RENESAS Tool News

RENESAS TOOL NEWS on August 1, 2007: 070801/tn1

A Note on Using the C/C++ Compiler Package V.5.01 Release 00 for the M32R Family of MCUs --With Using Initializing Expressions More Than Ten Thousand--

Please take note of the following problem in using the C/C++ compiler package--M3T-CC32R V.5.01 Release 00--for the M32R family of MCUs:

- With using ten thousand or more initializing expressions

---

1. Description

If ten thousand or more initializing expressions exist in a list of them enclosed with curry braces in a C/C++ source code, M3T-CC32R may malfunction and compilation be terminated with the following Windows' error message dispatched:

Windows' error message:

---

postpar.exe has encountered a problem and needs to close. We are sorry for the inconvenience.

---

2. Conditions

In the C++ and the C language, the conditions under which this problem occurs are different from each other.

2.1 In the C++ Language

This problem may occur if the following conditions are both satisfied:

(1) In the declaration of an array of type short or char (including unsigned or signed attached), there exists a list of initializing expressions containing about ten thousand or more of them.

(2) In lines before and after the list in (1) above, other declarations or functions are placed.

2.2 In the C Language

This problem may occur if the following conditions are both satisfied:

(1) In the declaration of an array of any type exists a list of initializing expressions containing ten thousand or more of them that are cast in their types.

(2) Either of the following conditions is met:

(a) In the declaration of an array in (1), the number of elements of the array is neglected.

(b) In a line after the list in (1) above, another declaration or a function is placed.

3. Examples

1. Source file sample1.cpp in C++

```
-----------------------------------------------------------------
unsigned short table1[] = {       /* Condition 2.1 (1) */
   0,
   112,
   93,
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   /* Thirty thousand initializing expressions */
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
};
int
check_table1(int index)          /* Condition 2.1 (2) */
{
   int ans;
   ans = table1[index];
   return ans;
}
-----------------------------------------------------------------
```

2. Source file sample2.c in C

```
-----------------------------------------------------------------
int table2[] = {              /* Conditions 2.2 (1) and (2)-(a) */
   (unsigned) 0,
   (unsigned) 100392,
   (unsigned) 33293,
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   /* Fifty thousand initializing expressions */
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
};
-----------------------------------------------------------------
```

3. Source file sample3.c in C

```
-----------------------------------------------------------------
int table3[50003] = {           /* Condition 2.2 (1) */
   (unsigned) 0,
   (unsigned) 100392,
   (unsigned) 33293,
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   /* Fifty thousand initializing expressions */
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
};
int
check_table3(int index)         /* Condition 2.2 (2)-(b) */
{
   int ans;
   ans = table3[index];
   return ans;
}
-----------------------------------------------------------------
```

4. Command line input

```
-----------------------------------------------------------------
% cc32R -S sample1.cpp
% cc32R -S sample2.c
% cc32R -S sample3.c
  NOTE: "%" denotes a prompt.
-----------------------------------------------------------------
```

4. Workarounds

Avoid this problem in either of the following ways:

(1) In C++, move the declaration of an array containing the list of initializing expressions to another C++ source file.

Modification of Example 1 (source file sample1.cpp):
Move the declaration of an array containing the list of initializing expressions to a newly created C++ source file, sample1init.cpp, for example; then in the original place, declare the array concerned to be extern with no initializing expressions.

sample1init.cpp (Newly created)

```
-----------------------------------------------------------------
unsigned short table1[] = {
```

```
 0,
 112,
 93,
. . . . . .
 /* Thirty thousand initializing expressions */
. . . . . .
};
```

---

sample1.cpp (After modified)

```
extern unsigned short table1[];
  /* Declared to be extern with no initializing expressions */
int
check_table1(int index)
{
   int ans;
   ans = table1[index];
   return ans;
}
```

---

(2) In C, do not cast the types of the initializing expressions.

Modification of Example 2 (source file sample2.c):
Remove all the cast expressions before the initializing expressions.

```
int table2[] = {
     0,                  /* Remove cast expressions */
   100392,                /* As above */
   33293,                 /* As above */
. . . . . .
 /* Fifty thousand initializing expressions */
. . . . . .
};
```

---

(3) In C, write the number of elements of the array in its declaration,
and move the declaration to the last of the source code.
This is the alternative to Workaround (2) above when it cannot be used.

Note that if Conditions (a) and (b) in 2.2 (2) are met, both the workarounds explained below for sample2.c and sample3.c must be taken.

Modification of Example 2 (source file sample2.c):
Do not neglect the number of elements of array table2 but declare it.

```
------------------------------------------------------------------
int table2[50003] = {       /* Declare the number of elements */
  (unsigned) 0,
  (unsigned) 100392,
  (unsigned) 33293,
      :
  /* Fifty thousand initializing expressions */
      :
};
------------------------------------------------------------------
```

Modification of Example 3 (source file sample3.c):
Move the declaration of array table3 with a list of initializing
expressions to the last of the function that follows, and
re-declare the array with no initializing expressions in the
original place.

```
------------------------------------------------------------------
int table3[50003];              /* Declare array only */
int
check_table3(int index)
{
   int ans;
   ans = table3[index];
   return ans;
}
int table3[50003] = {      /* Move original declaration here */
   (unsigned) 0,
   (unsigned) 100392,
   (unsigned) 33293,
      :
  /* Fifty thousand initializing expressions */
      :
};
------------------------------------------------------------------
```