

**【注意事項】**

R20TS0306JJ0100

Rev.1.00

**SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ**

2018.09.01 号

**概要**

SuperH RISC engine ファミリ用 C/C++コンパイラパッケージの使用上の注意事項を連絡します。

1. 静的な変数、または集成体型/共用体型の自動変数の初期値に定数式以外を記述した場合の注意事項 (SHC-0099)

注: 注意事項の後ろの番号は、注意事項の識別番号です。

**1. 静的な変数、または集成体型/共用体型の自動変数の初期値に定数式以外を記述した場合の注意事項 (SHC-0099)****1.1 該当製品**

SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ  
(V5.1A 以降は発生確認済)

**1.2 内容**

静的な変数、または集成体型/共用体型の自動変数の初期化子が定数式でない場合、ANSI 規格(C89 規格)に違反する記述にもかかわらず、コンパイルエラーを発生せず不正な初期値を設定することがあります。

**1.3 発生条件**

以下の条件をすべて満たしている場合に発生することがあります。

- (1) C プログラムとしてコンパイルしている。
- (2) 以下のいずれかの定義がある。
  - (2-1) 初期値を持つ静的な変数
  - (2-2) 初期値を持つ集成体型の自動変数
  - (2-3) 初期値を持ち、初期化の対象の型が集成体型である共用体型の自動変数
- (3) (2)の初期値に、構造体配列もしくは共用体配列のメンバのアドレスを指定している。
- (4) (3)の構造体配列もしくは共用体配列は静的に定義されている。
- (5) (4)の配列が以下(5-1)もしくは(5-2)のいずれかを満たしている。
  - (5-1) (4)の配列の添え字式が定数式でなく、その式に静的な変数を使用している。
  - (5-2) (4)の配列が条件演算子の結果となる式を記述し、その条件式に静的な変数を使用している。

## 1.4 発生例

発生条件の該当箇所を赤字で説明しています。

```
オプション -lang=cを指定している /* 発生条件(1) */
1: struct S {
2:     int m;
3:     } sa[3], sa2[3];
4:     int i;
5:     int *p1 = &sa[i].m; /* 発生条件(2-1) (3) (4) (5-1) */
6:     int *p2 = &(i?sa:sa2)[0].m; /* 発生条件(2-1) (3) (4) (5-2) */
7:     void func1(void)
8:     {
9:         int *p3[2] = {&sa[i+1].m, /* 発生条件(2-2) (3) (4) (5-1) */
10:                    &(i>3?sa:sa2)[1].m}; /* 発生条件(2-2) (3) (4) (5-2) */
11:     }
12:     void func2(void)
13:     {
14:         union {
15:             struct {
16:                 int *mem1_1;
17:             } mem1;
18:             int *mem2;
19:         } uni1 = {&sa[i].m}; /* 発生条件(2-3) (3) (4) (5-2) */
20:
21:         p1 = uni1.mem1.mem1_1;
22:     }
```

## 1.5 回避策

発生条件に該当する初期値を持つ変数があった場合、その変数に初期値を設定せず、実行時にその値を設定することで回避可能です。

```
1: struct S {
2:     int m;
3: } sa[3], sa2[2];
4: int i;
5: int *p1;      /* 回避策 */
6: int *p2;      /* 回避策 */
7:
8: void func1(void)
9: {
10:     int *p3[2]; /* 回避策 */
11:     p1 = &sa[i].m; /* 回避策 */
12:     p2 = &(i?sa:sa2)[0].m; /* 回避策 */
13:     p3[0] = &sa[i+1].m; /* 回避策 */
14:     p3[1] = &(i>3?sa:sa2)[1].m; /* 回避策 */
15: }
16: void func2(void)
17: {
18:     union {
19:         struct {
20:             int *mem1_1;
21:         } mem1;
22:         int *mem2;
23:     } uni1; /* 回避策 */
24:     uni1.mem1.mem1_1 = &sa[i].m; /* 回避策 */
25:
26:     p1 = uni1.mem1.mem1_1;
27: }
```

## 1.6 恒久対策

恒久対策の予定はありません。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2018.09.01	-	新規発行

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)

■総合お問い合わせ先  
<https://www.renesas.com/contact/>

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。

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。