RENESAS

High-performance Embedded Workshop V.4.09

Release Note

Contents

1. Notes on Using the High-performance Embedded Workshop

This section shows notes on using the High-performance Embedded Workshop.

1.1 Network drive

1.1.1 Time discrepancies between computers

The time kept by a computer differs between computers. The time when a source file or output file is updated depends on the computer on which the file has been saved.

Building of a project can be incorrect because of this difference in times kept by computers if a source file or output file is shared via a network.

If such a problem occurs, adjust the time between all computers or use [Build -> Build All] to build a project.

1.1.2 Building a project on a network drive

When you build a project on the network an error might occur depending on the condition of the network. For example, a C/C++ compiler might output the following error message.

```
C3019(F) Cannot open source file
```

If such an error occurs, build the project again.

1.1.3 Adding a file to a project

When you add a file on a remote drive on the network via [Project -> Add Files] or else, the High-performance Embedded Workshop might cause an application error depending on the condition of the network.

In this case, try to add the file to the project again. If application errors occur many times, consider copying the file to a local drive then adding to the project.

1.2 Syntax coloring of the editor

With the syntax coloring capability of the High-performance Embedded Workshop, comments and keywords in a source program file are colored. But the syntax coloring of the High-performance Embedded Workshop editor does not work correctly depending on the way the source code is written.

Disable the syntax coloring if you are not satisfied with the syntax coloring. To disable it, select [Setup -> Options], uncheck "Enable Syntax Coloring" on the [Editor] tab of the [Options] dialog box, and click on the OK button.

Syntax coloring for C/C++ source files does not work correctly in the following cases:

(i) A comment is nested.

```
Example: /* /*    */ */ <- The underlined part is colored as a comment.
```

(ii) "/*" or "*/" is contained in a string.

```
Example: /*              <- The underlined part is colored as a comment.
         char A[]="*/"; <- The underlined part is colored as a comment.
         */
```

To prevent this problem, avoid writing * or / characters adjacent to each other (unless such characters are part of a comment) in a C/C++ source file.

1.3 File dependencies scanned

The High-performance Embedded Workshop scans an include file of a source file to show dependencies of source files on the [Projects] tab of the [Workspace] window and to decide whether to build a file or not.

Even though the source code is syntactically correct, an error message may appear indicating that an include file has not been correctly recognized as a dependent file.

In such cases, (a) build all files by selecting [Build -> Build All] or (b) select the source file in the [Projects] tab of the [Workspace] window and select [Build -> Build (file name)].

Include files written in C/C++ source files are not correctly detected as dependent files in the following cases:

(i) "/*" or "*/" is contained in a string.

```
Example: char A[] = "/*";  <- Assumed as the beginning of a comment.
         #include "file.h" <- This file is not detected as a dependent file.
         char B[] = "*/";  <- Assumed as the end of a comment.
```

To support preprocessor statements such as #if, #ifdef, and #define by the High-performance Embedded Workshop, go to the [Build] menu to open the toolchain option dialog box. On the [Toolchain Option] tab, check "Support dependency scan of preprocessor statement".

Table 1.1 shows the preprocessor statements that are currently supported.

Table 1.1   Preprocessor Statements

Preprocessor Statement	Description
#define	Defines an identifier. When any preprocessor statements include the identifier, this identifier is replaced with the defined character string. e.g. #define NICE_FILE "nice.h" #include NICE_FILE
#undef	Disables replacement of the defined identifier.
#include	Informs the High-performance Embedded Workshop that the specified file is dependent on the current source file.
#if , #else, and #endif #elif	When #if <expression> is entered, the High-performance Embedded Workshop only analyzes the range of source code that satisfies the expression. When the value resulted from the expression is 0: #if to #else When the value resulted from the expression is 1: #else to #endif
defind macro	Entering defined (<identifier>) returns 1 when the identifier has already been defined. Otherwise 0 is returned. This macro is usually written as #if defined (MACRO).
#if def #ifndef	The High-performance Embedded Workshop assumes an #ifdef <identifier> statement as #if 1 when the identifier has already been defined. Otherwise #if 0 is assumed. An #ifndef <identifier> statement, on the other hand, is assumed as #if 0 when the identifier has already been defined, and otherwise #if 1.
#line	Ignored
#error	Ignored
#pragma	Ignored
#	Ignored

When a preprocessor statement has an expression, the following ten kinds of operators are currently supported with the dependency check. The dependency search does not operate correctly if an operator other than those listed below is used.

```
( ), !, <, <=, >, >=, ==, !=, &&, and ||.
```

RENESAS

When [Support dependency scan of preprocessor statement] has been selected, include files written in C/C++ source files are not correctly detected as dependent files in the following cases:

(ii) The toolchain included in either of the compiler packages listed below is in use and <expression> in a preprocessor statement contains a conditional statement for comparing characters.

- C/C++ Compiler Package for SuperH Family V.5.1 to V.9.03 Release 02
- C/C++ Compiler Package for H8SX, H8S, H8 Family V.3.0A to V.7.00 Release 00
- C/C++ Compiler Package for RX Family V.1.00 Release 00
- C Compiler Package for M16C Series and R8C Family V.5.30 Release 0 to V.5.45 Release 01
- C Compiler Package for M32C Series V.5.40 Release 0 to V.5.42 Release 00
- C Compiler Package for R32C/100 Series V.1.01 Release 00 to V.1.02 Release 01
- C/C++ Compiler Package for M32R Family V.5.00 Release 00 to V.5.01 Release 01
- C Compiler Package for 740 Family V.1.00 Release 1 to V.1.01 Release 02

```
Example: #define A 'a'
         #if (A == 'a')        <- This leads to an error and include files
                                  written on the subsequent lines will not
                                  be detected as dependent files.
            #include "file.h" <- This file is not detected as a dependent
         #endif                  file.
```

(iii) The toolchain included in either of the compiler packages listed below is in use and the operand of assembly code for the function is an immediate value.

- C/C++ Compiler Package for SuperH Family V.5.1 to V.9.03 Release 02
- C/C++ Compiler Package for H8SX, H8S, H8 Family V.3.0A to V.7.00 Release 00
- C Compiler Package for M16C Series and R8C Family V.5.30 Release 0 to V.5.45 Release 00
- C Compiler Package for M32C Series V.5.40 Release 0 to V.5.41 Release 01
- C Compiler Package for R32C/100 Series V.1.01 Release 00 to V.1.02 Release 00
- C/C++ Compiler Package for M32R Family V.5.00 Release 00 to V.5.01 Release 01
- C Compiler Package for 740 Family V.1.00 Release 1 to V.1.01 Release 02

```
Example: static void Change_PSW_PW_to_UserMode(void)
         {
            MVFC PSW,R1
            OR  #00100000h,R1 <- This leads to an error and dependent files
               :                 written on the subsequent lines will not
         }                       be detected as dependent files.
```

(iv) The toolchain included in either of the compiler packages listed below is in use and the operand of assembly code for the function is a constant integer with a radix.

- C/C++ Compiler Package for SuperH Family
- C/C++ Compiler Package for H8SX, H8S, H8 Family

```
Example: void func(void)
         {
         #pragma asm
            AD1: .EQU H'7FFF  <- This leads to an error and dependent files
               :                 written on the subsequent lines will not
         #pragma endasm         be detected as dependent files.
               :
         }
```

Include files written in assembly source files are not correctly detected as dependent files in the following case:

(v) A source file for the AS30, AS308, or AS100 contains directive "..FILE" or "@" for specifying an include file.

```
Example: <sample.a30 file>
        .INCLUDE ..FILE@.inc <- This leads to an error and sample.inc
                                will not be detected as dependent files.
```

1.4   Navigation facilities

1.4.1   Navigation to the definition of C functions

The High-performance Embedded Workshop scans the definitions of C functions in a C/C++ source file and displays tags of the definitions on the [Navigation] tab of the [Workspace] window so that double-clicking on a tag will navigate you to the line of the definition in a corresponding file shown in the editor.

When the High-performance Embedded Workshop searches for the definitions of C functions, it ignores all preprocessor directives. The High-performance Embedded Workshop thus may not be able to correctly analyze the definitions of C functions.

For example, two definitions of func() might be displayed on the [Navigation] tab of the [Workspace] window in the following code.

```
#define DEF 1
#ifdef  DEF
void func(void)
{
}
#else
int func(int a)
{
}
```

1.4.2   Default setting for "C++ Classes" of the Navigation facilities

"C++ Classes" of the navigation facilities is disabled (unchecked) at default.

You can check the setting in the [Select Categories] dialog box (figure 1.1).

To open the [Select Categories] dialog box, right-click on the window of the [Navigation] tab and then select "Select Categories".
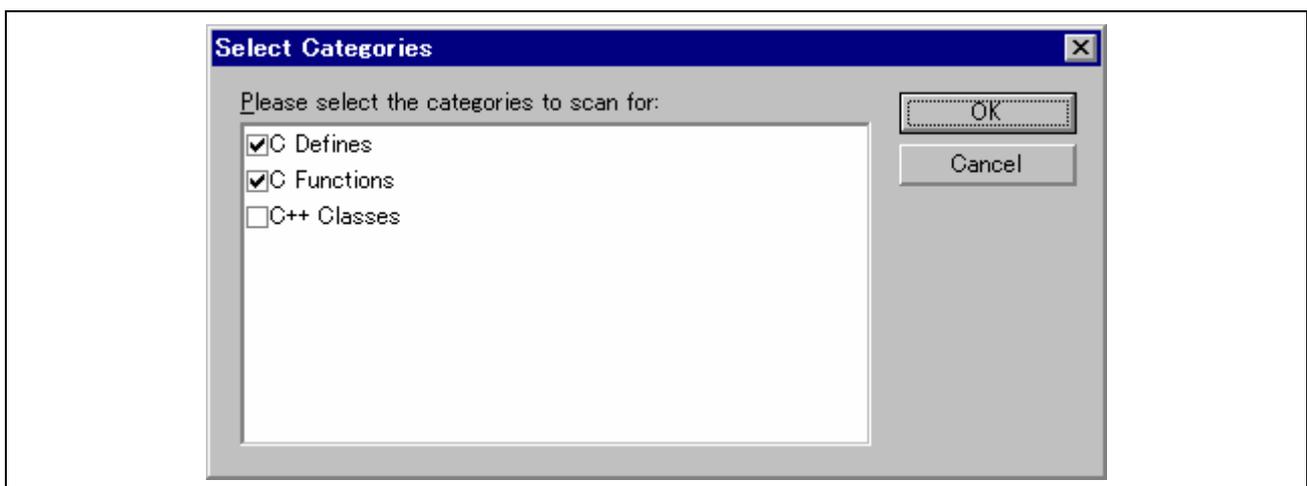
Figure 1.1   Select Categories Dialog Box

1.4.3   Warning message by "C++ Classes" of the Navigation facilities

When "C++ Classes" of the navigation facilities is enabled (checked), the warning message will appear (figure 1.2).

