

## 【注意事項】

R20TS0583JJ0100

Rev.1.00

2020.06.01 号

## RH850 ファミリ用 C コンパイラパッケージ

## 概要

RH850 ファミリ用 C コンパイラパッケージ CC-RH の使用上の注意事項を連絡します。

1. 2 バイトの領域へ定数値ゼロを書き込む場合の注意事項(No.29)

注：注意事項の後ろの番号は、注意事項の識別番号です。

## 1. 2 バイトの領域へ定数値ゼロを書き込む場合の注意事項(No.29)

## 1.1 該当製品

CC-RH V1.00.00~V2.02.00

## 1.2 内容

2 バイト長のメモリ領域へ定数値ゼロを書き込むソースコードをコンパイルするときに、1 バイトずれたアドレスへ書き込むコードを生成してしまいます。

このコードを実行すると、意図しない実行結果になるか、ミスアライン・アクセス例外が発生します。

本注意事項に該当する場合、ビルド時に次の番号の警告が出力されます。

W0550010 (V1.02.00 以前)

W0550019 (V1.03.00 以降)

## 1.3 発生条件

以下の条件をすべて満たす場合に発生します。

- (1) 2 バイト型の要素(メンバ)を持つ構造体型、または 2 バイト型の配列を使用している。  
※ 2 バイト型には、16 ビットのビットフィールドや、\_\_fp16 型も含まれます。
- (2) (1)の 2 バイトの要素に対して次のどちらかの形式で定数値ゼロを代入している。  
(2-1) ポインタ経由((\*p).m や p->m の形式)  
(2-2) 要素指定が定数式でない添字演算子(p[i]の形式)
- (3) (2)の代入先が、(1)の型の先頭要素ではない。

注意：-Xpack オプションや#pragma pack 指令でパッキング値に 2 を指定している場合は、4 バイトや 8 バイトの要素に対するアクセスも 2 バイト単位になり、該当する可能性があります。

## 1.4 発生例

以下に発生例を記します。赤文字が発生条件の該当箇所です。

ccrh -Osize tp.c

```
/* tp.c */
struct ST {
    char m1;
    char m2;
    short m3;      /* (1),(3) */
    short m4;      /* (1),(3) */
};

void func(struct ST *pst) {
    int i;
    for (i=0;i<10;++i) {
        (pst + i)->m2 = 1;
        (pst + i)->m3 = 0;    /* (2-1) */
        pst[i].m4 = 0;      /* (2-2) */
    }
}
```

メンバ m3 および m4 へ定数値ゼロを書き込む際、1 バイトずれたアドレスに書き込んでしまいます。

## 1.5 回避策

該当する要素の型を 2 バイトの型以外に変更してください。

構造体パッキング機能を使用することで該当している場合は、パッキングを解除してください。

## 1.6 恒久対策

CC-RH V2.03.00 で改修する予定です。リリース時期は未定です。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Jun.01.20	-	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。

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](http://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/](http://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。