

M3T-NC30WAご使用上のお願い --#pragma ADDRESSについて--

Cコンパイラパッケージ M3T-NC30WA の使用上の注意事項を連絡します。

- #pragma ADDRESSを使用した場合の注意事項

1. 該当製品

M16C/60, M16C/30, M16C/20, M16C/10, M16C/Tiny, およびR8C/Tinyシリーズ用:
M3T-NC30WA V.1.00 Release 1 ~ V.5.30 Release 02

2. 内容

#pragma ADDRESSで定義された変数に対してビットアクセスを行うとアセンブルエラーが発生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) #pragma ADDRESSで定義されている変数が配列型の集合体である。
- (2) (1)で定義されたアドレスが、01FFFH番地よりも小さいアドレスである。
- (3) ビットアクセスした番地が01FFFH番地より大きいアドレスである。

2.2 発生例

```
-----  
typedef struct{  
    unsigned char b0:1;  
    unsigned char b1:1;  
    unsigned char b2:6;
```

```
}BIT;
```

```
#pragma ADDRESS bit 01FFEh  
BIT bit[10];  
void func(void)  
{  
    bit[2].b0 = 1;  
}
```

3. 回避策

asm関数でビットアクセスしてください。

```
typedef struct{  
    unsigned char b0:1;  
    unsigned char b1:1;  
    unsigned char b2:6;  
}BIT;  
  
#pragma ADDRESS bit 01FFEh  
BIT bit[10];  
void func(void)  
{  
    asm(" or.b  #01H,$$,bit[2].b0);  
}
```

4. 恒久対策

次期バージョンで改修する予定です。

[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。