RENESAS Tool News

RENESAS TOOL NEWS on April 1, 2006: RSO-R0C00000SCW01R-060401D

Notes on Using the MISRA C Rule Checker SQMlint

Please take note of the following problems in using the MISRA C rule checker SQMlint:

- On delivering incorrect messages when Rule 43 violated
- On delivering incorrect messages that tell Rule 1 has been violated at initialization of global variables

1. Problem on Delivering Incorrect Messages When Rule 43 Violated

1.1 Versions Concerned
SQMlint V.1.00 Release 0--V.1.02 Release 00

1.2 Description
If a violation against MISRA C Rule 43 is detected in an initialization expression (an implicit type conversion is made to cause information loss), the type before conversion is displayed in place of the type after conversion in the error messages that are delivered.

Example:

```
   [MISRA(43) Complaining] information loss conversion
(from 'signed
      long' to 'signed long')
```

In this example, the second 'signed long' must be 'unsigned long'.

NOTE: The name of the option to validate the check of Rule 43 varies according to compilers.

Example of violation against MISRA C Rule 43 in an initialization expression:

```
 ----------------------------------------------------
-------
 void func(void)
 {
    char* p1;
    char* p2;
    unsigned long ul = p1 - p2;      /* Violates Rule 43 */
 }
 ----------------------------------------------------
-------
```

1.3 Workaround

Read "the correct type after conversion" for "the incorrect type after conversion" in message telling violation against Rule 43.

1.4 Schedule of Fixing the Problem

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

2. Problem on delivering incorrect messages that tell Rule 1 has been violated at initialization of global variables

2.1 Versions Concerned

SQMlint V.1.00 Release 0--V.1.02 Release 00

2.2 Description

When global variables are initialized, the following message telling violation against Rule 1 may be delivered though it is not to be done.

2.3 Conditions

This problem occurs if the following conditions are both satisfied:

(1) A structure or union having an array as its member is defined.

(2) A global variable that has the address of an element of the array in (1) as its initializer is

declared.

Example:

```
 ---------------------------------------------------------
------------
 struct S {
    char ary[5];                    /* (1) */
 } s;

 const char* p = &s.ary[3];         /* (2) */
 ---------------------------------------------------------
------------
```

2.4 Workaround

Neglect the messages incorrectly telling violation against Rule 1.

2.5 Schedule of Fixing the Problem

We plan to fix this problem in the next release 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.