

この度は、統合開発環境 CS+をご使用いただきまして、誠にありがとうございます。

この添付資料では、本製品をお使いいただく上での制限事項および注意事項等を記載しております。ご使用の前に、必ずお読みくださいますようお願い申し上げます。

目次

第 1 章	対象デバイスについて	2
第 2 章	ユーザズ・マニュアルについて.....	3
第 3 章	アンインストール時の選択キーワード	4
第 4 章	変更点.....	5
4.1	CC-RH の変更点	5
4.1.1	<i>standard/professional</i> 版への移行	5
4.1.2	<i>G3KH</i> コア対応.....	5
4.1.3	<i>-Xalign4</i> オプションの拡張.....	5
4.1.4	<i>CRC</i> 演算機能	6
4.1.5	<i>-Xpatch</i> オプションの拡張.....	6
4.1.6	注意事項の改修	7
4.1.7	その他変更・改善.....	7
4.1.8	<i>MISRA-C:2004</i> ルールによるソース・チェック機能 【professional】	7
4.1.9	<i>MISRA-C:2012</i> ルールによるソース・チェック機能 【professional】	8
4.1.10	スタック破壊検出機能 【professional】	8
第 5 章	注意事項	9
5.1	FE レベル例外	9

第1章 対象デバイスについて

CC-RH がサポートする対象デバイスに関しては、WEB サイトに掲載しています。

こちらをご覧ください。

CS+製品ページ：

<http://japan.renesas.com/cs+>

第2章 ユーザーズ・マニュアルについて

本製品に対応したユーザーズ・マニュアルは、次のようになります。本文書と合わせてお読みください。

マニュアル名	資料番号
CC-RH コンパイラ ユーザーズマニュアル	R20UT3516JJ0100
CS+ 統合開発環境 ユーザーズマニュアル CC-RH ビルド・ツール操作編	R20UT3283JJ0101

第3章 アンインストール時の選択キーワード

本製品をアンインストールする場合は、2つの方法があります。

- ・ 統合アンインストーラを使用する(CS+自体をアンインストールする)
- ・ 個別にアンインストールする(本製品のみをアンインストールする)

個別にアンインストールを行なう場合、コントロールパネルの

- ・ 「プログラムと機能」

から、「CS+ CC-RH V1.03.00」を選択してください。

第4章 変更点

本章では、CC-RH の変更点について説明します。

4.1 CC-RHの変更点

CC-RH V1.02.00 から V1.03.00 への主な変更点を説明します。

4.1.1 standard/professional版への移行

CC-RH を以下の 2 つのグレードに分けました。

- standard 版
- professional 版

以降、professional 版のライセンス登録時のみ使用できる機能は **【professional】** と明記します。

4.1.2 G3KHコア対応

-Xcpu オプションの引数に *g3kh* を指定出来るようにしました。

-Xcpu=*g3kh* オプションを指定すると、G3KH の命令セットに応じたコードを生成します。浮動小数点演算に対して単精度演算は浮動小数点演算命令を生成し、倍精度演算はランタイム関数呼び出し命令を生成します。

4.1.3 -Xalign4オプションの拡張

-Xalign4 オプションを拡張しました。V1.02.00 では-Xalign4 オプションを指定した場合、関数の先頭アドレスのアライメントのみを 4 にします。V1.03.00 では-Xalign4[=*mode*] の形式で指定します。*mode* には以下のいずれかが指定可能です。省略した場合、関数の先頭アドレスのアライメントを 4 にします。アライメントを 4 にすることにより、ROM サイズは増加しますが実行速度の向上が期待できます。

<i>mode</i>	内容
<i>function</i>	関数の先頭アドレスのアライメントを4にします。
<i>loop</i>	関数の先頭と全てのループの先頭アドレスのアライメントを4にします。
<i>innermostloop</i>	関数の先頭と最内側ループの先頭アドレスのアライメントを4にします。
<i>all</i>	関数の先頭と全ての分岐先アドレスのアライメントを4にします。

4.1.4 CRC演算機能

CRC(Cyclic Redundancy Check) 演算を行い、演算結果を特定アドレスに出力する-CRC リンカオプションを追加しました。演算方法には以下のいずれかが指定可能です。省略した場合、32-ETHERNETを指定したものとして演算を行います。

演算方法	内容
CCITT	CRC-16-CCITTでMSB First、初期値0xFFFF、XOR反転による演算結果を得ることができます。
16-CCITT-MSB	CRC-16-CCITTでMSB First によるによる演算結果を得ることができます。
16-CCITT-MSB-LITTLE-4	入力をLITTLEエンディアン4バイト単位としCRC-16-CCITTでMSB Firstによる演算結果を得ることができます。
16-CCITT-MSB-LITTLE-2	入力をLITTLEエンディアン2バイト単位としCRC-16-CCITTでMSB Firstによる演算結果を得ることができます。
16-CCITT-LSB	CRC-16-CCITTでLSB Firstによる演算結果を得ることができます。
16	CRC-16でLSB Firstによる演算結果を得ることができます。
SENT-MSB	入力をLITTLEエンディアン1バイト中下位4bit単位としSENT準拠でMSB Firstによる演算結果を得ることができます。
32-ETHERNET	CRC-32-ETHERNETによる演算結果を得ることができます。演算結果は初期値 0xFFFFFFFF、XOR反転、ビットリバースされています。

