

## 77xxシリーズ用Cコンパイラパッケージ M3T-NC77WA ご使用上のお願い

7700ファミリ 77xxシリーズ用CコンパイラパッケージM3T-NC77WAの使用上の注意事項を連絡します。

- 演算の途中でオーバーフローが発生する場合の注意事項

### 1. 該当製品

M3T-NC77WA V.3.00 Release 1 ~ V.5.20 Release 4

### 2. 内容

通常は演算の途中でオーバーフローが発生する場合、あふれたビットが演算に使用されることはありません。しかし発生条件に該当する場合に限り、あふれたビットを演算に使用します。

#### 2.1 発生条件

以下の条件を全て満たす場合に、発生します。

- (1) 整数定数の項が3つ以上連続する部分式がある（次の例では"0x4000 \* 5 / 2"が該当）。  
例)  $x = i + 0x4000 * 5 / 2 + j;$
- (2) (1)の部分式は、signed int型で計算される。
- (3) (1)の部分式は、演算の途中でオーバーフローした後に、続く演算でsigned int型の範囲に収まる値になる部分を含む。  
次の例では、 $0x4000 * 5$ の結果は $0x14000$ となり、オーバーフローするが、次の演算 $0x14000 / 2$ の結果は $0xA000$ となり、signed intの範囲に収まる。  
例)  $0x4000 * 5 / 2$

#### 2.2 発生例

(ソース例)

```
-----  
--  
void func(void)  
{  
    int i = 0x4000 * 5 / 2;    /* 発生条件(1),(2),(3) */  
}  
-----  
--
```

(NC77による生成コード例)

```
-----  
--  
;## # C_SRC :      int i = 0x4000U * 5 / 2;  
    ldm.W  #0a000H,DP:1  ; #2000H,DP:1が正しい。  
-----  
--
```

### 3. 回避策

以下のいずれかの方法で回避してください。

- (1) 符号無し整数型で計算するよう式を変更する。

発生例の回避例：

```
-----  
void func(void)  
{  
    int i = 0x4000U * 5 / 2;    /* 型接尾語Uを付加して  
                                0x4000を符号無し整数型にする */  
}  
-----
```

- (2) 一時変数を用いて式を分割する。

発生例の回避例：

```
-----  
void func(void)  
{  
    int tmp = 0x4000 * 5;    /* "0x4000 * 5"を一時変数  
                                tmpに代入して式をわけると */  
    int i = tmp / 2;  
}
```

}

-----

---

**[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。

© 2010-2016 Renesas Electronics Corporation. All rights reserved.