

【お知らせ】

R20TS0467JJ0100

Rev.1.00

動的メモリ管理関数のメモリ破壊を検出して品質と安全性向上！ 2019.09.01号

ルネサス製コンパイラ professional 版

動的メモリ管理関数のセーフティ向上機能のご紹介

概要

ルネサス製コンパイラ（CC-RL/CC-RX/CC-RH） professional 版の機能のひとつ、“動的メモリ管理関数のセーフティ向上機能”についてご紹介します。

本機能を使用することにより、動的メモリ管理関数で確保したヒープ領域に対してメモリ破壊や不正な操作を行っていないかをチェックし、プログラムの品質や安全性を向上させることが可能になります。

1. 特長

1.1 プログラムの品質と安全性確保に貢献

ルネサス製コンパイラ（CC-RL/CC-RX/CC-RH）は、動的メモリ管理関数を使用してメモリ領域（ヒープ領域）の動的確保および解放を行うことができます。

しかし、ヒープ領域を使用する場合、以下のような誤った処理によってプログラムの暴走や誤動作に至る可能性があります。

- ・ ヒープ領域として確保したメモリサイズを超えたデータの書き込み
- ・ ヒープ領域の二重解放

“動的メモリ管理関数のセーフティ向上機能”は「監視領域」と「ポインタ」のチェックで、ヒープ領域での誤った処理を検出し、プログラムの暴走や誤動作を防止します。

以下に“動的メモリ管理関数のセーフティ向上機能”の有無による処理フローを示します。

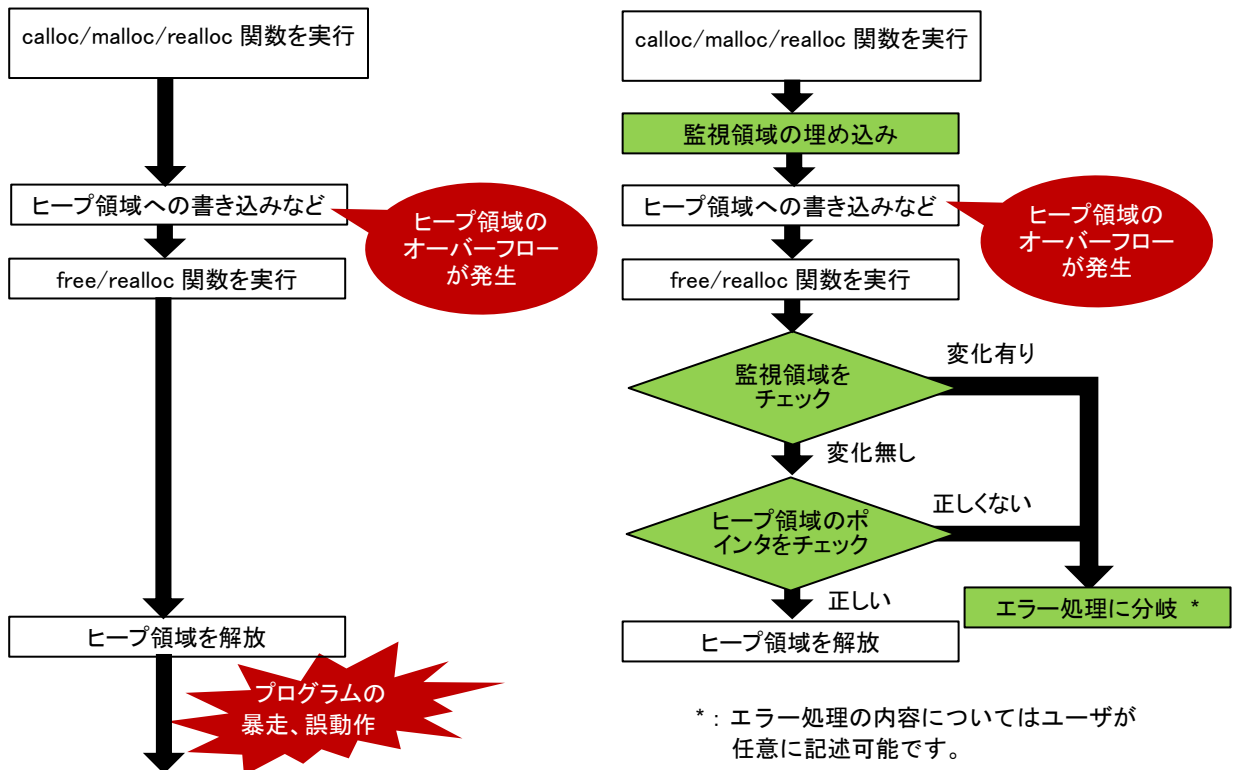


図 1-1 通常時の処理

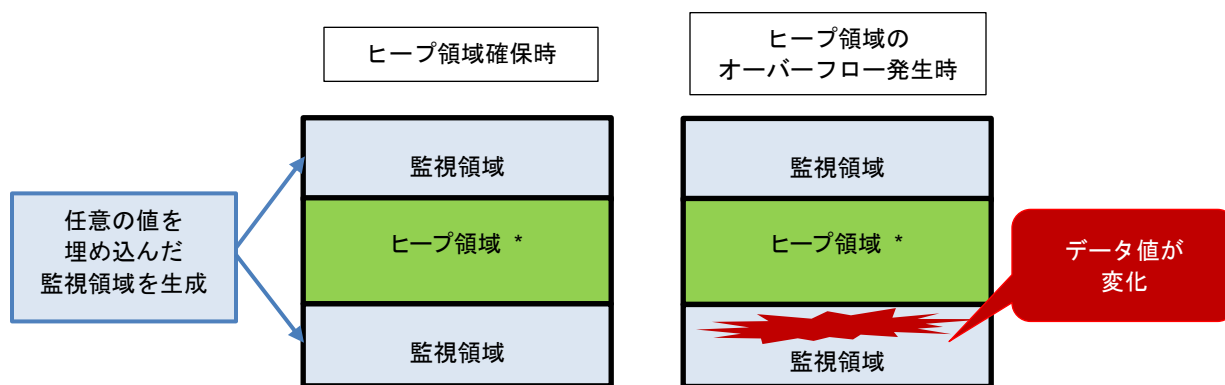
