

この度は、弊社製品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 ユーザーズマニュアルについて .....	2
第2章 変更点 .....	3
2.1. -SECURITY_IDオプションの仕様拡張 .....	3
2.2. .type疑似命令の追加 .....	3
2.3. .alias疑似命令の追加 .....	3
2.4. .weak疑似命令の追加 .....	4
2.5. シンボル名に指定できる文字の追加 .....	4
2.6. コンパイラライセンスに関するメッセージの改善 .....	4
2.7. メッセージの追加 .....	4
2.8. -DEBUG_MONITORオプションの仕様拡張 .....	4
2.9. 変数の記憶域期間の変更 .....	5
2.10. 命令MACHU/MACHのサポート .....	5
2.11. 1ビット論理演算の生成コードの改善 .....	8
2.12. 直下にある関数を呼び出す場合の生成コードの改善 .....	9
2.13. 注意事項の改修 .....	10

## 第1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-RL コンパイラ ユーザーズマニュアル	R20UT3123JJ0111
(統合開発環境 CS+と併用する場合) CS+ 統合開発環境 ユーザーズマニュアル CC-RL ビルド・ツール操作編	R20UT3284JJ0110

## 第2章 変更点

本章では、CC-RL V1.10.00 から V1.11.00 への主な変更点について説明します。

### 2.1. -SECURITY\_ID オプションの仕様拡張

-SECURITY\_ID オプションに指定可能な値の桁数を、従来の 10 バイト固定から、-DEVICE オプションの指定により判定するように変更しました。

### 2.2. .type 疑似命令の追加

アセンブリ言語上のシンボル定義に対して、関数であるか、変数であるか の情報と、サイズの設定する.type 疑似命令を追加しました。

指定した情報は、リンク・マップ上で確認することができます。

入力ソース

```
.TYPE symA, FUNCTION, 8  
symA:
```

出力リスト・ファイル

```
SYMBOL ADDR SIZE INFO COUNTS OPT  
symA  
00000000 8 func ,I 0
```

### 2.3. .alias 疑似命令の追加

シンボル定義に別名を設定する.alias 疑似命令を追加しました。

別名には元のシンボルと異なるリンケージ属性を与えることができます。

次のように記述すると、別名の symA のみがモジュール外から参照できるようになります。

入力ソース

```
symB:  
.PUBLIC symA  
.ALIAS symA, symB
```

出力リスト・ファイル

```
SECTION=  
SYMBOL ADDR SIZE INFO COUNTS OPT  
  
symA
```

```
00000120 0 none ,g 0
symB
00000120 0 none ,l 0
```

## 2.4. .weak 疑似命令の追加

シンボルのリンケージ属性に弱結合属性を指定する .weak 疑似命令を追加しました。

弱結合属性を持つシンボルは外部結合属性を持つシンボルと同じく、別のモジュールから参照することができますが、複数のモジュールに同名の弱結合属性シンボルがある場合、エラーにならず、いずれか1つのみがリンクされます。

## 2.5. シンボル名に指定できる文字の追加

シンボル名に '\$' を含めることが可能になりました。ただし、シンボル名の先頭には '\$' は使用できません。また、シンボル名の先頭以外に '.' を含めることが可能になりました。

## 2.6. コンパイラライセンスに関するメッセージの改善

コンパイラライセンスに関するメッセージに、対象となるライセンスや対処方法を追加しました。

メッセージ番号	メッセージ
E0511178	CC-RL V1 に対応した professional 版のライセンスが確認できませんでした。"string" オプションを使用できません。professional 版の購入を検討ください。
W0511180	CC-RL V1 の評価期間の有効期限が切れています。
W0511185	professional 版の機能の試用期間は残り <i>number</i> 日です。professional 版の購入を検討ください。
W0561016	The evaluation version of CC-RL V1 is valid for the remaining <i>number</i> days. After that, link size limit (64 Kbyte) will be applied. Please consider purchasing the product.
W0561017	The evaluation period of CC-RL V1 has expired. Please consider purchasing the product.
F0563430	The total section size exceeded the limit of the evaluation version of CC-RL V1. Please consider purchasing the product.

