

A Note on Using the C Compiler Packages M3T-NC308WA and M3T-NC30WA --On Defining More Than One Void Function--

Please take note of the following problem in using the C compiler packages M3T-NC308WA and M3T-NC30WA (they are used for the M16C family of MCUs):

- On defining more than one void function

1. Products and Versions Concerned

- M3T-NC308WA V.5.00 Release 1 through V.5.20 Release 02
(for the M32C/90, M32C/80, and M16C/80 series of MCUs)
- M3T-NC30WA V.5.00 Release 1 through V.5.30 Release 02
(for the M16C/60, M16C/30, M16C/20, M16C/10, M16C/Tiny and R8C/Tiny series of MCUs)

2. Description

Even when two or more void functions are placed in different sections in the C source program, optimization places them in one section after compilation.

2.1 Conditions

This problem occurs if the following conditions are all satisfied:

- (1) One source file contains two or more void functions.
- (2) The void functions in (1) are placed in more than one section.
- (3) Any of the compile options -O3, -O4, -O5, -OR, and -OS is used.

2.2 Example

```
-----  
void func_1(void) /* Condition (1) */  
{
```

```
}
```

```
#pragma SECTION program prog /* Condition (2) */  
void func_2(void) /* Condition (1) */
```

```
{  
}
```

3. Workaround

Insert a dummy `asm()` function into each of the void functions that are misplaced after compilation.

```
void func_1(void)  
{  
}
```

```
#pragma SECTION program prog  
void func_2(void)  
{  
    asm(); /* Dummy asm() inserted */  
}
```

4. Schedule of Fixing the Problem

We plan to fix this problem in the next release of the products.

[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.

© 2010-2016 Renesas Electronics Corporation. All rights reserved.