While this item is enabled, the navigation facilities run in the background. The CPU usage of your host computer may be shown as nearly 100% (depending on the performance of the computer) and the reaction of the High-performance Embedded Workshop may be slow.

Figure 1.2   Warning Message

1.5   Expression

(1) A function name of the C++ program cannot be specified for the expression symbol.

(2) No overloaded operator can be used as a function name.

1.6   Parallel Build

When a parallel build is executed, source-code messages (===> void …) output in the Output window by M16C, R8C C/C++ compiler V.6.00 may not be accompanied by the corresponding error messages as shown below.

```
C:\Workspace\sample\initsct.c(31) : C2570 (E) invalid function declare
C:\Workspace\sample\fvector.c(39) : C2766 (E) parse error at near '_asm'
===> void initsct(void)
===> _asm(" .id ""¥"#FFFFFFFFFFFFFF¥"");
```

1.7   Opening a file which caused an error

This is a note when you double-click on an error/warning message of the C/C++ Compiler or the Assembler on the [Output] window and open the corresponding line of the corresponding file.

If the corresponding file window, however, has been minimized in the editor window area, the file is not opened on double clicking on the error/warning message.

In this case, restore the file window or maximize it.

1.8   Specifying the directory on an Option dialog box

Using the [Build] menu of the High-performance Embedded Workshop, you can launch an option dialog box on which you can specify options to a tool such as a compiler.

Especially, when "Custom directory" is selected in the "Relative to" field on a dialog box as shown in figure 1.3, an absolute path must be specified in the "Directory" field.
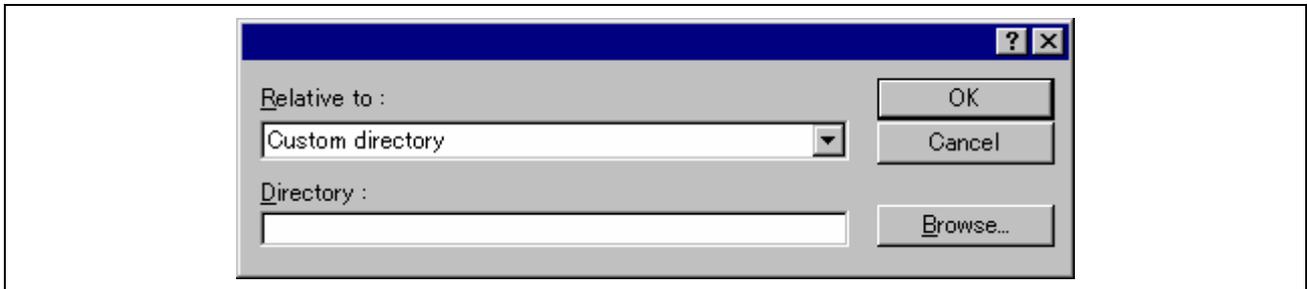
Figure 1.3   Dialog Box to Specify a Directory Path

1.9   Files to be added to projects

It is not possible to add multiple files with the same name (excluding the path and filename extension) to a single project. Save these files with separate names before adding them to a project.

1.10   Replacing and finding character strings

1.10.1   Replacing 2-byte character codes

Select [Edit -> Replace] from the menu to bring up a dialog box. If you type a 2-byte character code into the [Find what] or [Replace with] text box, select the [Regular expression] check box, and click on the [Replace] button, you will see the code replaced with an incorrect character string.

Even the [Replace All] button cannot replace 2-byte character codes correctly.

To prevent this problem, deselect the [Regular expression] check box (disable the regular expression) before replacing a 2-byte character code.

1.10.2   Finding 2-byte character codes

Select [Edit -> Find] from the menu to bring up a dialog box. If you type a 2-byte character code into the [Find what] text box, select the [Regular expression] check box, and click on the [Find] button, you may obtain an incorrect search result. The character string that has been found is highlighted in the [Source] window. This highlighted string and strings before and after it may have been replaced with incorrect ones. If you save the source file, these character strings will remain incorrect.

To prevent this problem, deselect the [Regular expression] check box (disable the regular expression) before searching for a 2-byte character code.

1.11   Editor window

(1) Character sets

The editor window does not support Unicode.

1.12 Tools Administration dialog box

1.12.1 Uninstallation of components and tools

For High-performance Embedded Workshop V.4.05 and later versions, the [Uninstall] button has been removed from the [Tools Administration] dialog box.

Thus you cannot uninstall components or tools individually.

If you wish to uninstall any component or tool, uninstall the whole set of the High-performance Embedded Workshop through the Windows® control panel.

1.12.2 Deregistration of components and tools

After you have deregistered a component or tool from the High-performance Embedded Workshop by clicking on the [Unregister] button in the [Tools Administration] dialog box, the component or tool will not be registered to the High-performance Embedded Workshop even if you install a software product including that component or tool.

To register the component or tool, use the [Register] button on the [Tools Administration] dialog box. The [Tools Administration] dialog box must be invoked before a workspace is opened.

Select [Administration] from the [Tools] menu to open the [Tools Administration] dialog box.

1.13 Opening a workspace file saved in an older format

V.4.04 and later versions of the High-performance Embedded Workshop allow workspace files to be saved in older formats. You can use a prefix to save workspace file as a separate file. When you open a workspace file saved in an older format, note the following points.

- If multiple projects belong to the workspace, the one that appears first is treated as the current project.
- The files previously opened in the main window are not automatically opened.

1.14   Debugging facility (common to all debuggers)

1.14.1   Source-level execution

Even standard C libraries are executed by [Step In]. To return to a higher-level function, click on the [Step Out] button.

In a "for" statement or a "while" statement, executing a single step does not move to the next line. To move to the next line, execute two steps.

If [Step Out] is executed within an inline function, the program counter (PC) returns to the caller of the non-inline function which called that inline function.

```
Example: When main() calls Func(), which calls inline InFunc()

    10    int a = 0;
    11    main()
    12    {
    13        Func();
    14    }
    15
    16    void Func()
    17    {
    18        InFunc();
    19    }
    20
    21    inline void InFunc()
    22    {
    23        a++;
    24    }
```

When [Step Out] is executed on line 23, the PC returns to line 14. That is, the result is the same as when [Step Out] is executed within Func() (i.e. the PC does not stop at line 19).

1.14.2   Moving source file position after creating load module

When the source file is moved after the load module has been created, the [Open] dialog box, which specifies the source file, may be displayed during debugging the created load module.
Select the corresponding source file and click on the [Open] button.

1.14.3   Editor window

After modifying the program displayed in the editor window and reloading the source files and load modules, close the editor window then open it again; otherwise the program displayed in the window may not be correct.

1.14.4   Disassembly window

When [Step] or [Go] is executed while the disassembly and editor windows show data in source mode and mixed mode, respectively, the program counter (PC) icon (⮕) may be placed at an incorrect position in the disassembly window.
If you wish to start debugging with source mode selected in the disassembly window, be sure to select source mode in the editor window.

1.14.5   Stack Trace window

When the stack trace window is opened after an interrupt function has been executed, display of data before the interrupt function is incorrect.

1.14.6 Memory window

(1) Finding a value in memory

Be sure to select a range with a mouse or click on a specific location in the [Data] or [Code] column in the [Memory] window before starting to search for a value in memory. If you just right-click on the [Memory] window and start searching for a value, the found value will not be highlighted.

(2) Fonts

When a proportional font is selected, part of the characters in the view may be hidden. Fixed fonts are recommended.

1.14.7 Entry point

Even when an entry point is specified by the ENTRY option of the linkage editor, the entry point address is not set for the PC during program downloading. Set the PC before program execution.

1.14.8 Register window

In High-performance Embedded Workshop V.4.04 and later versions, values in registers are not saved in sessions.

1.15 Debugging facility (SuperH family, H8X, H8S, H8 family, and RX family debuggers)

1.15.1 Line assembly

Regardless of the Radix setting, the default for line assembly input is decimal.
Specify H' or 0x as the radix for a hexadecimal input.

1.15.2 Loading of SYSROF format file

The debugging program of the SYSROF format cannot be loaded. Please make the debugging program by the ELF/DWARF2 format.

1.15.3 Session file of Hitachi Debugging Interface

The session file of Hitachi Debugging Interface cannot be used.
Please make the file as project workspace of High-performance Embedded Workshop.
The command file of Hitachi Debugging Interface can be used.

1.15.4 Profiler

Profiler does not support overlay function.

1.15.5 Stack Trace window

Inline functions that have been expanded as intended are not displayed in the history of function calls.

RENESAS

1.15.6   Reference of variables
- The Stack Trace window
- The Locals window
- The Watch window
- Tooltip Watch function
- Instant Watch function

(1) Local variables in inline functions

While the PC is pointing to an address in an inline function, you cannot use the windows or functions given above to reference the parameters of functions or the values of local variables defined in functions.

(2) Local variables after optimization

Depending on the generated object code, the windows and functions given above do not correctly show the values of local variables in a C source file that is compiled with the optimization option enabled.
Check the generated object code in the [Disassembly] window.