図 1-2 “動的メモリ管理関数のセーフティ向上機能”有効時の処理

■ 監視領域のチェック

以下のいずれかの関数を使用してヒープ領域を確保する際、ヒープ領域の前後に任意の値を埋め込んだ監視領域を生成し、ヒープ領域を解放する際、監視領域のデータをチェックします。

- ・ calloc、malloc、または realloc

チェックの結果、データが埋め込み値と異なる場合は、ヒープ領域の解放は行わずエラー処理^(注)に分岐します。



* : CC-RL は 2 バイト、CC-RX および CC-RH は 4 バイト

図 2 ヒープ領域確保時とオーバーフロー発生時の監視領域

■ ヒープ領域のポインタのチェック

ヒープ領域を解放する際、解放するヒープ領域のポインタについて、以下のチェックを行います。ポインタが正しくない場合は、ヒープ領域の解放は行わずエラー処理^(注)に分岐します。

- 以下のいずれかの関数を使用して確保したヒープ領域のポインタであること
 - ・ calloc、malloc、または realloc
- 解放済みのヒープ領域のポインタでないこと

注 : エラー処理の内容についてはユーザが任意に記述可能です。

1.2 実行例

“動的メモリ管理関数のセーフティ向上機能”は、統合開発環境(CS+または e² studio)の GUI 上で簡単に設定できます。

■ 統合開発環境の設定

【CS+の場合】

- (1) メニューバーの[表示]から[プロジェクト・ツリー]を表示し、[ビルド・ツール]を選択
- (2) [プロパティ]タブ→[リンク・オプション]タブ→[ライブラリ]カテゴリ→[メモリの解放時にメモリ破壊を検出する]プロパティで「はい」を指定

