

M32Rファミリ用C/C++コンパイラパッケージ ご使用上のお願い

M32Rファミリ用C/C++コンパイラパッケージ(M3T-CC32R) の使用上の注意事項を 連絡します。

- 関数形式マクロの実引数に、同じ関数形式マクロを使用する場合の注意事項

1. 該当製品

M32Rファミリ用C/C++コンパイラパッケージ
V.5.00 Release 00 および V.5.01 Release 00

2. 内容

複数行に渡って記述された関数形式マクロの実引数に、同じ関数形式マクロを記述した場合、実引数に記述された関数形式マクロが展開されない場合があります。

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) 関数形式マクロを複数行にわたって記述している。
- (2) (1)項のマクロの実引数に、(1)項と同じ関数形式マクロを使用している。
- (3) (2)項の関数形式マクロが行の先頭にある。あるいは、行の先頭から(2)項の関数形式マクロとの間に空白類文字およびコメントしかない。

2.2 発生例

ソースファイル (sample1.cとする):

```
-----  
#define SMP_MACRO(a,b) (a + b)  
int var = SMP_MACRO(1,                <- 発生条件(1)項  
                  SMP_MACRO(2,3));   <- 発生条件(2)および(3)項  
-----
```

マクロ置換結果(cc32R -E sample1.c の出力内容):

```
#line 1 "sample1.c"
```

```
int var = (1 + SMP_MACRO(2,3)) ; /* SMP_MACRO(2,3)が展開されない */
```

3. 回避策

以下のいずれかの方法で回避してください。

(1) 発生条件(3)項の関数形式マクロを記述した行と直前の行を¥ 記号で連結する

sample1.cの変更例:

```
-----  
#define SMP_MACRO(a,b) (a + b)  
int var = SMP_MACRO(1,      ¥ <- この行の末尾に ¥ 記号を記述する  
      SMP_MACRO(2,3));  
-----
```

(2) 同一フォーマットで名前の異なるもうひとつの関数形式マクロを定義し、
それを実引数で使用する

sample1.cの変更例:

```
-----  
#define SMP_MACRO(a,b) (a + b)  
#define SMP_SAME_MACRO(a,b) (a + b) <- SMP_MACROと同一内容のマクロ  
int var = SMP_MACRO(1,  
      SMP_SAME_MACRO(2,3)); <- SMP_MACROの代わりに使用する
```

[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。