## 2.7. メッセージの追加

次のメッセージを追加しました。アセンブラ疑似命令の記載に対して、必要なオペランドが記載されていない場合に出力します。

メッセージ番号	メッセージ
E0550272	"string"が必要です。

## 2.8. -DEBUG\_MONITOR オプションの仕様拡張

OCD モニタのメモリ領域を確保する際に、-DEVICE オプションで指定されたデバイス仕様にあわせて

領域を確保するように変更しました。

#### V1.10 以前

- 0x2, 0x3, 0xCE ~ 0xD7 番地と、OCD モニタ先頭アドレスから OCD モニタ終了アドレスの領域を 0xFF で充てんします。

#### V1.11

- 0x2, 0x3 番地と、デバイス仕様にあわせた領域、および、OCD モニタ先頭アドレスから OCD モニタ終了アドレスの領域を 0xFF で充てんします。

## 2.9. 変数の記憶域期間の変更

記憶域クラス指定子 `static` を使って定義した変数の参照を効率化する 機能を強化しました。

高速化のため、変数の記憶域期間を静的記憶域期間から自動記憶域期間に変更します。

本機能を有効にするには、オプション `-Olevel` の指定を省略するか、あるいは次のオプションを指定してください。

- `-Odefault/-Osize/-Ospeed` のいずれか

次のソースコードでは、コードサイズが小さくなり、実行が高速になります

ソースコード

```
unsigned id(unsigned parameter){
    static unsigned result;
    result = parameter;
    return result;
}
```

出力例

V1.10 (-cpu=S3)	V1.11 (-cpu=S3)
<pre>.SECTION .textf,TEXTF _id: .STACK _id = 4 movw !LOWW(_result@1@id), ax ret .SECTION .bss,BSS .ALIGN 2 _result@1@id: .DS (2)</pre>	<pre>.SECTION .textf,TEXTF _id: .STACK _id = 4 ret</pre>

## 2.10. 命令 MACHU/MACH のサポート

S3 コアが提供する積和命令 MACHU/MACH を生成をサポートします。

本機能を有効にするには、次のオプションを指定してください。

- `-use_mach=mach`

`-use_mach` オプションの説明は次の通りです。

=====

[指定形式]

`-use_mach={not_use|mach}`

`not_use` : 命令 MACHU/MACH を使わないコードを生成する

`mach` : 命令 MACHU/MACH を使うコードを生成する

[省略時解釈]

`-use_mach=not_use` オプションの指定と同じです。

[詳細説明]

- 積和命令 MACHU/MACH の使用の有無を指定します

- `-cpu=S1/-cpu=S2` オプションの指定時に `-use_mach=mach` オプションを指定するとコンパイル・エラーになります。

- `-use_mach=mach` オプションを指定すると、積和命令 MACHU/MACH が使用するシステム・レジスタ MACR は、関数呼出し前後、および、割込発生前後で、値を保持されるレジスタになります。

=====

統合開発環境 CS+では、このオプションは、次の欄で指定できます。

ビルド・ツール > コンパイル・オプション > その他 > その他オプション

統合開発環境 e2 studio では、このオプションは、プロジェクトのプロパティの 次の欄で指定できます。

C/C++ビルド > Settings > Compiler > ユーザー > ユーザー定義オプション

次のソースコードでは、コードサイズが小さくなり、実行が高速になります。

ソースコード

```
long mach(long src, short* lhs, short *rhs){
    src += (long)lhs[0] * (long)rhs[0];
    src += (long)lhs[1] * (long)rhs[1];
    src += (long)lhs[2] * (long)rhs[2];
    return src;
}
```

出力例

V1.10 (-cpu=S3)	V1.11 (-cpu=S3 -use_mach=mach)
<pre>.SECTION .textf,TEXTF _mach: .STACK _mach = 12 push bc subw sp, #0x04 movw hl, ax</pre>	<pre>.SECTION .textf,TEXTF _mach: .STACK _mach = 12 movw hl, !0xFFFF0 push hl movw hl, !0xFFFF2</pre>

