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: Apr.07.2004

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-068A/EA	Rev.	1.0
Title	H8S, H8/300 Series C/C++ compiler package Ver.6.0.01 Updates		Information Category	Specification Change		
Applicable Product	PS008CAS6-MWR PS008CAS6-SLR PS008CAS6-H7R	Lot No. All	Reference Document	H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual REJ10B0058-0100H Rev.1.0		

H8S, H8/300 Series C/C++ compiler package is updated in Ver.6.0.01.

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

Part No.	Package version	Compiler version
PS008CAS6-MWR	6.0.00	6.0.00
PS008CAS6-SLR	6.0.00	6.0.00
PS008CAS6-H7R	6.0.00	6.0.00

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 Navigation function

Category "C++ Classes" of the navigation function is unchecked as a default.

1.2 Problem fixed at HEW termination

We have fixed the problem that the HEW did not quit normally.

1.3 Confirmation Message for Re-reading the File

The problem of the confirmation message for re-reading the file not being displayed when a file currently open in the editor window is edited by the external editor, has been corrected. (This problem only occurs with Ver. 3.0.02)

1.4 Problem with the Search Function from the Memory Menu

The problem of an empty [Memory] window being displayed when carrying out a memory search selected from [Memory->Search...] in the menu bar, has been corrected.

1.5 Symbol Load Command

The problem with the Symbol Load command not operating has been corrected.

1.6 Problem of Disappearance of Project Types to Build the Software

The problem of disappearance of project types to build the software when the emulator was installed to the same folder where SHC Ver 8.0.00/8.0.01 or H8C Ver 6.0.00 compiler package had been installed, has been corrected.

1.7 Illegal Command in the TCL Procedure

The problem of the command not operating properly after specification of the prefix (H', D', O', or B') for a parameter in the TCL procedure(defined by the proc command), has been corrected.

[Example]

```
proc mem_read {} {
md H'FFEF80 H'32 LONG
}
```

2. Compiler

2.1 RENESAS TECHNICAL UPDATE TN-CSX-065A/EA

The following problem was fixed.

a) Incorrect object by division of an unsigned variable by a constant

b) Incorrect structure with the initial value

c) Incorrect bit field setting

d) Incorrect value setting with using a 3-bytes structure

e) Incorrect elimination of a function call

f) Incorrect result of operation on a two-dimensional array address

g) Incorrect setting and reference of bit field

h) Incorrect replacement of a loop control variable

i) Incorrect constant propagation

j) Incorrect unification of string data

k) Cpuexpand option was not enabled

l) Incorrect substitution with using 4 or less byte structure

2.2 Internal error

An internal error under the following conditions was fixed.

a) The loop had an fixed reference to an array and its index value was a constant that exceeded the range of the int (short) type (C4098).

b) An expression that included an operation of a variable and a constant was described in the index of an array, which was used as the condition for a conditional statement (C4098).

c) At initialization of a union of a local variable, the first member of the union was an array or a structure (C4774).

d) When there were consecutive loops and the first statement of the first loop and the initialization condition of the second loop were the same, the initial value assignment in the second loop was illegally integrated (C4098).

e) 500 or more variables, with or without specification of abs8/abs16, were declared (C4712).

f) When overflow occurs in the 1-byte subtraction of a constant(C4722).

2.3 Comparison from an operation with an overflow

The problem mentioned as "Comparison from an operation with an overflow" in notes of the H8S, H8/300 series C/C++ compiler ver.6.0.00 was fixed.

2.4 Change specification of cpuexpand option

When the cpuexpand option is specified and the expression is one of followings, the H8SX CPU does not apply this expression for expanded interpretation, even though the other CPU applies for.

a) signed long = signed int << constant

b) signed long = unsigned int << constant

c) unsigned long = signed int << constant

d) unsigned long = unsigned int << constant

e) signed int = (signed int << constant) / signed int

f) signed int = (unsigned int << constant) / signed int

g) signed int = (unsigned int << constant) / unsigned int

h) unsigned int = (signed int << constant) / signed int

i) unsigned int = (unsigned int << constant) / signed int

j) unsigned int = (unsigned int << constant) / unsigned int

3. Assembler

3.1 Incorrect address column value

The following problem was fixed.

When "locate" was described in the .SECTION directive the debugging information might be incorrect.

[Conditions]

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

a) The optimize option was specified.

b) Described "locate" in the .SECTION directive in a source file.

4. Optimizing linkage editor

4.1 Unnecessary error generated when sections are overlaid

Fixed the problem that outputs an unnecessary error message (L2022) appears when the following conditions are satisfied at the same time.

[Conditions]

a) Uninitialized data sections are overlaid in the start option.

b) The form=binary (or hexadecimal or stype) option is specified.

4.2 Illegal object due to specification of indirect addressing mode optimization

The following problem has been fixed.

When indirect addressing mode optimization is specified, the address indicated by a symbol may be 0x00000000 incorrectly.

[Conditions]

The problem may occur when all of the following conditions are satisfied.

a) Indirect addressing mode optimization (optimize=function_call) is valid.

b) A number of object files link to a relocatable file.

c) Object files of b) are comprised of those with or without goptimize option.

d) The relocatable file of b) was input.

e) Other files (relocatable file, object file, or libaray file) are input along with the relocatable file of b).

4.3 Fixed internal error

a) Fixed is the problem in which an internal error (8996) may occur when optimization regarding shortening of the addressing mode has been specified.

b) Fixed is the problem in which an internal error (L4001) may occur when optimization regarding unifying of same codes has been specified.