

M3T-NC30WA ご使用上のお願い

M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tinyシリーズ用Cコンパイラパッケージ M3T-NC30WAの使用上の注意事項を連絡します。

- unsigned long型変数またはsigned long型変数と即値との乗算についての注意事項

1. 該当製品

M3T-NC30WA V.5.00 Release 1 ~ V.5.30 Release 1

2. 内容

unsigned long型変数またはsigned long型変数と即値との乗算を行なった場合、コンパイル中に以下のエラーが発生し、Cコンパイラが強制終了する場合があります。

Windows版のエラー表示： 不正な処理を行いました

Linux, EWS版のエラー表示： coreダンプ

3. 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) 最適化オプション-O1,-O2,-O3,-O4,-O5,-O,-OR, および-OSのうち1つ以上を指定している。
- (2) unsigned long型またはsigned long型の変数がある。
- (3) 即値 0xffffffffがある。
- (4) (2)の変数と(3)の即値の乗算を行っている。

4. 発生例

-----unsigned long g; /* 発生条件

```
(2) */void func(void){ g = g * 0xffffffff; /* 発生条件(3)(4) */}-----  
-----
```

5. 回避策

以下の手順で回避してください。

- (1) テンポラリ自動変数を設け、その変数に発生条件(3)の即値0xffffffffを代入する。
- (2) (1)の代入文の直後にダミーasm関数を記述する。
- (3) 発生条件(3)の即値のかわりに、(1)で設けた自動変数を用いて変数と乗算を行う。

```
例-----unsigned long  
g;void func(void){ unsigned long tmp; /* (1)テンポラリ自動変数を設ける */ tmp =  
0xffffffff; /* (1)即値を代入する */ asm(); /* (2)ダミーasm関数を記述する */  
g = g * tmp; /* (3)即値のかわりに自動変数を用いる */ ...}-----  
-----
```

6. 恒久対策

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。