(3) Memory caching

With memory caching enabled, the windows and functions given above may not display the correct values of variables while a program using little endian is being debugged. For example, the 4-byte variable at address 0x2002 in the following code will not be displayed correctly.

```
#pragma address data=0x2002
long data;
```

To prevent this problem, use the CACHE command to disable caching.
For notes on the CACHE command, refer to section 1.19.1, CACHE command.

```
cache off
```

1.15.7   Viewing the value stored at the memory address indicated by a char-type variable pointer
- The Locals window
- The Watch window

The above windows simply display the message "Not available now." rather than the value stored at the memory address indicated by a char-type variable pointer (see example 2) when the address satisfies either (i) or (ii).

(i) 0xFFFFFF81 or higher
(ii) Within 128 bytes from the end address of any area in the memory map
Areas within 128 bytes from the end address of the memory map:
```
0x00000000 to 0x00007FFF, 0x00007F81 to 0x00007FFF
0xF0000000 to 0xF000FFFF, 0xF000FF81 to 0xF000FFFF
0xFFFF8000 to 0xFFFFFFFF, 0xFFFFFF81 to 0xFFFFFFFF
```

To avoid this problem, in the [Watch] window, enter an expression that directly represents the address indicated by the pointer (see example 3).

```
Example 1:  Addresses that satisfy both (i) and (ii) (on sixth and
            eighth lines)
 01    char* cp = (char*)0xFFFFFF80;
 02    void main(void)
 03    {
 04         *cp = '0';    // Assign 0x30 to 0xFFFFFF80
 05         cp++;
 06         *cp = '1';    // Assign 0x31 to 0xFFFFFF81
 07         cp++;
 08         *cp = '2';    // Assign 0x32 to 0xFFFFFF82
 09    }

Example 2:  [Watch] and [Local] windows
  - "cp" 0xffffff80 { 00001400 } (char*)
            // Referable
    - "*"  H'30 "0" { FFFFFF80 } (char)
  - "cp" 0xffffff81 { 00001400 } (char*)
            // Address corresponds to description (i) and (ii)
    - "*"  Not available now.
  - "cp" 0xffffff82 { 00001400 } (char*)
            // Address corresponds to description (i) and (ii)
    - "*"  Not available now.

Example 3:  Expressions to be entered in the [Watch] window
    - "*cp" H'31 "1" { FFFFFF81 } (char)
    - "*cp" H'32 "2" { FFFFFF82 } (char)
```

1.15.8   Changing the values of variables

- The Locals window
- The Watch window

(1) Variables allocated to registers

In the above windows, the size of each variable type is used for changing the value of the variable. If the size of a variable type does not match that of the register to which the variable is allocated, an incorrect value may be set in the register.

```
Example: When a one-byte variable is allocated to a four-byte register
   If a negative value is specified for the one-byte variable through
   the debugger, no sign-extended value is set in the upper three
   bytes of the register.

   int func() {
      int res;
      signed char ch;        // Reference register "R2" (*1)
      ch = -1;               // Assign -1 to variable "ch" (*2)
      if (ch == -1) {
         res = 0;
      } else {
         res = 1;
      }
      return res;
   }
```

```
      *1. The initial value of "R2" is 0x12345678.
      *2. - When -1 is assigned to "ch" through the program
              "R2": 0x12345678 -> 0xffffffff
            - When -1 is assigned to "ch" through the [Watch] window
              "R2": 0x12345678 -> 0x123456ff
```

To avoid this problem, use the [Register] window to change the values of registers.

1.15.9   Watch window

(1) Expanding variables

When a watch item is located outside the scope in debugging of a program, the item is automatically collapsed if it has been expanded.
The realtime update setting on the first child item applies to the entire watch item.

(2) Automatic update of variables

When a variable has been cast and then added to the Watch window, automatic update of the variable is not possible.

1.15.10   Configure Overlay dialog box

This dialog box may look as if a section adjacent to an overlay section is also adjacent to another overlay section.
If overlay sections are allocated as (1), (2), and (3) in example 1 and then re-allocated after build as in example 2, the [Configure Overlay] dialog box will be as shown in example 3.

```
Example 1:
   <Section>
   (1) "P11, P12"
   (2) "P21"
   (3) "P31, P32"

Example 2:
   <Sec> <Start - End >
   "P11" 0x1000 - 0x10FF (*1)
   "P12" 0x1100 - 0x12FF
   "P21" 0x1000 - 0x10FF (*1)
   "P31" 0x1000 - 0x105F
   "P32" 0x1060 - 0x11FF
   *1. The end address of "P11" having an adjacent section is the same
       as that of "P21" having no adjacent section.

Example 3:
   <Section>
   (1) "P11, P12"
   (2) "P21, P12" (*2)
   (3) "P31, P32"
   *2. Looks as if "P12" is adjacent to "P21".
```

1.15.11   Command

(1) WATCH_EDIT command

If two or more variables with the same name are assigned different scope and are being watched in the [Watch] window, editing the value of the variable by using the WATCH_EDIT command only changes the value of the first variable with that name listed in the [Watch] window.

1.15.12 SuperH family and H8SX, H8S, H8 family simulators/debuggers

(1) Memory resource setting function

Note that the specifications of the memory resource setting function of the simulator/debugger differ from those of Renesas emulators.

The specifications for the simulator/debugger are as follows:

- When the memory attribute (Read, Write, or Read/Write) is the same after the memory resource is modified:
  The simulator/debugger assumes the memory resource size has been modified and changes the memory resource size to the specified value.
- When the memory attribute changes due to modification of the memory resource:
  The simulator/debugger assumes the memory resource attribute has been modified and changes the memory resource attribute of the specified range to the specified value.

(2) The number of breakpoints and the number of Stop-at points in the Run menu

Up to a total of 1,024 PC breakpoints and Stop-at points in the Run menu can be specified.

If 1,024 breakpoints are specified, no Stop-at points in the Run menu can be specified.

Specify 1,024 or fewer points for the total numbers of PC breakpoints and Stop-at points in the Run menu.

(3) Debug Settings dialog box

- Do not perform automatic target connection
  The target is connected when [Debug Settings] have been completed irrespective of the setting for the [Do not perform automatic target connection] check box on the [Options] tab.

(4) SH-4 and SH-4 with BSC simulator/debugger

- In SH-4 with BSC simulator/debugger, if the lower three bits of the source and destination addresses differ in a DMA transfer, the last data transferred will be invalid.

```
Example: SAR0=2000    DAR0=4004    DMATCR0=2    CHCR0=5491
         Memory contents of address H'2000: 0102030405060708
         Contents of address H'4004 after DMA transfer: 0106
```

- In SH-4 and SH-4 with BSC simulator/debugger, even if an instruction is modified for an address where decoding has been completed, the pipeline is not reset and executed.
- In SH-4 with BSC simulator/debugger, if memory is accessed by a data size that differs from the size specified by the break data, the program may not break when the break conditions are satisfied. To avoid this, specify the same data size for the memory access data size and break data size.
- In SH-4 and SH-4 with BSC simulator/debugger, the pipeline execution for double precision FDIV and FSQRT instructions in the SH-4 simulator/debugger is different from those in the user system. For the SH-4 simulator/debugger, one more cycle is displayed for the F3 stage pipeline.

(5) SH-3DSP simulator/debugger

- Exception code during DSP loop execution
  If an exception is generated during the DSP loop execution, the exception code set in the EXPEVT (exception event register) will differ from that described in the programming manual.

```
The exception code in the DSP Loop
General exception events                          Programming  Simulator
                                                  Manual
TLB miss exception/TLB invalid exception (Read)   H'070        H'040
TLB miss exception/TLB invalid exception (Write)  H'070        H'060
TLB protection exception (Read)                   H'0D0        H'0A0
TLB protection exception (Write)                  H'0D0        H'0C0
CPU address error (Read)                          H'070        H'0E0
CPU address error (Write)                         H'070        H'100
```

- X/Y memory access conflict
  If an instruction code and data are allocated to the XRAM memory (or XROM, or YROM, or YRAM), stalls will not be generated by a conflict even if this XRAM memory is accessed by the instruction code fetch and the MOVX or MOVY instruction in the same slot. Therefore, the number of cycles for these two types of access will differ.
- Pipeline execution from an address other than a multiple of four
  If a pipeline execution is performed from an address other than a multiple of four, the fetch stages will differ from those described in the programming manual.

```
Example: When pipeline execution is performed from an address other than
         a multiple of four
    Programming Manual    Simulator
    IF IF ID EX           IF ID EX
        IF ID EX              IF ID EX
            if ID EX              if ID EX
```

(6) Trace function (only for the SH3, SH3E, and SH-3DSP)
- FPU, MAC, or DSP register access information
  When an instruction is executed to write data in the FPU, MAC, or DSP register, the simulator/debugger does not display access information in the trace information.
- Pipeline display
  For instructions that programming manuals do not describe as writing back data to registers, and if data is written to registers as a result of the execution of such instructions, the simulator/debugger displays such an operation as a stage of memory access and write back in the trace information. The numbers of cycles are correctly displayed.

```
Example: Trace display by the simulator/debugger          Description in the
                                                          programming manual

                        IF DE EX MA SW                    IF DE EX MA SW
05  NOP                 06 05                             06 05
06  TRAPA #H'10         07 06 05                          -- 06
--  - - - - -           -- -- 06                          -- -- 06
--  - - - - -           -- -- 06 06                       -- -- 06
--  - - - - -           -- -- 06 06                       -- -- 06
--  - - - - -           -- -- 06 06 06 (06):SSR<-60000001 -- -- 06
--  - - - - -           08 -- -- 06 06 (06):SPC<-00001006 07 -- --
--  - - - - -           09 08 -- -- --                    08 07 --
```

(7) H8SX Simulator/debugger
  H8SX Simulator/debugger does not support Middle mode of H8SX CPU.

(8) Selection of the simulator target
  When you create a project, select the simulator target corresponding to the CPU and the operating mode.

