To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Date: Oct.27.2004

RENESAS TECHNICAL UPD

Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan
RenesasTechnology Corp.

Product Category	User Development Environment		Document No.	TN-CSX-A086A/E	Rev.	1.0
Title	SuperH RISC engine C/C++ compiler package V.8.00 Release 04 Updates		Information Category	Technical Notification		
Applicable Product	R0C40700XSW08R (P0700CAS8-MWR) R0C40700XSS08R (P0700CAS8-SLR) R0C40700XSH08R (P0700CAS8-H7R)	Lot No. All	Reference Document	SuperH RISC engine C/C++ Compiler Assembler Optimizing Linkage Editor User's Manual REJ10B0047-0100H Rev.1.00		

SuperH RISC engine C/C++ compiler package is updated in V.8.00 Release 04.

See the notes below if you have the compiler package listed in the following table.

Part No.	Package version	Compiler version
P0700CAS8-MWR	8.0.00 8.0.01	8.0.00 8.0.01
R0C40700XSW08R	8.00 Release 01 8.00 Release 02 8.00 Release 03	8.0.01 8.00.02 8.00.03
P0700CAS8-SLR	8.0.00 8.0.01	8.0.00 8.0.01
R0C40700XSS08R	8.00 Release 01 8.00 Release 03	8.0.01 8.00.03
P0700CAS8-H7R	8.0.00 8.0.01	8.0.00 8.0.01
R0C40700XSH08R	8.00 Release 01 8.00 Release 03	8.0.01 8.00.03

If you have the compiler package of the Windows® version, download the update program from the following URL:

http://www.renesas.com/eng/products/mpumcu/tool/index.html

If you have the compiler package of the UNIX version, request the update program to an authorized product distributor.

The contents of updates in this package are shown below.

Descriptions of section 1 only apply to the Windows® version.

1. High-performance Embedded Workshop (Windows® version)

1.1 Function of Generating Make Files Improved

In relation to the dependencies of include files on a source file, directory information on include files has hitherto been output to a Make file using relative path names if the nesting levels of include files are less than 11. In this revision, this limitation is raised to less than 51 in the nesting levels. If they are equal to or greater than 51, absolute path names used.

1.2 Other Problem Fixed

The following known problems have been fixed:

(1) If comments in Japanese are included in a source line, a jump made from the Navigation window to the line in the Editor window that declares or defines a source file may reach an incorrect line locating several lines after the destination.

(2) If a project that contains source files written in assembly language is opened with the navigation function being activated, the automatic analyses of the navigation function will not terminate.

2. Compiler

2.1 Comparison of a result of subtraction and zero (SHC-0004)

We have changed so that bit size after the cast was used when overflow occurred in conversion if 1- or 2-byte same type variables are subtract expression, and the result was converted to the signed type of same size and compared with zero,

[Example]

```
extern void g(void);
unsigned short a,b;
f()
{
    short t = a - b;  // If a = 0xffff and b = 0x0000, then t = 0xffff.
    if (t > 0) {        // The result of comparison was false.
        g();
    }
}
```

3. Optimizing Linkage Editor

3.1 A new option

A following option was introduced.

(1) DAta_stuff

Reduces empty areas generated as the boundary alignment of sections after compilation.

3.2 Incorrect debug information by optimization at linkage

Fixed the problem that incorrect debug information was outputed by optimization at linkage, and a debugger can not show correctly.

[Conditions]

The problem might occur when all of the following conditions were fulfilled.

(1) Both debug and goptimize options were specified as compiler options.

(2) Some optimization of linker was valid.

(3) The difference between lines in debug line information became 255 lines by optimization of (2).

3.3 Illegal unifying same codes optimization

Fixed the problem that incorrect object codes were outputed by unifying same codes optimization(optimize=same_code).

The problem might occur by linker version 8.00.03 or later.

[Conditions]

The problem might occur when all of the following conditions (1)-(3) were fulfilled.

(1) A goptimize option was specified as compiler options.

(2) The unifying same codes optimization (optimize=same_code) was valid.

(3) Either of a following condition was fulfilled.

(a) An entry function (specified "#pragma entry" or entry option) was defined between one function and other functions. And these functions were affected by the optimization of (2).

[Example]

```
#pragma entry f_entry

void f_a(){
    :
}

void f_entry(){      // entry function
    :
}

void f_b(){          // following function was optimized
    :
}
```

(b) Some relocatable files were input. And the function which a literal pool does not exist next the code of end of a function was affected by the optimization of (2).

(c) Some relocatable files were input. And code size of some function which was in the relocatable file was reduced 4n+2 (2,6,10,...) bytes by the optimization of (2).

(d) Two functions were defined continuously (function "A" and "B") were optimized by the optimization of (2). And code size of the function "A" was reduced 4n+2 (2,6,10,...) byte by the optimization.