

## M3T-NC308WA, M3T-NC30WA ご使用上のお願い -- ポインタ間接による関数呼び出しについて --

Cコンパイラ(アセンブラ・統合化開発環境付き) M3T-NC308WAおよびM3T-NC30WAの使用上の注意事項を連絡します。

- ポインタ間接による関数呼び出しに関する注意事項

### 1. 該当バージョン

M32C/80, M16C/80, M16C/70シリーズ用Cコンパイラ :

M3T-NC308WA V.5.00 Release 1 および V.5.10 Release 1

M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tinyシリーズ用Cコンパイラ:

M3T-NC30WA V.5.20 Release 1

### 2. 内容

ポインタ間接による関数呼び出しを行った場合、正しいコードが生成されない場合があります。

### 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-OSを指定している
- (2) ある関数Aの終了直前の処理で、別の関数Bを、ポインタ間接を使って呼び出している。
- (3) 関数Aの外部で、関数Bへのポインタ変数を宣言している。
- (4) 関数Aの引数には、スタック渡しの引数が含まれている。
- (5) 関数Aの先頭で、"enter #00H"が生成されている。

### 4. 発生例

-----  
void (\*sub)(void); /\* 発生条件(3) \*/

```
    :  
void func(long l) /* 発生条件(4) */  
{  
    :  
    (*sub)(); /* 発生条件(2) */  
}
```

---

## 5. 回避策

ポインタ間接を使った関数呼び出しの直後に、ダミーasm関数を追加してください。

```
void func(long l)  
{  
    :  
    (*sub)();  
    asm(); /* ダミーasm関数を追加 */  
}
```

---

## 6. 恒久対策

本内容は次期バージョンにて改修する予定です。

---

### [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。