(9) Internal RAM area in the SH2A-FPU simulator/debugger
  Though the memory map of the SH2A-FPU simulator/debugger shows addresses 0xFFF80000 to 0xFFFBFFFF as the internal RAM area, the range of 0xFFFA0000 to 0xFFFBFFFF are actually reserved.
  For this reason, note the followings when using the SH2A-FPU simulator/debugger:
- Only use the range of 0xFFF80000 to 0xFFF9FFFF as the internal RAM area.
- When a stack area is to be assigned to the internal RAM area, set the initial stack-pointer value so that it will be within the range of 0xFFF80000 to 0xFFF9FFFF.

(10) Saving workspaces for older versions of the High-performance Embedded Workshop

In High-performance Embedded Workshop V.4.04 and later versions, workspaces can be saved in the format compatible with older versions of the High-performance Embedded Workshop. Note, however, that the operation of the simulator/debugger may be extremely slow. If you intend to use this new feature, set the trace capacity as 32,768 or less.

1.16   Windows and dialog boxes

1.16.1   Scrolling

(1) Up-scrolling may not be available with an intelligent mouse. In this case, use a scrolling button on the window.

(2) If your PC has a touchpad with a scroll zone, scrolling the [Disassembly] window or the [Editor] window in the mixed or disassembly mode while pressing the mouse or touchpad button may cause an application error. Do not use the touchpad-scroll feature on the [Disassembly] or [Editor] window while pressing the mouse or touchpad button.

1.16.2   Saving the location of windows

The location of windows displayed may be changed by the following operations.
- Refresh the session after it has been saved
- Re-connect the target after it has been disconnected
- Switch the virtual desktop

1.17   Help

When the help is used in Windows® English version, the following message box may be displayed:
Click on the [Download] button to download "Japanese Text Display Support", or check the [Never download any of these components.] check box and click on the [Cancel] button.
Even if "Japanese Text Display Support" is not downloaded, the help is normally displayed.
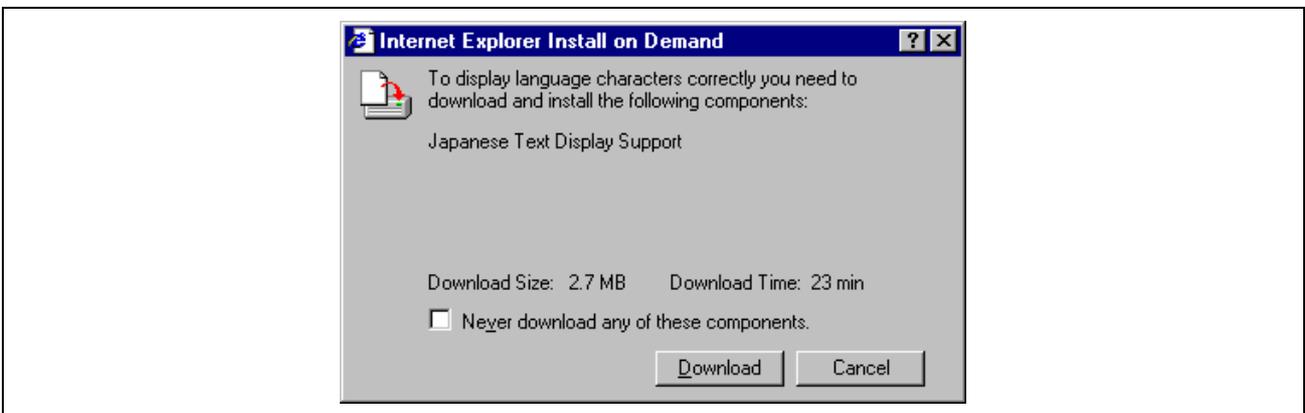
Figure 1.4   Internet Explorer Install on Demand message box

1.18   Macro-Recording Support facility

If you use the macro-recording facility to record the action of selecting [Debug -> Go] and [Debug -> Halt Program] into a macro, playing this macro will carry out [Debug -> Go] only. [Debug -> Halt Program] will not be carried out.

1.19   Commands

1.19.1   CACHE command

Do not specify the CACHE command unless it is defined in the user's manual of the emulator or debugger. The unit of an access to the cache memory is a fixed value such as 0x3FF. Thus the range of the cache memory being accessed may be larger than the access range specified by the user.

1.19.2   MEMORY_EDIT command

Even if characters are enclosed with quotation marks ('), the ASCII character strings cannot be input as data. When the character strings are input as data, enter the corresponding numerical values.

1.19.3   MEMORY_FIND command

DOUBLE (double-precision floating-point number; 8 bytes) is not specifiable for parameter <mode>.

1.19.4   Command batch file load timing (Debug Settings dialog box)

If you have specified a batch file for [Command batch file load timing] on the [Options] page, the restrictions listed below apply. In such cases, consider the use of other means via the GUI, e.g. menu options or toolbar buttons, instead of commands.

(1) When the [INITIALIZE] command is specified in the batch file that [At target connection] has been selected for [Command batch file load timing] on the [Options] tab, do not execute the [INITIALIZE] command in the [Command Line] window.
For initialization, select [Debug -> Initialize].

(2) Do not specify the following commands in the batch file that [At target connection] has been selected for [Command batch file load timing] on the [Options] tab.
(a) [OPEN_WORKSPACE] command
(b) [CHANGE_PROJECT] command
(c) [CHANGE_CONFIGURATION] command

(3) Do not execute the [FILE_LOAD] or [FILE_LOAD_ALL] command in the [Command Line] window that [After download of modules] has been selected for [Command batch file load timing] on the [Options] tab.

(4) Do not specify the following commands in the batch file that [Before download of modules] or [After download of modules] has been selected for [Command batch file load timing] on the [Options] tab.
(a) [OPEN_WORKSPACE] command
(b) [CHANGE_PROJECT] command
(c) [CHANGE_CONFIGURATION] command
(d) [GO] command
(e) [GO_RESET] command
(f) [GO_TILL] command
(g) [STEP] command
(h) [STEP_OUT] command
(i) [STEP_OVER] command

(5) Do not execute the [RESET] command in the [Command Line] window that [After reset] has been selected for [Command batch file load timing] on the [Options] tab.

(6) Do not specify the following commands in the batch file that [After reset] has been selected for [Command batch file load timing] on the [Options] tab.

(a) [OPEN_WORKSPACE] command

(b) [CHANGE_PROJECT] command

(c) [CHANGE_CONFIGURATION] command

(d) [RESET] command

1.19.5   Command line batch processing (Debug Settings dialog box)

If you have specified the relative path to a batch file for [Command line batch processing] on the [Options] tab, the file may not be correctly accessed.

Specify the absolute path to the batch file that cannot apply a placeholder.

1.19.6   Specifying a file in the command line

To specify a file in the command line, use a placeholder (excluding TCL).

If you wish to specify a directory not included in the placeholder, specify an absolute path.

After specifying the absolute path, this file will not be correctly found when it is in another host computer or environment where the path content is different. In such cases, specify the file again.

```
FILE_LOAD ELF/DWARF2 $(CONFIGDIR)\\demo.abs
```

1.19.7   Separating command parameters

You can use TCL commands while TCL is enabled. The use of the TCL command batch, in which parameters are separated by tabs, is also possible.

```
For   {set  i  0} {$i   <  2} {incr i} {
   puts [memory_display   300   10]
}
```

When TCL is disabled, TCL commands are not usable. Only white space can be used to separate parameters so using tabs instead will cause the following error.

```
>TCL
TCL Disabled
>memory_display   300   10
Error: Invalid command
```

1.19.8   Abbreviation of a command

- REMOVE_FILE

In High-performance Embedded Workshop V.3.01 or later version, the abbreviation has been changed from "RF" to "REM".

1.20   Enhanced compatibility between TCL commands and High-performance Embedded Workshop commands

Due to enhanced compatibility between TCL commands and High-performance Embedded Workshop commands*, the execution result of a High-performance Embedded Workshop command "memory_display 300 10" will not be output by the following specification.

```
for {set i 0} {$i < 2} {incr i} {
    memory_display 300 10
}
```

To make the execution result of the High-performance Embedded Workshop command be output, enclose the High-performance Embedded Workshop command with [] as a parameter of a TCL command "puts".

```
for {set i 0} {$i < 2} {incr i} {
    puts [memory_display 300 10]
}
```

*Note: Due to this enhanced compatibility, even the execution result of a High-performance Embedded Workshop command can be assigned to a variable as a parameter of a TCL command "set".
In the example below, since the execution result of "memory_display 300 10" is assigned to variable "md_300_10", you can view this result by specifying "md_300_10" with a TCL command "set".

```
set md_300_10 [memory_display 300 10]
```

1.21   Tcl/Tk command input

(1) Canceling the interactive mode
To return from the interactive mode to the command input mode while using the [TCL Toolkit], enter "/.". It is possible to know the current mode by entering "/.".

(2) Clearing the contents of a log file
The contents of the [Console] screen are logged to a log file while using [TCL Toolkit].
The log file is in plain text format and placed in the following location:

```
Example 1: Windows® XP operating system
    C:\Documents and Settings\<logon user name>\Local Settings\Temp\log.txt
Example 2: Windows Vista® or Windows® 7 operating system
    C:\users\<logon user name>\AppData\Local\Temp\log.txt
```

If you quit [TCL Toolkit], it automatically erases the contents of the log file.
If you want to clear the contents of the log file while using the [TCL Toolkit], execute the following commands.

```
set dir $env(TEMP)
set dataFile [open $dir/log.txt {RDWR TRUNC}]
close $dataFile
```

1.22   Commands in TCL Toolkit and Command Line

(1) "trace" command
(a) To execute the "trace" command for Tcl in [TCL Toolkit], the command name must be specified in lowercase letters.
(b) To execute the "TRACE" command for Hew in [TCL Toolkit], the command name must be specified in uppercase letters.
(c) To execute the "trace" command for Tcl in the [Command Line] window, the command name must be replaced with "tcl_trace".

