

## A Note on Using the C Compiler Packages for the M16C MCU Family --With Passing an Address as an Argument to a Function--

Please take note of the following problem in using the C compiler packages for the M16C MCU family:

- With Passing an Address as an Argument to a Function
- 

### 1. Products and Versions Concerned

- (1) The C compiler package for the M32C series\*1 (M3T-NC308WA)  
V.5.00 Release 1 through V.5.40 Release 00
- (2) The C compiler package for the M16C series\*2 (M3T-NC30WA)  
V.5.00 Release 1 through V.5.40 Release 00A

\*1. Generic name of the M32C/90, M32C/80, and M16C/80 series

\*2. Generic name of the M16C/60, /30, /20, /10, /Tiny, and R8C/Tiny series

### 2. Description

When the address of a variable is used as an argument to a function, an incorrect value may be passed to the function.

### 3. Conditions

This problem occurs if the following conditions are all satisfied:

- (1) A call to a function is made, where the address of an automatic variable is used as an argument to the function.
- (2) Immediately after the function called in (1) is performed, execution returns from the function that made the call to the function in (1).
- (3) Optimizing options -OS and -O5 are both selected.

Example:

```
-----  
void sub(int *);  
void func(void)  
{  
    int i;  
    sub(&i); /* Conditions (1) and (2) */  
}  
void sub(int *i)  
{  
.....  
}  
-----
```

#### 4. Workaround

Place a dummy asm function immediately after the line describing the call to a function that takes the address of an automatic variable as an argument

```
-----  
void func(void)  
{  
    int i;  
  
    sub(&i);  
    asm("");  
}  
-----
```

#### 5. Schedule of Fixing the Problem

This problem has already been fixed in the following versions:

- (1) The C compiler package for the M32C series V.5.41 Release 00 and later
- (2) The C compiler package for the M16C series V.5.42 Release 00 and later

So use the latest version of each package.

---

#### [Disclaimer]

The past news contents have been based on information at the time of publication. Now changed or invalid information may be included. The URLs in the Tool News also may be subject to change or become invalid without prior notice.

