

## M3T-NC30WAご使用上のお願い

-- switch文について --

M16C/60, M16C/30, M16C/20, M16C/10, M16C/Tiny, およびR8C/Tinyシリーズ用 Cコンパイラパッケージ(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。

- switch文に関する注意事項

### 1. 該当製品

M3T-NC30WA V.1.00 Release 1 ~ V.5.00 Release 1, および  
V.5.10 Release 1 ~ V.5.30 Release 1

### 2. 内容

switch文の制御式の型がsigned charの場合、正しいcaseラベルに分岐しないコードを生成する場合があります。

#### 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) switch文の制御式の型がsigned charである。
- (2) case値の最大値と最小値の符号が異なる。
- (3) (1)のswitch文の中にcaseラベルが12個以上存在する。
- (4) コンパイルの結果、(1)のswitch文について、分岐先アドレスのテーブルとそのテーブルを用いた間接分岐のアセンブリ言語コードが生成されている。

#### 2.2 発生例

C言語ソース:

-----  
signed char sc;  
void func(void)

```
{
  switch (sc) {
  case -8:
    ...
  case 1:
    ...
  }
}
```

---

上記C言語ソースから生成されたアセンブリ言語コード:

---

```
mov.b  _sc:R0L
mov.b  R0L,A0
sub.b  #0f8H,A0
cmp.b  #11H,A0
jnc    M1
mov.w  #10H,A0
M1:
  sha.w  #1,A0
S1:
  jmp.w  L35[A0]
L35:
  .word  L19-S1&0ffffH
  .word  L21-S1&0ffffH
  .word  L23-S1&0ffffH
  .word  L25-S1&0ffffH
  ...
}
```

---

### 3. 回避策

キャスト演算子を使ってswitch文の制御式をsigned int型に変換してください。

例 :

---

```
signed char sc;
void func(void)
{
  switch((signed int)sc) {
  case 0:
    . . . . .
  }
}
```

---

#### 4. 恒久対策

M3T-NC30WA V.5.30 Release 02以降では本問題は改修されています。

---

#### **[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.