(2) "clock" command
(a) To execute the "clock" command for Tcl in [TCL Toolkit], the command name must be specified in lowercase letters.
(b) To execute the "CLOCK" command for the emulator in [TCL Toolkit], the command name must be specified in uppercase letters.
(c) To execute the "clock" command for Tcl in the [Command Line] window, the command name must be replaced with "tcl_clock".

(3) "event" command
(a) To execute the "event" command for Tk in [TCL Toolkit], the command name must be specified in lowercase letters.
(b) To execute the "EVENT" command for the emulator in [TCL Toolkit], the command name must be specified in uppercase letters.

Note that some emulators do not support commands "CLOCK" and "EVENT".

1.23   Correction in the High-performance Embedded Workshop user's manual

(1) Document No: R20UT0372EJ0100
The line that begins with "Toolbars can be created, modified and removed" in section 1.2.3, Toolbars, (page 3) will be corrected as shown below.
This correction will also be applied to the online help for the High-performance Embedded Workshop.

Before:
Toolbars can be created, modified and removed via [Tools -> Customize] (see section 6.1, Customizing the toolbars, for further information).
Now:
Toolbars can be created, modified and removed via [Setup -> Customize] (see section 6.1, Customizing the toolbars, for further information).

(2) Document No: R20UT0372EJ0100
The line that begins with "A file dropped onto a user folder" in section 2.3.2, Drag and drop of files and folders, (page 47) will be corrected as shown below.
This correction will also be applied to the online help for the High-performance Embedded Workshop.

Before:
A file dropped onto a user folder is directly added to this folder. Even if a file with the same name is already in the folder, the new file can be added to the folder only when these two files have different paths.
Now:
A file dropped onto a user folder is directly added to this folder.

2.   Supplement on toolchain

2.1   File extensions

When the High-performance Embedded Workshop executes a tool of the toolchains in a build, the High-performance Embedded Workshop leaves a subcommand file in a configuration directory.

A subcommand file of the C/C++ Compiler or the Assembler uses a file extension shown in table 2.1 with the same filename as the input file.

The C/C++ Library Generator or the OptLinker uses a file extension shown in table 2.1 with a project name as a filename.

The subcommand file has an attribute of a hidden file. If you want to see a hidden file, change the properties of a directory window so that it can display hidden files.

Table 2.1   File Extensions of Subcommand Files of the Toolchains

File extension	File group
shg	SuperH RISC engine C/C++ Library Generator
shc	SuperH RISC engine C/C++ Compiler
sha	SuperH RISC engine Assembler
h8g	H8S, H8/300 C/C++ Library Generator
h8c	H8S, H8/300 C/C++ Compiler
h8a	H8S, H8/300 Assembler
hlk	OptLinker
m16cl	M16C Linker
m16ci	M16C Librarian
m16cc	M16C C/C++ Compiler
m16ca	M16C Assembler
m16ct	M16C mkmrtbl (M3T-MR30/4)
r8ct	R8C mr8ctbl (MR8C/4)
m32cl	M32C Linker
m32ci	M32C Librarian
m32cc	M32C C Compiler
m32ct	M32C mr308tbl (M3T-MR308/4)
m32rl	M32R C Compiler
m32ri	M32R Librarian
m32rm	M32R Load module converter
m32rc	M32R C Compiler
m32ra	M32R Assembler
74lk	740 Linker
74lb	740 Librarian
100l	R32C/100 Linker
100i	R32C/100 Librarian
100c	R32C/100 C Compiler
100t	R32C/100 mr100tbl (M3T-MR100/4)
rxg	RX C/C++ Library Generator
rxc	RX C/C++ Compiler
rxa	RX Assembler
r600t	RX mkritbl (RI600/4)

3. Upgrading the toolchain

When any of the following toolchains is used in a project, you can install the latest version to upgrade the toolchain currently in use.

- C/C++ Compiler Package for SuperH Family V.5.1 or later
- C/C++ Compiler Package for H8SX, H8S, H8 Family V.3.0A or later
- C Compiler Package for M16C Series and R8C Family V.5.20 Release 1 or later
- C Compiler Package for M32C Series V.5.20 Release 1 or later
- C Compiler Package for R32C Series V.1.01 Release 00 or later
- C/C++ Compiler Package for M32R Series V.4.20 Release 1 or later
- C Compiler Package for 740 Family V.1.01 Release 01 or later
- Assembler Package for 740 Family V.4.10 Release 02 or later

To upgrade the toolchain in use

Select [Tools -> Change Toolchain Version] to open the [Change Toolchain Version] dialog box and select a version number from [Toolchain version].
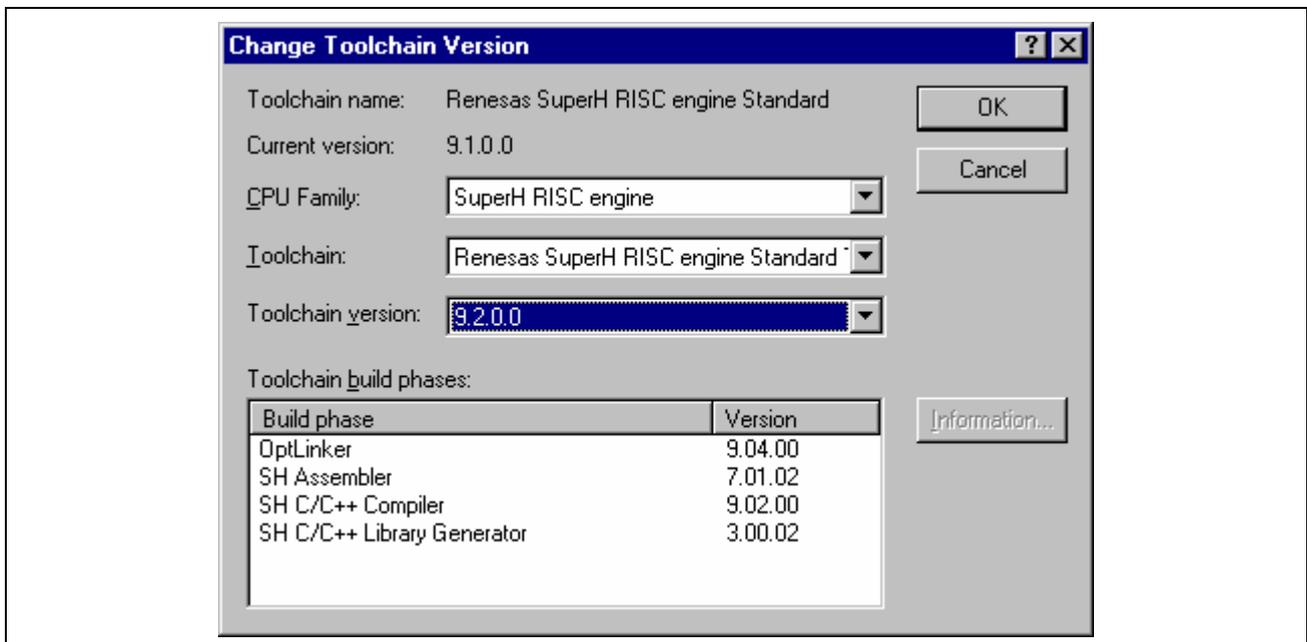
Figure 3.1 Change Toochain Version Dialog Box

(1) Upgrading from the SuperH Family C/C++ Compiler Package V.5.1x or H8SX, H8S, H8 Family C/C++ Compiler Package V.3.0x

(a) Setting the standard library configuration tool

A standard library configuration tool was added.

The High-performance Embedded Workshop checks the optional information on the IM Optlnker at upgrade.

When there is a specification to enter the standard library, an option that generates a library is set (Mode -> Build a library file (anytime)), and when there is no specification, an option that does not generate a library is set (Mode -> Do not add a library file).

(b) Setting the optimizing linker

The IM Optlnker, Librarian, and Stype Converter were integrated and became OptLinker.

According to the registered states of each tool before upgrade, the optional setting of OptLinker differs after upgrade. Table 3.1 shows the succession of the option at upgrade.

Table 3.1　　Succession of Option at Upgrade

No.	Before Upgrade				After Upgrade		
	IM OptLinker state	Librarian state	Stype Converter state	OptInker state	IM OptLinker optional information	Librarian Optional information	Stype Converter optional information
1	—	—	—	—	X	X	X
2	—	—	□	□	X	X	O
3	—	—	■	■	X	X	O
4	—	□	—	□	X	O	X
5	—	□	□	□	X	O	X
6	—	□	■	■	X	X	O
7	—	■	—	■	X	O	X
8	—	■	□	■	X	O	X
9	—	■	■	■	X	O	X
10	□	—	—	□	O	X	X
11	□	—	□	□	O	X	O
12	□	—	■	■	X	X	O
13	□	□	—	□	O	X	X
14	□	□	□	□	O	X	O
15	□	□	■	■	X	X	O
16	□	■	—	■	X	O	X
17	□	■	□	■	X	O	X
18	□	■	■	■	X	O	X
19	■	—	—	■	O	X	X
20	■	—	□	■	O	X	X
21	■	—	■	■	O	X	O
22	■	□	—	■	O	X	X
23	■	□	□	■	O	X	X
24	■	□	■	■	O	X	O
25	■	■	—	■	O	X	X
26	■	■	□	■	O	X	X
27	■	■	■	■	O	X	O

—: Unregistered, □: Registration not checked, ■: Registration checked, O: Succeeded, X: Not succeeded

(2) Upgrading from the SuperH Family C/C++ Compiler Package

(a) Automatic Build for optimizing MAP

In High-performance Embedded Workshop V.2.00 (SH Ver.7.0B/Ver.7.0.01/Ver.7.0.02), to realize optimization by using the external symbol allocation information that has been output by the optimizing linkage editor in the C compiler, a custom phase has been provided for rebuild.

In High-performance Embedded Workshop V.2.01 or later version, automatic rebuild is enabled when MAP optimization is executed.

