

M3T-NC308WA ご使用上のお願い

M32C/80,M16C/80,M16C/70シリーズ用Cコンパイラ(アセンブラ・統合化開発環境つき) M3T-NC308WAの使用上の注意事項を連絡します。

- 整数型からポインタ型へのキャストに関する注意事項

1. 該当製品

M3T-NC308WA V.2.00 Release 1 ~ V.5.10 Release 1

2. 内容

アドレス空間の範囲外の番地を表す整数をポインタ型にキャストした場合、その値と別のポインタ型の値との比較が正しく行われなことがあることがあります。

M3T-NC308WAでは、farポインタ型は32ビットですが、全アドレス空間が24ビットで表現可能なので、24ビットのアドレスレジスタに割り付ける場合があります。farポインタを保持する記憶域の実際のサイズが最小24ビットであるので、アドレス空間外を指示する整数値をfarポインタ型にキャストした場合には上位8ビットをゼロにする必要がありますが、ゼロにしていまませんでした。そのため、アドレス空間外を指示する整数値からfarポインタ型にキャストして、その値を別のポインタ型の値と比較すると、キャストした値がアドレスレジスタに割り当てられたかどうかによって値が異なるため、正しく比較できません。なお、メモリ上の変数や関数を指しているポインタの値は、上位8ビットが必ずゼロになりますので問題ありません。

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) 整数型の変数または整数型の定数から、ポインタ型にキャストしている。
- (2) (1)でキャストした結果の値を、別のポインタ型の値と比較している。
- (3) (1)のキャスト前の値が、0x00000000から0x00ffffffの範囲でない。
- (4) コンパイルの結果、(2)で比較するポインタ型の値の少なくとも一方が、比較のときまたは比較の前に、アドレスレジスタに割り付けられている。

2.2 発生例

以下は、M3T-NC308WA V.5.10 Release 1 を用いて、最適化オプション -O を指定した場合に発生する例です。レジスタの割付けはバージョンにより異なるため、問題が発生しないこともあります。

```
-----  
void func(char *p)  
{  
    if (p == 0) return;    /* 0 との比較は問題ありません */  
  
    if (p == (int *)(-1)) { /* 発生条件(1)整数型の定数(-1)をキャスト */  
        /* 発生条件(2)キャストの結果を用いた比較 */  
        /* 発生条件(3)-1は0から0x00ffffffの範囲外 */  
        return;  
    }  
    *p = 0;  
}  
  
void caller(void)  
{  
    func((int *)(-1));    /* 発生条件(1)整数型の定数(-1)をキャスト */  
}  
-----
```

3. 回避策

0 から 0x00ffffffの範囲外の値をポインタ型にキャストする場合には、キャスト前の値と定数 0x00ffffffとをAND演算し上位8ビットをマスクしてください。以下は整数型の定数の場合の例ですが、整数型の変数の場合も同様の方法で回避してください。

```
-----  
void func(char *p)  
{  
    if (p == 0) return;  
  
    if (p == (int *)(-1 & 0x00ffffff)) { /* 上位8ビットをマスクする */  
        return;  
    }  
    *p = 0;  
}  
  
void caller(void)  
{
```

```
func((int *)(-1 & 0x00ffffff)); /* 上位8ビットをマスクする */  
}
```

4. 恒久対策

本内容は、次期バージョンアップの際に改修する予定です。

[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。

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