

## 78K0用Cコンパイラ CA78K0およびCC78K0 ご使用上のお願い

78K0用CコンパイラCA78K0およびCC78K0の使用上の注意事項を連絡します。

- 浮動小数点定数の多重キャスト処理で誤ったコードを出力する注意事項
- strtol関数で文字列数値変換が誤った値になる注意事項

### 1. 浮動小数点定数の多重キャスト処理で誤ったコードを出力する注意事項

#### 1.1 該当製品およびバージョン

CA78K0 V1.20 ~ V1.30 (統合開発環境 CubeSuite+)

CA78K0 V1.00 ~ V1.11 (統合開発環境 CubeSuite)

CC78K0 V1.00 ~ V4.10 (統合開発環境 PM+)

#### 1.2 内容

浮動小数点定数に多重キャストを用いると演算結果が誤った結果になります。

#### 1.3 発生条件

以下のすべての条件を満たす場合に発生します。

- (1) 浮動小数点定数、または浮動小数点型にキャストした定数を、浮動小数点型にキャストしている。
- (2) (1)を整数型にキャストしている。
- (3) (2)を下記以外の演算で使用している。
  - 単純代入演算: =
  - 論理演算: &&または||
  - 条件演算: ? :
  - 単項演算: !

#### 1.4 発生例

-----  
[\* .C]

```
#define A ((long)((double)6031.0)) //(1)および(2)
```

```
void func(void)
{
    long x;
    x=A<<1;           //(3)
}
```

-----  
x = 12062 (= 6031 \* 2) となりません。

## 1.5 回避策

浮動小数点定数、または浮動小数点型にキャストした定数を、浮動小数点型にキャストしないでください。

```
#define A ((long)6031.0)
void func(void)
{
    long x;
    x=A<<1;
}
```

-----  
x = 12062 (= 6031 \* 2) となります。

## 2. strtol関数で文字列数値変換が誤った値になる注意事項

### 2.1 該当製品およびバージョン

CA78K0 V1.20 ~ V1.30 (統合開発環境 CubeSuite+)

CA78K0 V1.00 ~ V1.11 (統合開発環境 CubeSuite)

CC78K0 V1.00 ~ V4.10 (統合開発環境 PM+)

### 2.2 内容

strtol関数を使用して文字列数値変換を行った場合、正しく数値変換せずに、オーバーフロー値を返す場合があります。

### 2.3 発生条件

以下のすべての条件を満たす場合に発生します。

- (1) strtol関数を使用して文字列を数値に変換している。
- (2) (1)の関数処理で、文字列の先頭から順に数値変換を行い、変換の途中に 0x10000 ごとの桁上がりが発生する。  
0x10000 ごとの桁上がりが発生する場合とは、数値変換中に下記の条件を満たす場合です。

$$((N * \text{base}) / 0x10000) \neq ((N * \text{base} + c) / 0x10000)$$

N: 先頭から n文字目までを変換した値

base: 基数

c: n+1 文字目の値

発生例:

strtol("65537", &err, 10)を実行した場合、

4文字目までを変換した値 6553、基数 10、5文字目の値 7となり、

左辺 =  $((6553 * 10) / 0x10000) = 0$

右辺 =  $((6553 * 10 + 7) / 0x10000) = 1$

左辺 != 右辺 が成立して、オーバフロー値になります。

## 2.4 回避策

strtol関数で基数が10の場合には、atol関数に置き換えてください。

上記で対応できない場合は、回避策はありません。

## 3. 恒久対策

これら2件の注意事項の改修予定はありません。

---

### [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。