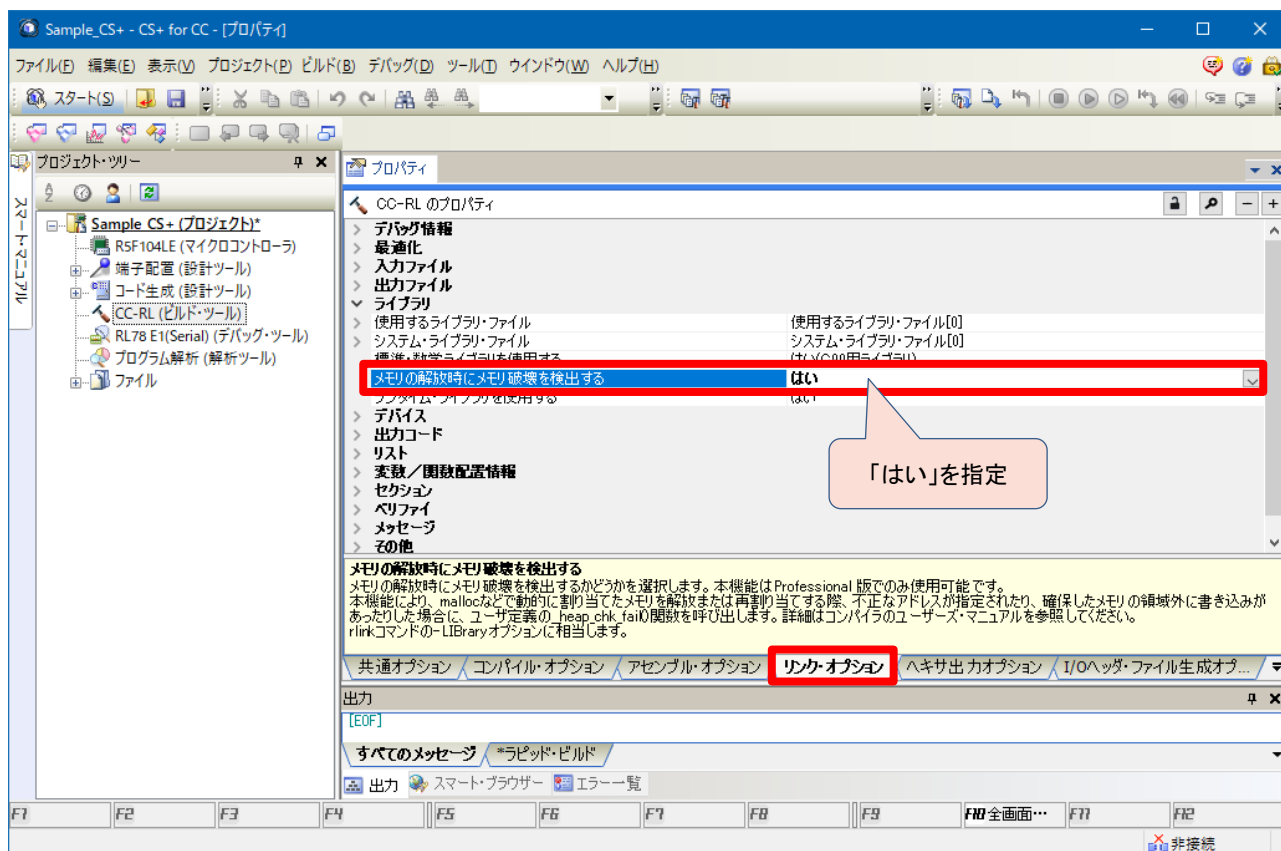


図 3 CS+の設定方法(CC-RL の場合)

【e² studio の場合】

- (1) メニューバーの[プロジェクト]から[プロパティ]を選択し、プロパティダイアログを起動
- (2) [C/C++ ビルド]→[設定]を選択、[ツール設定]タブ内で[Linker]→[入力]→[メモリの解放時にメモリ破壊を検出する]にチェック

