Change to the rule of section name

When the beginning of section name is described by numeric letter, it will cause an error in V1.02.00.

[example]

```
 .section   "123.data", data       ; It will cause E0551225 error in V1.02.00.
```

(h) Support for ld.dw/st.dw instruction

ld.dw and st.dw, 8-byte access instruction, have been supported.

(i) Not expanding the instructions in inline assembler

The assembler instruction which is described in inline assembler is not expanded.

[example]

```
#pragma inline_asm func
static void func(void)
{
    ld.w 0x40000[r0], r10
}
```

The assembler instruction is expanded as follows in V1.01.00.

```
movhi       0x4,r0,r1
ld.w        0x0[r1],r10
```

But it is assembled as follows in V1.02.00.

```
ld.w        0x0[r0],r10
```

Chapter 5. Points for Caution

This section describes points for caution regarding CC-RH.

5.1 FE level exceptions

When a #pragma interrupt directive is defined for the interrupt function for an FE level exception from which return or recovery is not possible, i.e. when FENMI or SYSERR is designated as the priority, the code for the exit from the interrupt function is not output. This should be handled correctly in the program.

5.2 STARTOF, SIZEOF operators

No error occurs when a non-existent section is designated as the argument of the section aggregation operators “STARTOF” and “SIZEOF” in assembler. Operators with such settings are simply ignored.

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したものですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、
   各品質水準は、以下に示す用途に製品が使用されることを意図しております。
   標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、
   家電、工作機械、パーソナル機器、産業用ロボット等
   高品質水準：輸送機器（自動車、電車、船舶等）、交通用信号機器、
   防災・防犯装置、各種安全装置等
   当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいいます。
注2. 本資料において使用されている「当社製品」とは、注１において定義された当社の開発、製造製品をいいます。

ルネサス エレクトロニクス株式会社

■営業お問合せ窓口 http://www.renesas.com

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社　〒100-0004　千代田区大手町2-6-2（日本ビル）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：http://japan.renesas.com/contact/