

A Note on Using the C/C++ Compiler Package V.5.00 Release 00 for the M32R MCU Family

Please take note of the following problem in using the C/C++ Compiler Package (M3T-CC32R) V.5.00 Release 00 for the M32R MCU family:

- With compiling programs written in C/C++ language that contain asm functions with the -cs option selected

1. Description

If any program written in C/C++ language that contains asm functions is compiled with the -cs option selected, the object module file (with filename extension .mo) will be created with no asm functions outputted. At the same time, malfunctioning branch instructions will be generated for the selection and iteration statements (if, for, while, do, and switch) and the controlling (relational, equality, logical and conditional) expressions (a==b && c!=d, for example) within functions in the C/C++ program. However, the source file in assembly language containing C/C++ source lines (with filename extension .cs) will be created properly.

2. Conditions

This problem occurs if the following conditions are all satisfied:

- (1) The compile option -cs selected.
- (2) In the program exists a function containing an asm function.
- (3) The "#pragma keyword asm on" directive is applied to the asm function in (2), so the capability of inline-assembling the asm function is valid.

Examples:

1. sample1.c

#pragma keyword asm on /* Condition (3) */

```
void set_psw(unsigned long psw) /* Condition (2) */
{
    asm(" MVTC R0,PSW", psw); /* Condition (2) */
}
-----
```

2. sample2.c

```
void wait_by_asm_nop(int count) /* Condition (2) */
{
    int i;
    for (i = 0; i < count; i++) {
#pragma keyword asm on /* Condition (3) */
        asm(" NOP"); /* Condition (2) */
    }
}
-----
```

Examples in command lines (Symbol % denotes a prompt.)

```
% cc32R -cs sample1.c      : Condition (1)
% cc32R -cs sample2.c      : Condition (1)
-----
```

3. Workarounds

This problem can be avoided in either of the following ways:

- (1) Instead of compiling the program once using the `-cs` option, do it twice using the `-c` and the `-CS` option, each once.

Example of the command-line input using `-cs`:

```
% cc32R -cs -Ospace -I include file.c
```

Example of the command-line inputs using `-CS` and `-c`:

```
% cc32R -CS -Ospace -I include file.c
% cc32R -c -Ospace -I include file.c
```

- (2) Use internal controlling option `-Qa-XL*` together with `-cs`

Example of the command-line input using `-cs` only:

```
% cc32R -cs -Ospace -I include file.c
```

Example of the command-line input using `-Qa-XL` with `-cs`:

```
% cc32R -cs -Qa-XL -Ospace -I include file.c
```

*Be sure to use -Qa-XL together with -cs.

4. 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.

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