RENESAS Tool News

**RENESAS TOOL NEWS on August 16, 2011: 110816/tn1**

# A Note on Using the C/C++ Compiler Package (M3T-CC32R) for the M32R Family of MCUs

When using the C/C++ compiler package (M3T-CC32R) for the M32R family of MCUs, take note of the following problem:

- With calling a function whose return value is cast to any type except void

## 1. Product and Versions Concerned

C/C++ compiler package for the M32R family
V.4.00 Release 1 through V.5.01 Release 01

## 2. Description

If a call is made to a function, its return value is cast to any type except void, and this type-cast value is not referenced, function calls may not be made properly when the program is executed.

### 2.1 Conditions

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

(1) The following optimizing options are used or not in compilation:
  (a) Any of the options -O4, -O5, -O6, -O7, and -O is used.
  (b) Either -Ospace or -Otime is used, and none of the options -O0, -O1, -O2, and -O3 is used.
(2) A function (A) calls another function (B).
(3) The return value of function B is cast to any type except void.
(4) The type-cast value in (3) is not referenced.
(5) Function B is called during or after execution of a construct with a conditional branch.
(6) In the call made to function B, a variable is referenced to use it for either of the following:

    (a) An argument of function B
    (b) the pointer to function B
(7) The variable referenced in (6) satisfies either of the following:
    (a) It is a parameter of function A, and its value is not changed by an assignment expression.
    (b) It is an automatic local variable or a parameter of function A, and the value of the variable referenced in (6) is the one assigned to the variable before the construct in (5) is executed.

## 2.2 Examples

Source program 1: sample01.c

```
------------------------------------------------------------------------
extern int cal_func01(int);
extern void dummy_func(void);
void func01(int a)          /* Condition (7a) */
{
   if (a) {                /* Condition (5) */
      dummy_func();
   }
   (void*)cal_func01(a+1);  /* Conditions (2), (3), (4), and (6a);
}                          (void*) satisfies Condition (3), and
                          variable a satisfies Condition (6a) */
------------------------------------------------------------------------
```

Source program 2: sample02.c

```
------------------------------------------------------------------------
extern short dummy_func(void);
void func02(int a)
{
   short (*p_func)(void);   /* Condition (7b) */
   p_func = dummy_func;     /* Condition (7b) */
   if (a) {               /* Condition (5) */
      (long)(*p_func)();   /* Conditions (2) (3), (4), and (6b);
   }                       (long) satisfies Condition (3),
}                          and variable p_func satisfies
                          Condition (6b) */
------------------------------------------------------------------------
```

Command line:

```
------------------------------------------------------
  cc32R -c -O7 sample01.c        Condition (1a)
  cc32R -c -Ospace sample01.c    Condition (1b)
-------------------------------------------------------
```

## 3. Workarounds

To avoid this problem, use either of the following methods:

(1) Cast the return value in Condition (3) to type void.

Source program 1 modified:

```
------------------------------------------------------------
extern int cal_func01(int);
extern void dummy_func(void);
void func01(int a)
{
    if (a) {
        dummy_func();
    }
    (void)cal_func01(a+1);  /* (void*) changed to (void) */
}
------------------------------------------------------------
```

(2) Do not cast the return value in Condition (3) to any type.

Source program 1 modified:

```
------------------------------------------------------------
extern int cal_func01(int);
extern void dummy_func(void);
void func01(int a)
{
    if (a) {
        dummy_func();
    }
    cal_func01(a+1);  /* Not cast to any type */
}
------------------------------------------------------------
```

(3) Assign the return value type-cast in Condition (3) to a variable.

Source program 2 modified:

```
------------------------------------------------------------
extern short dummy_func(void);
void func02(int a)
{
    short (*p_func)(void);
    long dummy;                 /* Variable defined */
    p_func = dummy_func;
    if (a) {
        dummy = (long)(*p_func)(); /* Assigned to variable */
    }
}
```

-----------------------------------------------------------------

(4) Suppress the optimization of Level 4.
To suppress the optimization of Level 4, use no optimizing options, or use any of the options -O0, -O1, -O2, and -O3.

When you use any optimizing option, avoid the problem as follows:
(a) If you are using -O4, -O5, -O6, -O7, or -O, replace it with -O0, -O1, -O2, or -O3.
(b) If you are using -Ospace or -Otime and none of the options -O0, -O1, -O2, and -O3, use -O0, -O1, -O2, or -O3 in addition.