

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	ユーザーズ・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	コンパイラ・ライセンス .....	5
4.2	G4MHコアの命令セット【G4MH】 .....	5
4.3	拡張浮動小数点システム・レジスタ【G4MH】 .....	5
4.4	レジスタ・バンク機能【G4MH】 .....	6
4.5	割り込み／例外処理ハンドラの割り込み仕様【G4MH】 .....	6
4.6	G4MHコアの命令セット用組み込み関数【G4MH】 .....	8
4.7	デバッグ命令用組み込み関数.....	8
4.8	C99標準ライブラリ関数.....	9
4.9	標準ライブラリ関数のインライン展開 .....	9
4.10	浮動小数点演算に対する実行効率を重視したコード生成 .....	9
4.11	浮動小数点数の逆数演算に対するコード生成.....	11
4.12	不正な間接関数呼び出し検出機能の拡充【professional】 .....	11
4.13	バイナリ・ファイルに対するCRC演算結果出力 .....	11
4.14	ヘキサ・ファイルの先頭ロード・アドレス変更 .....	11
4.15	注意事項の改修.....	12
4.16	その他の変更・改善 .....	12

## 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：

<https://www.renesas.com/cs+>

## 第2章 ユーザーズ・マニュアルについて

本製品に関連したユーザーズ・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	R20UT3516JJ0105
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	R20UT3283JJ0106

## 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 V2.00.00」を選択してください。

## 第4章 変更点

本章では、CC-RH V1.07.00 から V2.00.00 への主な変更点について説明します。

なお、professional 版のライセンス登録時のみ使用できる機能・変更点は【professional】と明記します。

また、G4MH コアを搭載したマイコンに特化した機能・変更点は【g4mh】と明記します。

### 4.1 コンパイラ・ライセンス

CC-RH V2.xx.xx を使用するには、V2 用のライセンスが必要です。

CC-RH V1.xx.xx を使用するために必要な V1 用ライセンスとは異なります。

### 4.2 G4MHコアの命令セット【g4mh】

CPU コアを指定する-Xcpu オプション（コンパイル・オプション及びアセンブル・オプション）の引数に **g4mh** を指定できるようにしました。

-Xcpu=g4mh オプションを指定すると、コンパイル時に G4MH の命令セットを使用したコードを生成します。また、G4MH の命令セットのアセンブルが可能です。

### 4.3 拡張浮動小数点システム・レジスタ【g4mh】

割り込み関数内で、拡張浮動小数点システム・レジスタ FXSR, FXXP の退避、復帰コードの出力可否を選択する-Xfxu オプションを追加しました。-Xfxu={on/off} の形式で指定します。

ただし、V2.00.00 では C ソース・ファイルをコンパイル時に拡張浮動小数点演算命令(FXU 命令)は生成しません。そのため、割り込み関数内でインライン・アセンブラを使用して FXU 命令を使用している場合に本オプションを指定してください。

引数	内容
on	割り込み関数内で、拡張浮動小数点システム・レジスタ FXSR, FXXP の退避、復帰コードを出力します。
off	割り込み関数内で、拡張浮動小数点システム・レジスタ FXSR, FXXP の退避、復帰コードを出力しません。

## 4.4 レジスタ・バンク機能【g4mh】

RESBANK 命令を使用する割り込み関数内で r20-r29, r31 の退避、復帰コードを出力する・しないを選択する **-Xresbank\_mode** オプションを追加しました。-Xresbank\_mode=num の形式で指定します。

num	内容
0	RESBANK命令を使用する割り込み関数内で、r20-r29, r31の退避・復帰コードを出力します。
1	RESBANK命令を使用する割り込み関数内で、r20-r29, r31の退避・復帰コードを出力しません。

なお本オプションは、マイコンの退避モードを設定するオプションではありません。あくまで退避モードに応じたレジスタの退避・復帰コードを出力するオプションです。マイコンの退避モードを

