

【注意事項】

R20TS0066JJ0100

Rev.1.00

RH850 ファミリ用 C コンパイラパッケージ

2016.09.01 号

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 回以上のループを含むプログラムに関する注意事項 (No.12)

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 回以上のループを含むプログラムに関する注意事項 (No.12)

1.1 該当製品

CC-RH V1.00.00 ~ V1.04.00

1.2 内容

最適化オプション指定時に、C ソースファイル上の 2 回以上ループするプログラム記述に対して、1 回しかループしないコードを出力する場合があります。

1.3 発生条件

以下の条件をすべて満たす時に発生する場合があります。

- (1) 最適化オプション-Osize または-Ospeed を指定している。
- (2) 「ループ制御変数を含む式 > 判定値を含む式」の形のループ判定式がある。
「式」は、ループ制御変数のみ、判定値のみの記述も含む。
- (3) 条件(2)の「ループ制御変数」は signed 型の変数であり、volatile 変数ではない。
- (4) 条件(2)の「判定値」は定数ではなく、volatile 変数でもない。
- (5) 条件(4)の値が条件(3)の型の最小値であり、ループ中で不変である。

```
1: unsigned long count;
2: signed char var1, var2; /* 発生条件(3)(4) */
3: void func(void)
4: {
5:     count = 0x00000000;
6:     var2 = -128; /* 発生条件(5) */
7:     for( var1 = 127; var1 > var2; var1--){count++;} /* 発生条件(2) */
8:     if(0x000000FF != count){funcsub();}
9:     ...
10: }
```

7 行目は 255 回のループとなり、変数”count”は 255 となるプログラムですが、最適化により 1 回のループとなります。

その結果、8 行目以降で変数”count”は 1 として動作します。

1.4 回避策

以下のいずれかにより回避可能です。

- (1) 不具合発生条件を満たさないよう最適化オプションを変更する。
- (2) ループ判定式の形を変更して、発生条件を満たさないようにする。

```
for( var1 = 127; var2 < var1; var1--){count++;}
```

```
for( var1 = 127; var1 >= var2+1; var1--){count++;}
```

- (3) ループ制御変数を、同じサイズの **unsigned** 型に変更するか、**volatile** 変数にする。
ループ制御変数を **unsigned** 型に変更した場合は、初期値や判定値等の型も合わせて変更する。

```
volatile signed char var1;
```

- (4) 判定値を定数に変更するか、**volatile** 変数にする。

```
volatile signed char var2;
```

1.5 恒久対策

次期バージョンで改修予定です。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2016.09.01	-	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

<http://japan.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。