

M3T-NC308WAおよびM3T-NC30WA ご使用上のお願い

Cコンパイラパッケージ M3T-NC308WAおよびM3T-NC30WA の使用上の注意事項を連絡します。

- 浮動小数点型変数と定数値との乗算文、加算文、または減算文が連続している場合の注意事項

1. 該当製品

M32C/90, M32C/80シリーズ, M16C/80 およびM16C/70シリーズ用
M3T-NC308WA V.5.00 Release 1 ~ V.5.20 Release 1

M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10 およびR8C/Tinyシリーズ用
M3T-NC30WA V.5.00 Release 1 ~ V.5.30 Release 02

2. 内容

浮動小数点型変数と定数値との乗算文、加算文、または減算文が連続している場合、誤ったコードを生成します。

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) コンパイル時に最適化オプション -O3, -O4, -O5, -ORおよび -OSのいずれか1つ以上を指定している。
- (2) 以下のいずれかの演算文がある。
 - a. 浮動小数点型変数と浮動小数点型定数との乗算文が連続している。
 - b. 浮動小数点型変数と浮動小数点型定数との加算文が連続している。
 - c. 浮動小数点型変数と浮動小数点型定数との減算文が連続している。
- (3) (2)の連続した演算文の演算結果をすべて同一の変数に代入している。

2.2 発生例

```
-----  
float f;  
  
void func(void)  
{  
    f = f * 10;    /* 発生条件(2)a.および(3) */  
    f = f * 20;    /* 発生条件(2)a.および(3) */  
}  
-----
```

3. 回避策

次のいずれかの方法で回避してください。

- (1) 連続した演算文の間にダミーのasm()関数を挿入する。

例

```
-----  
float f;  
  
void func(void)  
{  
    f = f * 10;  
    asm();    /* ダミーのasm() */  
    f = f * 20;  
}  
-----
```

- (2) コンパイルオプション-Ono_break_souce_debug (-ONBSD)を追加で指定する。

4. 恒久対策

M3T-NC308WAおよびM3T-NC30WAの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。