- ✓ 0 (RBCR0.MD=0) に設定している場合：-Xresbank\_mode=0を指定してください。
- ✓ 1 (RBCR0.MD=1) に設定している場合：-Xresbank\_mode=1を指定してください。

## 4.5 割り込み／例外処理ハンドラの割り込み仕様【g4mh】

割り込み／例外ハンドラの指定を以下のような #pragma interrupt 指令で行います。

```
#pragma interrupt [( ) 関数名[( 割り込み仕様 [, 割り込み仕様]...)] [... ] ( )]
```

V2.00.00 では「割り込み仕様」で指定可能な引数を以下のように変更しました。

- **priority =**

引数に **FPE,FXE** を指定できるようにしました。

FPE,FXE を指定した場合、それぞれの例外発生要因に応じたレジスタの退避・復帰コードを出力します。

- **channel =**

引数として指定可能な上限値を 511 から **2047** まで拡張しました。

また V2.00.00 では、以下の割り込み仕様を追加しました。

- **resbank**

EI レベル割り込み関数内で RESBANK 命令を使用します。

RESBANK 命令使用時は、多重割り込み関数でも EIPC,EIPSW の退避、復帰コードを出力しません。

次の場合はエラーE0523005 を出力します。

- ✓ -Xcpu=g4mh を未指定時
- ✓ priority=を指定し、かつその引数に EIINT\_PRIORITY 以外を指定時

次の場合は警告 W0523116 を出力します。いずれも特定レジスタの退避、復帰コードの出力を抑制しますが、resbank 指定を行うと割り込み関数内で対象のレジスタを変更しても割り込み関数からの復帰時に RESBANK 命令により値が復帰されます。

- ✓ #pragma interrupt 指令の割り込み仕様で(fpu=false)を同時に指定時
- ✓ -Xreg\_mode=32 オプション以外を指定時
- ✓ -Xreserve\_r2 を指定時
- ✓ -Xep=fix を指定時

- **fxu =**

割り込み関数内で FXSR,FXXP の退避、復帰コードの出力可否を指定します。引数として *true*, *false*, *auto* が指定できます。

引数	内容
<i>true</i>	割り込み関数内でFXSR,FXXPの退避、復帰コードを出力します。
<i>false</i>	割り込み関数内でFXSR,FXXPの退避、復帰コードを出力しません。
<i>auto</i>	-Xfxuオプションの指定に従います。

- **param =**

割り込み関数先頭で例外要因レジスタの値を引数に設定します。

要因には次の 6 種類を指定できます。これにより割り込み関数内で例外要因を引数で最大 4 種類まで取得することができるようになります。

- *eiic*<sup>注</sup>, *feic*<sup>注</sup>, *fpsr*<sup>注</sup>, *fxsr*, *fxxc*, *fxxp*

**【使用例】**

```
volatile unsigned long v;
#pragma interrupt handler (param=(eiic,fxsr,fxxc))
void handler (unsigned long a, unsigned long b, unsigned long c) {
    v = a; // 引数 a に EIIC の値が格納
    v = b; // 引数 b に FXSR の値が格納
    v = c; // 引数 c に FXXC の値が格納
}
```

注：-Xcpu=g4mh 指定時以外でも使用できます。

## 4.6 G4MHコアの命令セット用組み込み関数【g4mh】

以下の組み込み関数を追加しました。-Xcpu=g4mh オプションを指定時に使用できます。

アセンブラ命令	組み込み関数の指定形式
clip.b	long __clipb(long a);
clip.bu	unsigned long __clipbu(unsigned long a);
clip.h	long __cliph(long a);
clip.hu	unsigned long __cliphu(unsigned long a);
ldl.bu	long __ldlbu(unsigned char* a);
ldl.hu	long __ldlhu(unsigned short* a);
stc.b	long __stcb(unsigned char* a, unsigned char b);
stc.h	long __stch(unsigned short* a, unsigned short b);