To disable the custom phase for rebuild

Since the custom phase for rebuild is not required, open the Build Phases dialog box and remove a check for the "Map optimize" phase.
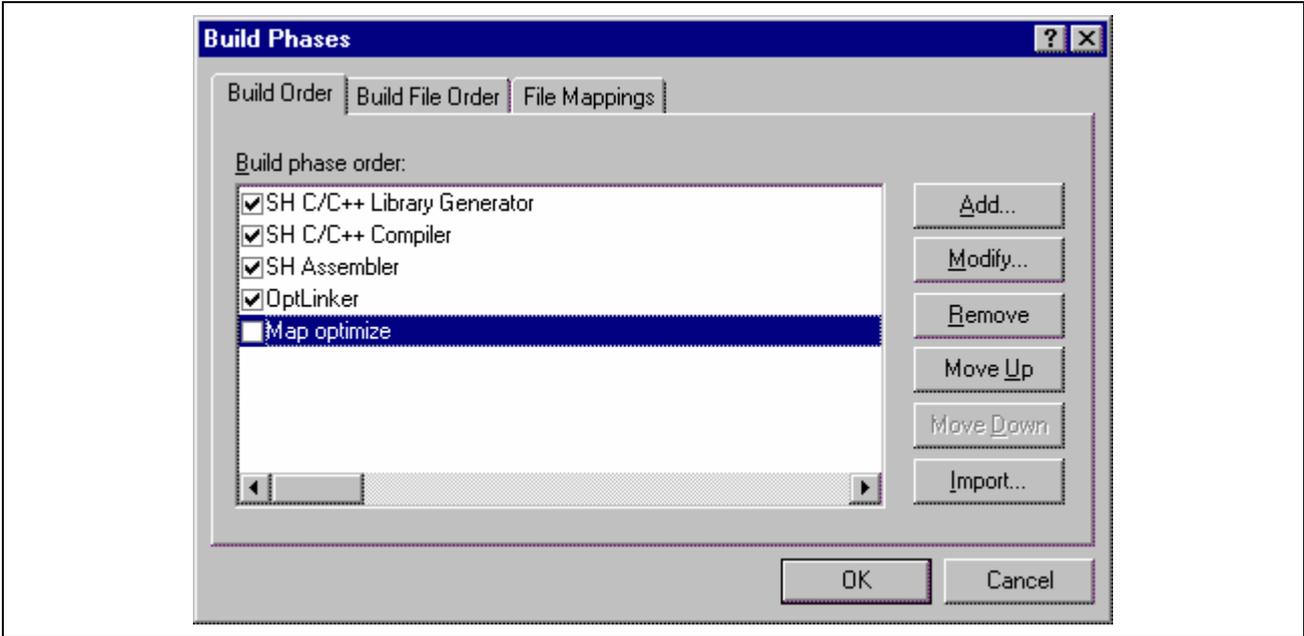
Figure 3.2　Build Phases Dialog Box

(3) Upgrading from the C Compiler Package for M16C Series and R8C Family V.5.x

(a) Changing the object format

The object format for M16C Series and R8C Family C/C++ Compiler Package V.6.00 Release 00 and later versions is ELF/DWARF2.

Select [Debug -> Debug Settings] to open the [Debug Settings] dialog box and select ELF/DWARF2 for [Debug format]. After that, click on the [Modify] button to open the [Download Module] dialog box. Select ELF/DWARF2 for [File format] and also change the file extension from .x30 to .abs.
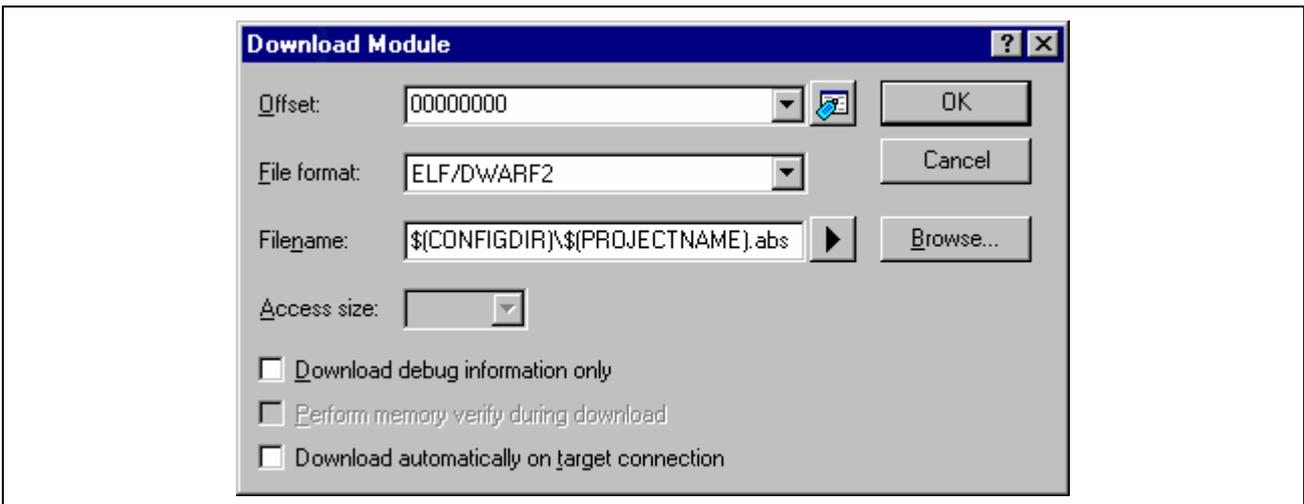
Figure 3.3　Download Module Dialog Box

4.   Converting to an RX project

When any of the following toolchains is used in a project, you can convert the project to an RX project by installing C/C++ Compiler Package for RX Family V.1.01 Release 00 or later.

- C/C++ Compiler Package for SuperH Family V.9.04 Release 00
- C/C++ Compiler Package for H8SX, H8S, H8 Family V.6.02 Release 00 to V.6.02 Release 02
- C/C++ Compiler Package for H8SX, H8S, H8 Family V.7.00 Release 00
- C Compiler Package for M16C Series and R8C Family V.5.45 Release 00 and V.5.45 Release 01
- C/C++ Compiler Package for M16C Series and R8C Family V.6.00 Release 00

This conversion only applies to the settings given below and involves updating of the current project generator to the RX project generator.

- Toolchain build-phase options set via [Build -> RX Standard Toolchain]
- System phases of build phases (also called toolchain build phases) set via [Build -> Build Phase]

All other settings and files registered with the project (e.g. source files) or output in the configuration folder or make folder are not converted.

Making a backup of the current environment
It is not possible to restore projects after they have been converted.
Thus, before starting conversion, copy the workspace folder where the project has been registered (which can be found by using the explorer) and save it with another name.

Using a debugger
If you wish to use a debugger after converting the project, add a new session in which the RX project generator is selected.
For how to add a new session, select [Help -> Help Topic] from the main menu to open the High-performance Embedded Workshop online help and see the topic of "Adding a session" (Debugging Facility -> Preparations for debugging -> Debugger sessions).

Updating the contents of output files
After the project has been converted, add a session as required and take the following operations to update the contents of the old output files.

- [Build -> Update All Dependencies]
- [Build -> Generate Makefile] *****
- [Build -> Build All]
Note: *****. This is necessary only when a makefile is being used.

To convert the project to an RX project

Select [Tools -> Change Toolchain Version] to open the [Change Toolchain Version] dialog box and select "RX" from [CPU Family].
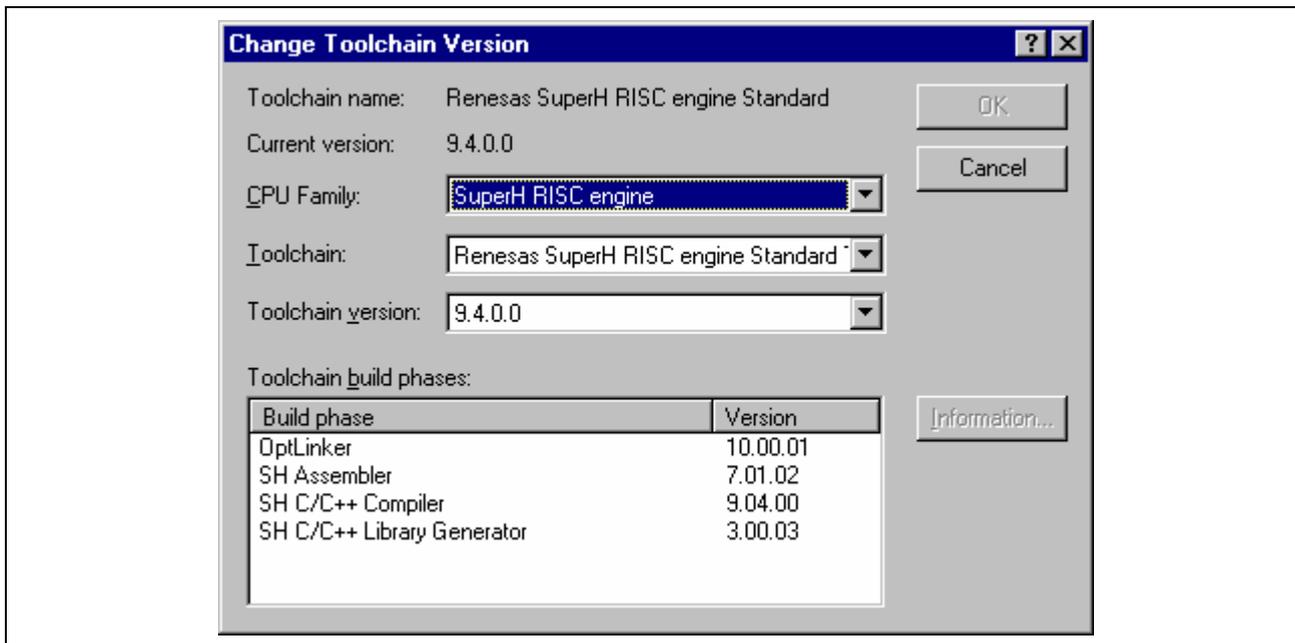
Figure 3.4　Change Toochain Version Dialog Box

The toolchain build-phase options given in the table below are inherited even after the project has been converted. The [Skip dependency scan of nonexistent file] and [Support dependency scan of preprocessor statement] settings (under [Toolchain Option]) are also inherited.

