RENESAS Tool News

RENESAS TOOL NEWS on November 1, 2011: 111101/tn4

A Note on Using C/C++ Compiler Package for M16C Series and R8C Family (M3T-NC30WA) V.6.00 Release 00

When using the C/C++ compiler package for the M16C series and the R8C family of MCUs (M3T-NC30WA) V.6.00 Release 00, take note of the following problem:

- With defining uncalled inline functions

---

1. Description

If an inline function is defined to which no call is made, Internal Error may arise in the linkage editor (optlnk), or external variables referenced by the uncalled inline function may not correctly be accessed by the debugger.

2. Conditions

2.1 Condition Group 1

This problem may arise if the following conditions are all satisfied:
(1) An inline function is defined.
(2) An external variable is referenced within the inline function in (1).
(3) No call is made to the inline function in (1), and the external variable in (2) is not referenced within the same file.

Note that if the external variable is placed in an include file, it is interpreted to be referenced within the same file.

Example:

```
-----------------------------------------------------------------
 extern int val;
 inline void func( void )  // Conditions (1) and (3)
```

```
  {
    val = 0x01;          // Conditions (2) and (3)
  }
-----------------------------------------------------------------
```

2.2 Condition Group 2

This problem may arise if the following conditions are all satisfied:

(1) Options -Ostatic_to_inline (-OSTI) and -Oforward_function_to_inline (-OFFTI) are used, or -OS_MAX (-OSM) is used.
(By using these options, static functions are treated as inline functions.)

(2) A static function is defined.

(3) An external variable is referenced within the static function in (2).

(4) The external variable in (3) is not referenced within the same file.
Note that if the external variable is placed in an include file, it is interpreted to be referenced within the same file.

(5) No call is made to the static function in (2), and the definition of the function is deleted by optimization.

Example:

```
-----------------------------------------------------------------
  extern int val;
  static void static_func( void )    // Condition (2)
  {
    val = 0x01;                 // Condition (3)
  }
-----------------------------------------------------------------
```

3 Workarounds

3.1 Workaround for Condition Group 1

Change the inline function involved to a static function.
Or, do not define any inline function to which no call is made.

3.2 Workaround for Condition Group 2

Deselect the -OFFTI option to suppress the deletion of the static function involved.
Or, do not define any static function to which no call is made.

4. Schedule of Fixing the Problem

We plan to fix this problem in the next version of the product.

[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.