

## M3T-NC30WA V.5.10 Release 1 ご使用上のお願い

M16C/60, M16C/30, M16C/20, M16C/10シリーズ用Cコンパイラ(アセンブラ・統合化開発環境付き)  
M3T-NC30WAの使用上の注意事項を連絡します。

- 代入文を連続して記述した場合の注意事項

### 1. 内容

代入文を連続して記述した場合、正しいコードが生成されない場合があります。

### 2. 発生条件

下記の条件をすべて満たした場合に発生します。

- (1) 最適化オプション-O, -O[1-5], -OR, -OSのいずれかを使用している。
- (2) 連続で記述した2つ以上の代入文がある。
- (3) 最初の代入文(以下、代入文Aとよぶ)の代入先および2番目の代入文(以下、代入文Bとよぶ)の代入元は、同一の自動変数(以下、変数Cとよぶ)である。
- (4) 代入文Aの代入元は、配列あるいはポインタ参照のいずれかで、かつ配列あるいはポインタ参照のオフセットには変数Cを使用している。
- (5) 変数Cには、メモリが割り当てられている。
- (6) 変数Cの型は以下のいずれかでかつ、volatile修飾されていない。
  - unsigned char、signed char
  - unsigned int、signed int
  - unsigned short、signed short
  - unsigned long、signed long
  - unsigned long long、signed long long

### 3. 発生例

---

```
extern char bbb;
extern char aaa[];

void func(void)
{
    char c;    /* 発生条件(6) */

    c = aaa[c]; /* 発生条件(2), (3), (4) */
    bbb = c;    /* 発生条件(2), (3) */
}
```

---

#### 4. 回避策

連続する代入文の間にダミーasm関数を記述してください。

---

```
void func(void)
{
    char c;

    c = aaa[c];
    asm();    /* <== ダミーasm関数 */
    bbb = c;
}
```

---

#### 5. 恒久対策

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。