また、-Xcpu=g4mh オプション指定時、割り込みレベルを制御する組み込み関数\_\_set\_il\_rh() を使用できません。使用した場合、エラーF0523073 になります。

-Xcpu=g4mh オプション指定時は割り込み制御レジスタを直接操作してください。

## 4.7 デバッグ命令用組み込み関数

以下の組み込み関数を追加しました。

アセンブラ命令	組み込み関数の指定形式
dbcp	void __dbcp(void);
dbpush	void __dbpush(long regID1, long regID2);
dbtag	void __dbtag(long a);



## 4.8 C99標準ライブラリ関数

下記の C99 用標準ライブラリ関数に対応しました。

- fmax(), fmaxf()
- fmin(), fminf()
- copysign(), copysignf()

## 4.9 標準ライブラリ関数のインライン展開

下記の標準ライブラリ関数を呼び出すコードを生成する際、関数呼び出しにするかインライン展開するかを制御する **-library** オプション<sup>(注)</sup>を追加しました。-library={function|intrinsic} の形式で指定します。

- abs(), labs(), llabs()
- fabs(), fabsf()
- sqrt(), sqrtf()
- fmax(), fmaxf()
- fmin(), fminf()
- copysign(), copysignf()

引数	内容
<i>function</i>	常に関数呼び出しにするコードを生成します。
<i>intrinsic</i>	インライン展開することにより実行効率を向上します。 なお、呼び出し条件等によってはインライン展開されない場合もあります。

(注) コンパイル・オプションです。リンク・オプションの-libraryとは別のオプションです。

## 4.10 浮動小数点演算に対する実行効率を重視したコード生成

浮動小数点演算に対して実行効率のよい演算コードを生成する **-relaxed\_math** オプションを追加しました。本オプションを指定した場合、下記に対してC言語規格やIEEE754の厳密な規定を緩和して演算コードを生成します。なお、厳密な規定に従った場合と演算結果が異なる場合があります。

- 演算精度
- 例外発生
- 非数
- 0.0の符号

## 【例1】演算精度を考慮しないケース

```
float func1(float a) {
    return (a * 3.0f) * 0.3f;
}
```

## オプション未指定時

```
_func1:
    movhi 0x00004040, r0, r2
    mulf.s r2, r6, r2
    mov 0x3E99999A, r5
    mulf.s r5, r2, r10
    jmp [r31]
```

## オプション指定時

```
_fun1:
    mov 0x3F666667, r2
    mulf.s r2, r6, r10
    jmp [r31]
```

オプション未指定時 : (a \* 3.0f) の演算後、0.3f との乗算

オプション指定時 : 3.0f \* 0.3f = 0.899...f をコンパイル時に計算し、a \* 0.899...f の乗算 1 回に変換

## 【例2】非数の入力を想定しないケース

```
float func2(float a, float b) {
    return (a > b) ? a : b;
}
```

## オプション未指定時

```
_func2:
    cmpf.s 0x00000006, r7, r6
    cmovf.s 0, r6, r7, r10
    jmp [r31]
```

## オプション指定時

```
_func2:
    maxf.s r6, r7, r10
    jmp [r31]
```

オプション未指定時 : cmpf.s 命令によって a または b が非数の場合は偽となり b を返却

オプション指定時 : maxf.s 命令によって a または b が非数の場合、非数でない方の値を返却

## 【例3】0.0の符号を無視するケース

```
float func3(float a, float b) {
    return -(a - b);
}
```

## オプション未指定時

```
_func3:
    subf.s r7, r6, r2
    negf.s r2, r10
    jmp [r31]
```

## オプション指定時

```
_func3:
    subf.s r6, r7, r10
    jmp [r31]
```

オプション未指定時 : a と b が -0.0f のとき、-0.0f を返却

オプション指定時 : -(a-b)を(b-a)に変換したため、a と b が -0.0f のとき+0.0f を返却

## 4.11 浮動小数点数の逆数演算に対するコード生成

浮動小数点数の逆数演算に対して `recipf` 命令(`recipf.d`, `recipf.s`)を生成する `-use_recipf` オプションを追加しました。

