

M32Rファミリ用C/C++コンパイラパッケージ ご使用上のお願い

M32Rファミリ用C/C++コンパイラパッケージ(M3T-CC32R)の使用上の注意事項を 連絡します。

- inline関数を含むC/C++プログラムをコンパイルする場合の注意事項

1. 該当製品

M32Rファミリ用C/C++コンパイラパッケージ(M3T-CC32R)
V.4.00 Release 1 ~ V.5.00 Release 00

2. 内容

次のいずれかの問題が発生する場合があります。

- (1) 分岐命令の分岐先が誤ったコードを生成する。
- (2) コンパイラが次のようなエラーメッセージを出力して終了します。
このエラーメッセージは2つ以上出力されることがあります。

```
a132R: "File_name", line xxx: error: "Lxxx": symbol redeclared
```

File_name: コンパイラが生成する作業ファイル名が表示されます。

この作業ファイルは、コンパイル終了後は削除されます。

xxx: 数値が表示されます。

3. 発生条件

次の(1)~(4)の条件すべてを満たす場合に前項の問題(1)が、(1)~(3)の条件をすべて満たす場合に前項の問題(2)が発生することがあります。

- (1) -noinlineオプションを選択せずにコンパイルしている。
- (2) 次の文および演算子のいずれかを含む関数がある。
制御文: if, for, while, do, switch, goto, およびreturn
条件演算子: ? :
比較(等価・関係)演算子: ==, !=, <, >, <=, および>=

論理演算子: &&, および ||

(3) (2)に該当する関数の少なくとも1つがinline宣言されている。

(4) 次のいずれかの条件に該当している。

(a) -O2, -O3, -O6, -O7, -O のいずれかのオプションを選択している。

(b) -O, -O0, -O1, -O2, -O3, -O4, -O5, -O6, および -O7のいずれのオプションも選択せずに、-Ospace または -Otime のいずれかのオプションを単独で選択している。

発生例:

以下のソース例は、問題が実際に発生したソースから、主要部分のみを抜き出したもので、以下のソースだけでは問題は再現できません。

実際にお客様のソースコードが該当するかを判定するには、次項の検出方法を参照ください。

ソースファイル例 : sample.c

```
-----  
struct stag1 { int a, b, flag; };  
extern void func03(int data, int mode);  
inline void func02(int data, int flag) /* 発生条件(3) */  
{  
    switch (flag) { /* 発生条件(2)(3) */  
        case 0:  
            func03(data, 0);  
            break;  
        case 1:  
            func03(data, 1);  
            break;  
        default:  
            func03(data, -1);  
    }  
}  
void func01(struct stag1 *ptr)  
{  
    if (ptr->a <= 10) { /* 発生条件(2) */  
        func02(ptr->a, ptr->flag);  
    }  
    if (ptr->b <= 100) { /* 発生条件(2) */  
        func02(ptr->b, ptr->flag);  
    }  
}  
-----
```

コマンドライン例(% は、プロンプトを表します) :

% cc32R -S sample.c : 発生条件(1)
% cc32R -S -Otime sample.c : 発生条件(1)(4)

4. 検出方法

ソースコードが、条件(1)~(4)のすべてに該当しているとき、第2項で示した問題(1)が実際に発生しているかどうかを判断する方法を示します。

次のようにコンパイル時のコマンドラインオプションを変更してください。

(a) -O2, -O3, -O6, -O7, -O のいずれかひとつを選択している場合:

変更前 変更後

-O2 -O0

-O3 -O1

-O6 -O4

-O7 -O5

-O -O5

(b) -O, -O0, -O1, -O2, -O3, -O4, -5, -O6, および -O7のいずれも選択せずに、-Ospace, -Otime のいずれかを単独で選択している場合:

変更前 変更後

-Ospace -Ospace -O5

-Ospace -Otime -O5

オプション変更後に再コンパイルまたはリビルドした結果、第2項の問題(2)が発生しない場合は、問題(1)も発生していないと判断できます。

注意:

上記の検出方法は、問題(1)に限って発生するか否かを検出できるものであり、このほかの全ての生成コードが正常に動作することを検査するものではありません。上記により、問題(1)が発生していないと判断された場合でもお客様にて十分な動作検証を実施していただくようお願いします。

5. 回避策

次のいずれかの方法で回避できます。

ただし、(1)または(2)の方法で回避した場合は、前項に示した方法で実際に回避できたかを必ず確認してください。

(1) ソースファイルを分割する。

ソースファイル内の分岐処理を減らすことで、回避できる場合があります。

(2) 発生条件に該当するinline関数を #defineマクロに置き換える。

(3) -noinlineオプションを選択してコンパイルする。

この場合、今回の問題は確実に回避できますが、inline宣言した関数がインライン展開されなくなるため、生成コードの処理速度が低下することがあります。

6. 恒久対策

本問題は、次期バージョンアップの際に改修する予定です。

[免責事項]

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。

© 2010-2016 Renesas Electronics Corporation. All rights reserved.