

【注意事項】

R20TS0102JJ0100

Rev.1.00

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

2016.12.16 号

概要

SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ V.9 の使用上の注意事項を連絡します。

1. 標準 C ライブラリ関数“strcmp()”を使用した場合の注意事項 (SHC-0096)

注: 注意事項の後ろの番号は、注意事項の識別番号です。

1. 標準 C ライブラリ関数 strcmp を使用した場合の注意事項 (SHC-0096)

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 V.9.01 Release 00～V.9.04 Release 03

1.2 内容

ループ内^(注1)または割り込み関数で標準 C ライブラリ関数“strcmp()”を使用した場合、レジスタ^(注2)の値が保持されない場合があります。

注1: ループ内で使用した場合の不具合は V.9.02 Release 00～V.9.04 Release 03 で発生します。

注2: V.9.01 Release 00～V.9.03 Release 02 の場合、R1, R2, R3, R6 レジスタの値が保持されない場合があります。

V.9.04 Release 00～V.9.04 Release 03 の場合、R2, R3 レジスタの値が保持されない場合があります。

1.3 発生条件

以下の (1)または(2) に該当する場合に発生することがあります。

(1) 発生条件(1a)～(1d) をすべて満たしている。

(1a) SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ V9.02 Release 00～V9.04 Release 03 を使用している。

(1b) “-optimize=0” および “-optimize=debug_only” オプションを使用していない。

(1c) ループ内で標準 C ライブラリ関数“strcmp()”を使用し、かつその関数名を括弧で囲んでいない。

(1d) (1c) に該当するループ内で“strcmp()” より前にユーザー定義関数を呼び出していない。

ただし、インライン展開されないユーザー定義関数を呼び出している場合は、発生条件に該当しません。

(2) 発生条件(2a)と(2b) のいずれも満たしている。

(2a) “#pragma interrupt” を指定した関数で“strcmp()”を使用し、かつその関数名を括弧で囲んでいない。

(2b) (2a) に該当する関数内でユーザー定義関数を呼び出していない。

ただし、インライン展開されないユーザー定義関数を呼び出している場合は、発生条件に該当しません。

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。

(1) 発生条件(1) の場合：V9.02 Release 00～V9.03 Release 02 使用時 (発生条件(1a))

■ 発生ソース例：“-optimize=1” を使用 (発生条件(1b))

```
#include <string.h>
: :
void func(char *c1, char *c2, int *ip1, int i1, int i2, int i3){
    int lc_i1, lc_i2, lc_i3, lc_i4;
: :
    for(lc_i2= 0; lc_i2< 100; lc_i2++){
        /* ここにユーザー定義関数呼び出しの記述がない(発生条件(1d)) */
        if (strcmp(tbl[lc_i2].a, cp1)==0) { /* (発生条件(1c)) */
: :
            }
: :
        }
: :
    }
: :
}
```

■ コンパイル結果

```
: :
MOV.L L15+12,R1 ; H'FFE7FFFF ここでR1を設定(★)
MOV.L L15+16,R10 ; __slow_strcmp1
AND R1,R2
LDS R2,FPSCR
L11:
MOV.L @R13,R4
JSR @R10 ; "strcmp()"の呼び出し。ここでR0～R6を変更→(★)の値を保持すべきR1を変更
MOV R11,R5
STS FPSCR,R2
TST R0,R0
AND R1,R2 ; R1の値は(★)で設定したものとして、R1を利用
BF/S L13
LDS R2,FPSCR
: :
```

(2) 発生条件(2)の場合

■ 発生ソース例

```
#include <string.h>
#pragma interrupt(func)
int fl;
char str1[100], str2[100];

void func(){
    /* 関数内にユーザー定義関数呼び出しの記述がない(発生条件(2b)) */
    fl = strcmp(str1,str2); /* (発生条件(2a)) */
}
```

■ コンパイル結果

```
: :
MOV.L R0,@-R15
MOV.L R2,@-R15
MOV.L R4 @-R15
MOV.L R5,@-R15
MOV.L R6,@-R15
: :
MOV.L L11+2,R2 ; __slow_strcmp1
                ; "strcmp()"の呼び出し。
                ;ここでR0-R6(V.9.02 Release 00~V.9.03 Release 02 使用時)または
                ;R0,R2-R5(V.9.04 Release 00~V.9.04 Release 03 使用時)を変更
: :
JSR @R2
: :
MOV.L @R15+,R6
MOV.L @R15+,R5
MOV.L @R15+,R4
MOV.L @R15+,R2
MOV.L @R15+,R0
RTE            ; R3 と R1(V.9.02 Release 00~V9.03 Release 02 使用時) または
                ; R3(V.9.04 Release 00~V.9.04 Release 03 使用時)を
                ; 変更したまま割り込み処理を終了
```

1.5 回避策

以下のいずれかにより回避可能です。

- (1) “strcmp()” の関数名を括弧で囲む。
- (2) 別ファイルに以下の関数を定義し、“strcmp()” の代わりに使用する。かつ、“-file_inline” オプションで定義した関数を含むファイルを指定しない。

```
#include <string.h>

int strcmp2(const char *s1, const char *s2) /* 関数名は任意 */
{
    return (strcmp(s1,s2));
}
```

- (3) “-optimize=0” または “-optimize=debug_only” オプションを使用する。^(注)

注：発生条件(1) に該当している場合のみ有効な回避策です。

1.6 恒久対策

次期バージョンで改修する予定です。なお、リリース日程は未定です。

以上

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2016.12.16	-	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。