4.1.5 -Xpatchオプションの拡張

G3M コアを搭載したマイコン向けに-Xpatch オプションを拡張し、引数に *syncp* 及び *switch* を指定できるようにしました。-Xcpu=g3m オプション指定時に-Xpatch=*syncp* オプションを指定すると、以下の割り込み関数の先頭に syncp 命令を挿入します。

- #pragma interrupt 指令の「割り込み仕様」に priority=SYSERR/FPI/FENMI/FEINT/EIINT_PRIORITYX (Xは0から15) のいずれかを指定した関数。または priority も channel も指定していない割り込み関数

-Xcpu=g3m オプション指定時に-Xpatch=*switch* オプションを指定すると、switch 文に対して switch 命令を使用するコードを生成する可能性があります。本オプションを指定しない場合は switch 命令を使用しません。

4.1.6 注意事項の改修

以下 5 件の注意事項を改修しました。注意事項の詳細につきましてはツールニュースをご確認ください。

- アセンブリ命令のオペランドに即値を指定した場合の注意事項 (No.2)
- ラベルの定義と参照が同一アセンブル単位にある場合の注意事項 (No.3)
- アセンブリ命令のベースレジスタ省略時の注意事項 (No.4)
- 仕様外のアセンブリ記述時の注意事項 (No.5)
- 外部変数アクセス最適化で 4M バイト以上離れた変数を参照する場合の注意事項 (No.6)

4.1.7 その他変更・改善

主に以下のような変更・改善を行いました。

(a) デバッグ情報の改善

不要なデバッグ情報を削除することで、オブジェクト・ファイル(*.obj)やロード・モジュール・ファイル(*.abs)のサイズが小さくなるように改善しました。

(b) セクション演算子の仕様変更

アセンブラのセクション演算子"STARTOF", "SIZEOF"の項に、同一ファイル内に定義がないセクションも指定可能にしました。

注：V1.02.00 まではセクション集合演算子という名称でしたが、V1.03.00 でセクション演算子に変更しました。

(c) インラインアセンブラの仕様変更

#pragma inline_asm で宣言したアセンブリ記述関数内のインラインアセンブラに対して、アセンブラ制御命令\$MACRO/\$NOMACROによる命令展開の可否を指定できるようにしました。なお、デフォルトでは命令展開は行いません。

(d) 内部エラーの改善

ビルド時にコンパイラが内部エラーを発生するケースがありましたが、これを改善しました。

4.1.8 MISRA-C:2004ルールによるソース・チェック機能 **【professional】**

MISRA-C:2004 ルールによりソース・チェックを行う以下のオプションが professional 版のライセンス登録時のみ使用できるようにしました。

- -Xmisra2004
- -Xignore_files_misra
- -Xcheck_language_extention

4.1.9 MISRA-C:2012ルールによるソース・チェック機能 **【professional】**

MISRA-C:2012 ルールによりソース・チェックを行う-Xmisra2012 オプションを追加しました。また、ソース・チェックの対象外とするファイルを指定する-Xignore_files_misra オプション、言語拡張により部分抑止されるソース・チェックを有効にする-Xcheck_language_extension オプションと同時指定できるようにしました。

本オプションを指定して静的なソース・チェックを行うことにより、ユーザープログラムの早期の品質向上が可能となります。

4.1.10 スタック破壊検出機能 **【professional】**

スタック破壊を検出する機能を追加しました。本機能は-Xstack_protector/-Xstack_protector_all オプションにより、あるいは拡張言語仕様 #pragma stack_protector/#pragma no_stack_protector により使用可能です。

本機能を使用することにより、関数の入口・出口にスタック破壊検出コードを生成します。具体的には以下の3つの処理を実行する命令を生成します。

1. 関数の入口で、当該関数スタック・フレームのローカル変数領域の直前（上位方向）に4バイトの領域を確保し、オプションまたは#pragma の引数として指定した値を格納。
2. 関数の出口で、1.で格納した値が書き換わっていないかチェック。
3. 書き換わっている場合にはスタックが破壊されたとして __stack_chk_fail 関数を呼び出し。

__stack_chk_fail 関数はユーザーが定義する必要があります。スタックの破壊検出時に実行する処理を記述してください。例えば、スタック領域を破壊してしまう以下のようなプログラムを実行した場合に __stack_chk_fail 関数が実行されることになります。

```
1: void f1() {
2:   volatile char str[10];
3:   int i;
4:   for (i = 0; i <= 10; i++){
5:     str[i] = i;           // i=10 の場合にスタックが破壊
6:   }
7:   ...
```

本機能を使用した動的なチェックにより、セキュリティ対策（スタック・オーバーフロー、セキュリティ攻撃の防止）が可能となります。

第5章 注意事項

本章では、CC-RHの注意事項について説明します。

5.1 FEレベル例外

#pragma interrupt 指令により復帰/回復不可のFE レベル例外の割り込み関数を定義した場合、つまり priority に FENMI や SYSERR 等を指定した場合、割り込み関数の出口コードは出力しません。プログラムで適切に処置してください。

すべての商標および登録商標は、それぞれの所有者に帰属します。

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い合わせください。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、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用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

営業お問い合わせ窓口

<http://www.renesas.com>

営業お問い合わ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 〒135-0061 東京都江東区豊洲3-2-24 (豊洲フォレシア)

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