V2.00.00 未満ではデフォルトで、実行効率向上のため `recipf` 命令を生成しています。ただし、`recipf` 命令は常に FPU の不正確演算例外が発生します。このため、V2.00.00 では常に `divf` 命令を生成します。`recipf` 命令を生成させるには `-use_recipf` オプションを指定してください。

## 4.12 不正な間接関数呼び出し検出機能の拡充【professional】

V1.07.00 でサポートした不正なアドレスへの間接関数呼び出しを検出する機能を拡充し、`-cfi_ignore_module` オプションにライブラリファイル(\*.lib)を指定できるようにしました。

## 4.13 バイナリ・ファイルに対するCRC演算結果出力

バイナリ・ファイルに CRC 演算結果を出力できるようにしました。

V2.00.00 未満では `-form={hexadecimal | stype}` の場合にのみ `-crc` オプションが有効でしたが、V2.00.00 では `-form=binary` 指定時にも有効になります。

## 4.14 ヘキサ・ファイルの先頭ロード・アドレス変更

出力ファイルを指定するリンクオプション `-output=suboption` の `suboption` に `load-address` を指定できるようにしました。`-output=/load-address` を指定した場合、インテル拡張ヘキサ・ファイルまたはモトローラ・S タイプ・ファイルを出力する際に、ファイル上の先頭ロード・アドレスを `load-address` で指定した値に変更します。PIC 機能を有効にしたファイルを生成する際に効果的なオプションです。

## 4.15 注意事項の改修

以下 4 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。

- 初期化子の指定がある構造体/配列/共用体型の static 変数宣言に関する注意事項 (No.19)
- 予約シンボルを使用した場合のアセンブラの注意事項 (No.20)
- -Xmulti\_level オプション指定時に自動変数の初期化子が配置されるセクションに関する注意事項 (No.21)
- -store\_reg オプションに関する注意事項 (No.22)

## 4.16 その他の変更・改善

主に以下のような変更・改善を行いました。

(a) アセンブル・リスト・ファイル(\*.prn)

\*.prn ファイルの末尾"Command Line Parameter" の内容に差分が出る場合があります。  
ただし生成コードには影響ありません。

(b) 無効なオプション指定

無効なオプションを指定した際にエラーや警告が出る場合があります。

【例 1】 -P オプションを指定せずに-Xpreprocess オプションを指定した場合

V2.00.00 未満：何も出力せず本オプションを無視

V2.00.00 以降：警告「W0511151:"-P"オプションが指定されていないので、"-Xpreprocess"オプションを無視します。」を出力して本オプションを無視

【例 2】 引数を指定できない-V オプションを-VV と指定した場合

V2.00.00 未満：エラー「E0511109:"-V"オプションに引数は指定できません。」を出力

V2.00.00 以降：エラー「E0511108:"-VV"は認識されないオプションです。」を出力

(c) 最適化の強化

主にループ文の解析処理を強化し、実行速度を改善しました。

また、その他の最適化を強化しました。

(d) 内部エラーの改善

ビルド時に内部エラーが発生する場合がありますでしたが、これを改善しました。

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生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品、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またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 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、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング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）、交通制御（信号）、大規模通信機器、  
金融端末基幹システム、各種安全制御装置等

- 当社製品は、データシート等により高信頼性、Harsh environment向け製品と定義しているものを除き、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（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じて、当社は一切その責任を負いません。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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、データシート等において高信頼性、Harsh environment向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他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。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. 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。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
  10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
  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。
  12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 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直接的、間接的に支配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。

(Rev.4.0-1 2017.11)



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 〒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。  
総合お問合せ窓口：<https://www.renesas.com/contact/>