<pre> push de movw ax, [de+0x02] movw bc, ax movw ax, [sp+0x0C] movw de, ax movw ax, [de+0x02] mulh movw [sp+0x02], ax movw ax, bc movw [sp+0x04], ax pop de push de movw ax, [de] movw bc, ax movw ax, [sp+0x0C] movw de, ax movw ax, [de] mulh addw ax, hl movw hl, ax movw ax, [sp+0x06] sknc .BB@LABEL@1_1:: entry incw ax .BB@LABEL@1_2:: entry addw ax, bc movw [sp+0x06], ax movw ax, [sp+0x02] movw bc, ax movw ax, hl addw ax, bc movw [sp+0x02], ax movw ax, [sp+0x04] movw bc, ax movw ax, [sp+0x06] sknc .BB@LABEL@1_3:: entry incw ax .BB@LABEL@1_4:: entry addw ax, bc movw [sp+0x06], ax </pre>	<pre> push hl push ax push bc movw ax, [sp+0x0C] movw hl, ax movw ax, [de] movw bc, ax movw ax, [sp+0x02] movw 0xffff0, ax movw ax, [sp+0x00] movw 0xffff2, ax movw ax, [hl] mach movw ax, [de+0x02] movw bc, ax movw ax, [hl+0x02] mach movw ax, [de+0x04] movw bc, ax movw ax, [hl+0x04] mach movw ax, 0xffff2 movw bc, ax movw ax, 0xffff0 addw sp, #0x04 movw hl, ax pop ax movw 0xffff2, ax pop ax movw 0xffff0, ax movw ax, hl ret </pre>
--	---

```
    pop hl
    push hl
    movw ax, [hl+0x04]
    movw bc, ax
    movw ax, [de+0x04]
    mulh
    movw de, ax
    movw ax, [sp+0x02]
    addw ax, de
    movw de, ax
    movw ax, [sp+0x06]
    sknc
.BB@LABEL@1_5:; entry
    incw ax
.BB@LABEL@1_6:; entry
    addw ax, bc
    movw bc, ax
    movw ax, de
    addw sp, #0x08
    ret
```

## 2.11.1 ビット論理演算の生成コードの改善

定数アドレス値によりメモリを参照し、1bitの論理演算を適用する場合の生成コードを改善しました。

次のソースコードでは、コードサイズが小さくなり、実行が高速になります。

ソースコード

```
typedef struct{
    unsigned char _b0:1;
    unsigned char _b1:1;
    unsigned char _b2:1;
    unsigned char _b3:1;
    unsigned char _b4:1;
    unsigned char _b5:1;
    unsigned char _b6:1;
    unsigned char _b7:1;
} MyStruct;

#define SFR0 ((MyStruct*)0xfffe)
#define SFR1 ((MyStruct*)0xffff)
void test(void){
    SFR0->_b2 &= SFR1->_b3;
```



```
}

```

## 出力例

V1.10 (-cpu=S1)	V1.11 (-cpu=S1)
<pre> _test:     .STACK_test = 4     movw hl, #0xFFFF     mov a, 0xFFFFE     mov1 CY, a.2     and1 CY, [hl].3     mov1 a.2, CY     mov 0xFFFFE, a ret </pre>	<pre> _test:     .STACK_test = 4     mov a, 0xFFFFE     mov1 CY, a.2     and1 CY, 0xFFFFF.3     mov1 a.2, CY     mov 0xFFFFE, a ret </pre>

## 2. 12. 直下にある関数を呼び出す場合の生成コードの改善

関数の末尾から、直下にある関数を呼び出す場合の生成コードを改善しました。  
本機能は、-Onothing を指定しない場合に、有効になります。

次のソースコードでは、コードサイズが小さくなり、実行が高速になります。

ソースコード

```

#pragma noline callee
void callee(void);
void caller(void){
    callee();
}
void callee(void){
}

```

## 出力例

V1.10 (-cpu=S1)	V1.11 (-cpu=S1)
<pre> _caller:     .STACK_caller = 4     br \$_callee _calllee:     .STACK_callee = 4 ret </pre>	<pre> _caller:     .STACK_caller = 4 _calllee:     .STACK_callee = 4 ret </pre>

## 2.13. 注意事項の改修

以下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(または FAQ)をご確認ください。

- pack 機能の使用に関する注意事項 (CCRL#027)
- ビット操作命令のビット位置指定を誤った場合の注意事項 (CCRL#028)
- 構造体型または共用体型の引数の使用に関する注意事項 (CCRL#029)
- ポインタ型のキャストの使用に関する注意事項 (CCRL#030)
- 無名共用体または無名構造体の使用に関する注意事項 (CCRL#031)
- アドレスを記憶領域に書き込んだ後で、読み出して参照する場合の注意事項 (CCRL#032)
- FAQ 3000575 [ワーニング] W0561321,W0561110 (CC-RL 使用時)
- FAQ 3000576 [エラー] C0564001 (W0561321 も同時に発生している場合)

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。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
Rev.1.00	2021.12.01		新規発行
Rev.1.01	2022.01.16	10	2.13 注意事項の改修の一覧を修正

## 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。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、お客様の責任において、お客様の機器・システムを設計ください。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みます。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. 当社製品または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. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 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。
6. 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も、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因またはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
  8. 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。
  9. 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。
  10. 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。
  11. 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。
  12. 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いたします。
  13. 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。
  14. 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.5.0-1 2020.10)

## 本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

[www.renesas.com](http://www.renesas.com)

## 商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。

## お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

[www.renesas.com/contact/](http://www.renesas.com/contact/)