All other toolchain build-phase options and system phases of build phases have the default values for the RX.

(a) C/C++ Compiler Package for SuperH Family V.9.04 Release 00 -> RX

Phase	Options before Conversion		Options after Conversion	
Compiler	C/C++ <Source> [Show entries for :] [Include file directories]	-include	C/C++ <Source> [Show entries for :] [Include file directories]	-include
	C/C++ <Source> [Show entries for :] [Defines]	-define	C/C++ <Source> [Show entries for :] [Defines]	-define
	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude
	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message
	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path

Phase	Options before Conversion		Options after Conversion	
	There is no corresponding GUI item. The High-performance Embedded Workshop recognizes the language by the extension of input files.	-lang	C/C++ <Source> [Show entries for:] [Source file] [Language :]	-lang
	C/C++ <Object> [Output file type :]	-code=asmcode	C/C++ <Object> [Output file type :]	-output=src
	C/C++ <Object> [Output file type :]	-preprocessor	C/C++ <Object> [Output file type :]	-output=prep
	C/C++ <Object> [Output file type :]	-noline	C/C++ <Object> [Output file type :]	-noline
	C/C++ <Object> [Output directory]		C/C++ <Object> [Output directory]	
	C/C++ <Object> [Generate debug information]	-debug	C/C++ <Object> [Generate debug information]	-debug
	C/C++ <Object> [Details] [Section :]	-section	C/C++ <Object> [Details] [Section :]	-section
	C/C++ <Optimize> [Optimize level :]	-optimize	C/C++ <Optimize> [Optimize level :]	-optimize
	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case
	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline
	C/C++ <Optimize> [Details] [Inline] [Automatic inline expansion :]	-inline	C/C++ <Optimize> [Details] [Inline] [Automatic inline expansion :]	-inline
	C/C++ <Optimize> [Inter-module optimization]	-goptimize	C/C++ <Optimize> [Inter-module optimization]	-goptimize
	CPU [Details] [Use try, throw and catch of C++]	-exception	CPU [Details] [Use try, throw and catch of C++]	-exception
	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on
	CPU [Details] [Pack struct, union and class]	-pack=1	CPU [Details] [Pack struct, union and class]	-pack
	CPU [Endian :]	-endian	CPU [Endian :]	-endian
Assembler	Assembly <Source> [Show entries for:] [Include file directories]	-include	Assembly <Source> [Show entries for:] [Include file directories]	-include
	Assembly <Source> [Show entries for:] [Defines]	-define	Assembly <Source> [Show entries for:] [Defines]	-define
	Assembly <Object> [Generate debug information]	-debug	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Output directory]		Assembly <Object> [Output directory]	

RENESAS

Phase	Options before Conversion		Options after Conversion	
OptLinker	Link/Library <Input> [Show entries for :] [Library files]	-library	Link/Library <Input> [Show entries for :] [Library files]	-library
	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input
	Link/Library <Input> [Show entries for :] [Binary files]	-binary	Link/Library <Input> [Show entries for :] [Binary files]	-binary
	Link/Library <Input> [Show entries for :] [Defines:]	-define	Link/Library <Input> [Show entries for :] [Defines:]	-define
	Link/Library <Input> [Use entry point :]	-entry	Link/Library <Input> [Use entry point :]	-entry
	Link/Library <Input> [Prelinker control :]	-noprelink	Link/Library <Input> [Prelinker control :]	-noprelink
	Link/Library <Output> [Type of output file :]	-form	Link/Library <Output> [Type of output file :]	-form
	Link/Library <Output> [Data record header :]	-record	Link/Library <Output> [Data record header :]	-record
	Link/Library <Output> [Length of data record :]	-byte_count	Link/Library <Output> [Length of data record :]	-byte_count
	Link/Library <Output> [Debug information :]	-debug	Link/Library <Output> [Debug information :]	-debug
	Link/Library <Output> [Debug information :]	-sdebug	Link/Library <Output> [Debug information :]	-sdebug
	Link/Library <Output> [Show entries for :] [Output file path]	-output	Link/Library <Output> [Show entries for :] [Output file path]	-output
	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom
	Link/Library <Output> [Show entries for :] [Divided output files:]	-output	Link/Library <Output> [Show entries for :] [Divided output files:]	-output
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage
	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused
	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space

RENESAS

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc
	Link/Library <List> [Generate list file]	-list	Link/Library <List> [Generate list file]	-list
	Link/Library <List> [Contents :]	-show	Link/Library <List> [Contents :]	-show
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid
	Link/Library <Section> [Show entries for :] [Section]	-start	Link/Library <Section> [Show entries for :] [Section]	-start
	Link/Library <Section> [Show entries for :] [Symbol file]	-fsymbol	Link/Library <Section> [Show entries for :] [Symbol file]	-fsymbol
	Link/Library <Verify> [CPU information check :]	-cpu	Link/Library <Verify> [CPU information check :]	-cpu
	Link/Library <Verify> [CPU information check :]	-cpu=stride	Link/Library <Verify> [CPU information check :]	-cpu=stride
	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section
	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack
	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9
	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory
	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress
	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand

(b) C/C++ Compiler Package for H8SX, H8S, H8 Family V.6.02 Release 00 to V.6.02 Release 02 -> RX

Phase	Options before Conversion		Options after Conversion	
Compiler	C/C++ <Source> [Show entries for :] [Include file directories]	-include	C/C++ <Source> [Show entries for :] [Include file directories]	-include
	C/C++ <Source> [Show entries for :] [Defines]	-define	C/C++ <Source> [Show entries for :] [Defines]	-define
	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude
	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message
	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path
	C/C++ <Object> [Output file type :]	-code=asmcode	C/C++ <Object> [Output file type :]	-output=src
	C/C++ <Object> [Output file type :]	-preprocessor	C/C++ <Object> [Output file type :]	-output=prep
	C/C++ <Object> [Output directory]		C/C++ <Object> [Output directory]	
	C/C++ <Object> [Generate debug information]	-debug	C/C++ <Object> [Generate debug information]	-debug
	C/C++ <Object> [Output character code:]	-outcode	C/C++ <Object> [Output character code:]	-outcode
	C/C++ <Object> [Details] [Section :]	-section	C/C++ <Object> [Details] [Section :]	-section
	C/C++ <Optimize> [Optimize level :]	-optimize	C/C++ <Optimize> [Optimize level :]	-optimize
	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case

Phase	Options before Conversion		Options after Conversion	
	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline
	C/C++ <Optimize> [Inter-module optimization]	-goptimize	C/C++ <Optimize> [Inter-module optimization]	-goptimize
	CPU [Details] [Use try, throw and catch of C++]	-exception	CPU [Details] [Use try, throw and catch of C++]	-exception
	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on
	CPU [Details] [Pack struct, union and class]	-pack=1	CPU [Details] [Pack struct, union and class]	-pack
Assembler	Assembly <Source> [Show entries for:] [Include file directories]	-include	Assembly <Source> [Show entries for:] [Include file directories]	-include
	Assembly <Source> [Show entries for:] [Defines]	-define	Assembly <Source> [Show entries for:] [Defines]	-define
	Assembly <Object> [Generate debug information]	-debug	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Inter-module optimization]	-goptimize	Assembly <Object> [Inter-module optimization]	-goptimize
	Assembly <Object> [Output directory]		Assembly <Object> [Output directory]	
OptLinker	Link/Library <Input> [Show entries for :] [Library files]	-library	Link/Library <Input> [Show entries for :] [Library files]	-library
	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input
	Link/Library <Input> [Show entries for :] [Binary files]	-binary	Link/Library <Input> [Show entries for :] [Binary files]	-binary
	Link/Library <Input> [Show entries for :] [Defines:]	-define	Link/Library <Input> [Show entries for :] [Defines:]	-define
	Link/Library <Input> [Use entry point :]	-entry	Link/Library <Input> [Use entry point :]	-entry
	Link/Library <Input> [Prelinker control :]	-noprelink	Link/Library <Input> [Prelinker control :]	-noprelink
	Link/Library <Output> [Type of output file :]	-form	Link/Library <Output> [Type of output file :]	-form
	Link/Library <Output> [Data record header :]	-record	Link/Library <Output> [Data record header :]	-record
	Link/Library <Output> [Length of data record :]	-byte_count	Link/Library <Output> [Length of data record :]	-byte_count
	Link/Library <Output> [Debug information :]	-debug	Link/Library <Output> [Debug information :]	-debug
	Link/Library <Output> [Debug information :]	-sdebug	Link/Library <Output> [Debug information :]	-sdebug

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Output> [Show entries for :] [Output file path]	-output	Link/Library <Output> [Show entries for :] [Output file path]	-output
	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom
	Link/Library <Output> [Show entries for :] [Divided output files:]	-output	Link/Library <Output> [Show entries for :] [Divided output files:]	-output
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage
	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused
	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space
	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc
	Link/Library <List> [Generate list file]	-list	Link/Library <List> [Generate list file]	-list
	Link/Library <List> [Contents :]	-show	Link/Library <List> [Contents :]	-show
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid
	Link/Library <Section> [Show entries for :] [Section]	-start	Link/Library <Section> [Show entries for :] [Section]	-start
	Link/Library <Verify> [CPU information check :]	-cpu	Link/Library <Verify> [CPU information check :]	-cpu
	Link/Library <Verify> [CPU information check :]	-cpu=stride	Link/Library <Verify> [CPU information check :]	-cpu=stride
	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section
	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack
	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size
	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9
	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory
	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress
	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand

(c) C/C++ Compiler Package for H8SX, H8S, H8 Family V.7.00 Release 00 -> RX

Phase	Options before Conversion		Options after Conversion	
Compiler	C/C++ <Source> [Show entries for :] [Include file directories]	-include	C/C++ <Source> [Show entries for :] [Include file directories]	-include
	C/C++ <Source> [Show entries for :] [Defines]	-define	C/C++ <Source> [Show entries for :] [Defines]	-define
	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude