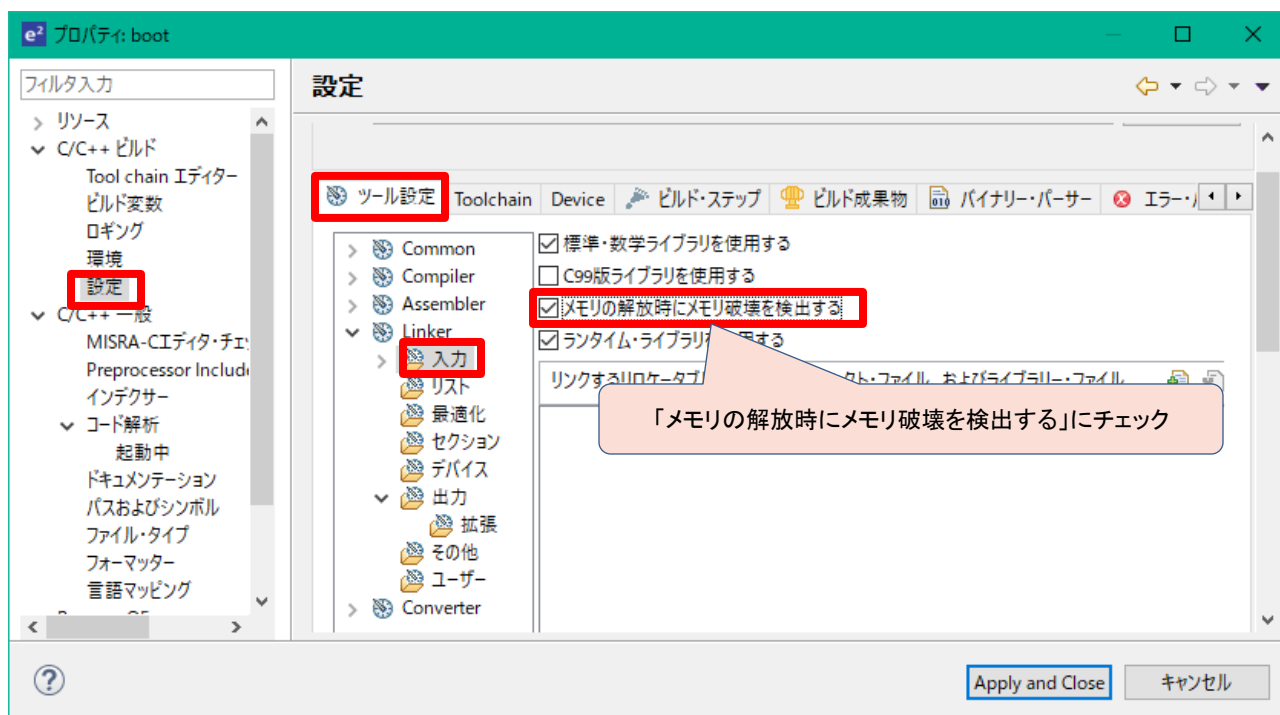


図 4 e² studio の設定方法(CC-RL の場合)

■ エラー処理関数の作成

不正な動作を検出した場合、エラー処理として__heap_chk_fail関数が呼ばれますので、__heap_chk_fail関数を作成してください。__heap_chk_fail関数の処理内容はユーザが任意に記述可能です。

以下は、エラー処理として、halt命令を実行する場合の記述例です。

```
void __heap_chk_fail(void) {
// エラー処理内容を記述
    __halt();
}
```

2. その他の professional 版の機能のご紹介

professional 版は、その他に、以下のような付加機能を提供しています。

ツールニュースおよびリーフレットで各機能についてご紹介していますので、ぜひ、ご参照ください。

MISRA-C ルールチェック機能
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0342 【お知らせ】 MISRA-C ルールチェックとコンパイルの同時実行でプログラム開発の工数削減と品質向上！ ルネサス製コンパイラ professional 版 MISRA-C ルールチェック機能のご紹介
制御レジスタ更新時の同期化機能
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0347 【お知らせ】 同期化処理の自動挿入で RH850 ファミリの開発工数の削減！ ルネサス製コンパイラ professional 版 制御レジスタ更新時の同期化機能のご紹介
スタック破壊検出機能
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0378 【お知らせ】 スタック領域の破壊を動的にチェックして品質と安全性向上！ ルネサス製コンパイラ professional 版スタック破壊検出機能のご紹介
不正な間接関数呼び出し検出機能
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ts0438 【お知らせ】 不正な間接関数呼び出しを未然に防止して品質と安全性を向上！ ルネサス製コンパイラ professional 版 不正な間接関数呼び出し検出機能
その他の便利な機能
https://www.renesas.com/search/keyword-search.html#genre=document&q=r20pf0024 ルネサス製コンパイラ professional 版 CC-RH の professional 版では、以下の機能もサポートしています。 <ul style="list-style-type: none"> ・半精度浮動小数点数

各機能の使用方法については、以下のアプリケーションノートをご参照ください。
 コピー&ペーストですぐに試せる C ソース例も掲載しています。

<https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ut4026>
 ルネサス製コンパイラ professional 版コンパイラ編

3. 購入方法

ご注文の際には、最寄りの弊社営業または特約店までご連絡ください。

standard 版ノードロック・ライセンスをすでにお持ちのお客様は「アップグレード(エディション)ライセンス」を追加でご購入いただくことで、standard 版から professional 版へアップグレードすることができます。製品型名は以下コンパイラパッケージの web ページをご参照ください。

CC-RL : https://www.renesas.com/rl78_c

CC-RX : https://www.renesas.com/rx_c

CC-RH : https://www.renesas.com/rh850_c

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Sep.01.19	-	新規発行

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。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。

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。

ニュース本文中の URL を予告なしに変更または中止することがありますので、あらかじめご承知ください。

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

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/

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。