

M3T-CC32R ご使用上のお願い -- 標準ライブラリの累乗関数について --

M32Rファミリ用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。

- 標準ライブラリの累乗関数(powf5, powf, およびpow)の注意事項

1. 該当製品

M3T-CC32R V.4.20 Release 1 および V.4.20 Release 1A

2. 内容

powf5関数、powf関数およびpow関数の演算結果が正しくない場合があります。

2.1 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) 以下のいずれかの関数を呼び出している。
 - (a) powf5関数を呼び出している。
 - (b) powf関数を呼び出しており、かつコンパイルオプション -m32re5を指定している。(*1)
 - (c) pow関数を呼び出しており、かつコンパイルオプション -float_onlyおよび -m32re5を指定している。(*1)
- (2) (1)の関数の入力引数を順にx, yとすると、以下のいずれにも該当しない。
 - (a) x, yのいずれかが0.0
 - (b) xが負(-0.0含む)で、かつyが非整数 (*2)
 - (c) x, yのいずれかが無限大(∞) または非数(NaN) (*2)

*1 オプション指定で、powf5関数の呼び出しに置き換わります。

*2 累乗演算は行われません。エラー番号を保持する外部変数errnoにマクロ名EDOMが代入されます。

※ 無限大(∞)および非数(NaN)について

これらの数値は、浮動小数点演算で不正な演算 (0.0で割る演算など) の答えとして返される特殊な数値です。通常は演算に使用しないでください。

2.2 発生例

```
ソースファイル例: sample.c-----  
-----#include <math.h>#include <mathf.h>float ans1, ans2, ans3, ans4;void  
func(void){/* 発生条件(1)(a), (2) */ ans1 = powf5(4.0, 0.5);/* 発生条件(1)(b) (-  
m32re5を指定している場合) , (2) */ ans2 = powf(10000.0, 0.25); /* 発生条件(1)(c)  
(-float_onlyと-m32re5を指定している場合) , (2) */ ans3 = pow(27.0, 1.0/3.0);/* 発  
生条件に該当しない */ ans4 = powf5(0.0, 1.0);}-----  
-----
```

3. 回避策

以下のいずれかの方法で回避してください。

(1) powf5関数をpowf関数に置換する

以下のように、mathf.h および mathf.h の後に、powf5関数をpowf関数に置換する記述(2行)を追加してください。

```
sample.c の回避例-----  
- #include <math.h> #include <mathf.h> #undef powf /* 追加 */  
#define powf5 powf /* 追加 */ float ans1, ans2, ans3, ans4; void func(void)  
{ ans1 = powf5(4.0, 0.5); ans2 = powf(10000.0, 0.25); /* -m32re5を指定し  
ても問題は発生しない */ ans3 = pow(27.0, 1.0/3.0); /* -  
float_onlyと-m32re5を指定しても問題は発生しない */ ans4  
= powf5(0.0, 1.0); }-----  
-----
```

(2) fixed_powf5関数を用いる

powf5関数を修正した fixed_powf5関数 (ファイル名: fixed_powf5.zip、サイズ: 3.01KB) をダウンロードし、「fixed_powf5関数の使用方法」にしたがって使用してください。

ダウンロードファイル: fixed_powf5.zipには、以下のファイルが含まれています。

small/fixed_powf5.mo ... smallモデル用fixed_powf5関数

small/fixed_powf5.stk ... 上記のスタックサイズ計算ファイル
medium/fixed_powf5.mo ... mediumモデル用fixed_powf5関数
medium/fixed_powf5.stk ... 上記のスタックサイズ計算ファイル
large/fixed_powf5.mo ... largeモデル用fixed_powf5関数
large/fixed_powf5.stk ... 上記のスタックサイズ計算ファイル

fixed_powf5関数の使用方法：

1. 発生条件(1)に該当する関数呼び出しを確認してください。
2. 該当する関数呼び出しを、全て「fixed_powf5」に変更してください。
(これにより、今回の改修のための対応であることが明確になります)
3. math.h または mathf.h をインクルードする行の直後に、次の宣言を必ず追加してください。

```
-----  
extern float fixed_powf5(float,float);  
-----
```

4. ロードモジュールを作成する際に、ダウンロードしたファイルに含まれる fixed_powf5.mo(*)をリンクしてください。
* メモリモデルに合わせて3ファイルのうち1つを選択してください。

4. 恒久対策

本問題は、次期バージョンで改修する予定です。

[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。