Phase	Options before Conversion		Options after Conversion	
	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message	C/C++ <Source> [Show entries for :] [Messages] [Repressed information level messages]	-message
	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path	C/C++ <Source> [Show entries for :] [File inline path]	-file_inline_path
	C/C++ <Source> [Select language of C :]	-lang	C/C++ <Source> [Show entries for:] [Source file] [Language :]	-lang
	C/C++ <Object> [Output file type :]	-code=asmcode	C/C++ <Object> [Output file type :]	-output=src
	C/C++ <Object> [Output file type :]	-preprocessor	C/C++ <Object> [Output file type :]	-output=prep
	C/C++ <Object> [Output directory]		C/C++ <Object> [Output directory]	
	C/C++ <Object> [Generate debug information]	-debug	C/C++ <Object> [Generate debug information]	-debug
	C/C++ <Object> [Output character code:]	-outcode	C/C++ <Object> [Output character code:]	-outcode
	C/C++ <Object> [Details] [Section :]	-section	C/C++ <Object> [Details] [Section :]	-section
	C/C++ <Optimize> [Optimize level :]	-optimize	C/C++ <Optimize> [Optimize level :]	-optimize
	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case	C/C++ <Optimize> [Details] [Miscellaneous] [Switch statement :]	-case
	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline	C/C++ <Optimize> [Details] [Inline] [Inline file path]	-file_inline
	C/C++ <Optimize> [Inter-module optimization]	-goptimize	C/C++ <Optimize> [Inter-module optimization]	-goptimize
	CPU [Details] [Use try, throw and catch of C++]	-exception	CPU [Details] [Use try, throw and catch of C++]	-exception
	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on
	CPU [Details] [Pack struct, union and class]	-pack=1	CPU [Details] [Pack struct, union and class]	-pack
Assembler	Assembly <Source> [Show entries for:] [Include file directories]	-include	Assembly <Source> [Show entries for:] [Include file directories]	-include
	Assembly <Source> [Show entries for:] [Defines]	-define	Assembly <Source> [Show entries for:] [Defines]	-define
	Assembly <Object> [Generate debug information]	-debug	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Inter-module optimization]	-goptimize	Assembly <Object> [Inter-module optimization]	-goptimize

Phase	Options before Conversion		Options after Conversion	
	Assembly <Object> [Output directory]		Assembly <Object> [Output directory]	
OptLinker	Link/Library <Input> [Show entries for :] [Library files]	-library	Link/Library <Input> [Show entries for :] [Library files]	-library
	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input
	Link/Library <Input> [Show entries for :] [Binary files]	-binary	Link/Library <Input> [Show entries for :] [Binary files]	-binary
	Link/Library <Input> [Show entries for :] [Defines:]	-define	Link/Library <Input> [Show entries for :] [Defines:]	-define
	Link/Library <Input> [Use entry point :]	-entry	Link/Library <Input> [Use entry point :]	-entry
	Link/Library <Input> [Prelinker control :]	-noprelink	Link/Library <Input> [Prelinker control :]	-noprelink
	Link/Library <Output> [Type of output file :]	-form	Link/Library <Output> [Type of output file :]	-form
	Link/Library <Output> [Data record header :]	-record	Link/Library <Output> [Data record header :]	-record
	Link/Library <Output> [Length of data record :]	-byte_count	Link/Library <Output> [Length of data record :]	-byte_count
	Link/Library <Output> [Debug information :]	-debug	Link/Library <Output> [Debug information :]	-debug
	Link/Library <Output> [Debug information :]	-sdebug	Link/Library <Output> [Debug information :]	-sdebug
	Link/Library <Output> [Show entries for :] [Output file path]	-output	Link/Library <Output> [Show entries for :] [Output file path]	-output
	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom
	Link/Library <Output> [Show entries for :] [Divided output files:]	-output	Link/Library <Output> [Show entries for :] [Divided output files:]	-output
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage
	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space
	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc
	Link/Library <List> [Generate list file]	-list	Link/Library <List> [Generate list file]	-list
	Link/Library <List> [Contents :]	-show	Link/Library <List> [Contents :]	-show
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of same code]	-samecode_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid
	Link/Library <Section> [Show entries for :] [Section]	-start	Link/Library <Section> [Show entries for :] [Section]	-start
	Link/Library <Verify> [CPU information check :]	-cpu	Link/Library <Verify> [CPU information check :]	-cpu
	Link/Library <Verify> [CPU information check :]	-cpu=stride	Link/Library <Verify> [CPU information check :]	-cpu=stride
	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section
	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack

RENESAS

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size
	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9
	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory
	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress
	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand

(d) C Compiler Package for M16C Series and R8C Family V.5.45 Release 00 and V.5.45 Release 01 -> RX

Phase	Options before Conversion		Options after Conversion	
Compiler	C <Source> [Show entries for :] [Include file directories]	-I	C/C++ <Source> [Show entries for :] [Include file directories]	-include
	C <Source> [Show entries for :] [Defines]	-D	C/C++ <Source> [Show entries for :] [Defines]	-define
	C <Object> [Output file type :]	-S	C/C++ <Object> [Output file type :]	-output=src
	C <Object> [Output file type :]	-P	C/C++ <Object> [Output file type :]	-output=prep
	C <Object> [Output file type :]	-E	C/C++ <Object> [Output file type :]	-output=prep
	C <Object> [Specifies the directory to output the file(s) to :]	-dir	C/C++ <Object> [Output directory]	
	C <Object> [Debug options :] [Outputs debugging information. Therefore you can perform C language-level debugging.]	-g	C/C++ <Object> [Generate debug information]	-debug
Assembler	Assembly <Source> [Show entries for:] [Include file directories]	-I	Assembly <Source> [Show entries for:] [Include file directories]	-include
	Assembly <Source> [Show entries for:] [Defines]	-D	Assembly <Source> [Show entries for:] [Defines]	-define
	Assembly <Object> [Output file directory :]	-O	Assembly <Object> [Output directory]	-output

Phase	Options before Conversion		Options after Conversion	
	Assembly <Object> [Specifies file local symbol information be output.]	-S	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Specifies system label and local symbol information output.]	-SM	Assembly <Object> [Generate debug information]	-debug
Optlnk	Link <Input> [Show entries for :] [Library files]	-L	Link/Library <Input> [Show entries for :] [Library files]	-library
	Link <Input> [Show entries for :] [Relocatable files]		Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input
	Link <Input> [Specifies start address of absolute module :]	-E	Link/Library <Input> [Use entry point :]	-entry
	Link <Output> [Outputs source debug information to absolute module file.]	-G	Link/Library <Output> [Debug information :]	-debug
			Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack
	Link <Output> [Specifies absolute module file name :]	-O	Link/Library <Output> [Output file path]	-output
	Link <Subcommand file> [Use external subcommand file.]	@	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand
	Lmc <Output> [Format :]	-H	Link/Library <Output> [Type of output file :]	-form
	Librarian <Operation> [Operation :]	-C	Link/Library <Output> [Type of output file :]	-form
RTOS	RTOS <Mkritbl> [Search directory for MRC files]		RTOS <Mkritbl> [Search directory for MRC files]	

(e) C/C++ Compiler Package for M16C Series and R8C Family V.6.00 Release 00 -> RX

Phase	Options before Conversion		Options after Conversion	
Compiler	C/C++ <Source> [Show entries for :] [Source file] [Specifies the language of the source file]	-lang	C/C++ <Source> [Show entries for:] [Source file] [Language :]	-lang
	C/C++ <Source> [Show entries for :] [Include file directories]	-I	C/C++ <Source> [Show entries for :] [Include file directories]	-include
	C/C++ <Source> [Show entries for :] [Defines]	-D	C/C++ <Source> [Show entries for :] [Defines]	-define

RENESAS

Phase	Options before Conversion		Options after Conversion	
	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude	C/C++ <Source> [Show entries for :] [Preinclude files]	-preinclude
	C/C++ <Object> [Output file type :]	-S	C/C++ <Object> [Output file type :]	-output=src
	C/C++ <Object> [Output file type :]	-P	C/C++ <Object> [Output file type :]	-output=prep
	C/C++ <Object> [Specifies the directory to output the file(s) to :]	-dir	C/C++ <Object> [Output directory]	
	C/C++ <Object> [Debug options :] [Outputs debugging information. Therefore you can perform C language-level debugging.]	-g	C/C++ <Object> [Generate debug information]	-debug
	C/C++ <Optimize> [Inter-module optimization]	-goptimize	C/C++ <Optimize> [Inter-module optimization]	-goptimize
	CPU [Use try, throw and catch of C++]	-exception	CPU [Details] [Use try, throw and catch of C++]	-exception
	CPU [Use dynamic_cast and typeid of C++]	-rtti=on	CPU [Details] [Use dynamic_cast and typeid of C++]	-rtti=on
Assembler	Assembly <Source> [Show entries for:] [Include file directories]	-I	Assembly <Source> [Show entries for:] [Include file directories]	-include
	Assembly <Source> [Show entries for:] [Defines]	-D	Assembly <Source> [Show entries for:] [Defines]	-define
	Assembly <Object> [Specifies file local symbol information be output.]	-S	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Specifies system label and local symbol information output.]	-SM	Assembly <Object> [Generate debug information]	-debug
	Assembly <Object> [Inter-module optimization]	-goptimize	Assembly <Object> [Inter-module optimization]	-goptimize
	Assembly <Object> [Output file directory :]	-O	Assembly <Object> [Output directory]	
OptLinker	Link/Library <Input> [Show entries for :] [Library files]	-library	Link/Library <Input> [Show entries for :] [Library files]	-library
	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input	Link/Library <Input> [Show entries for :] [Relocatable files and object files]	-input
	Link/Library <Input> [Show entries for :] [Binary files]	-binary	Link/Library <Input> [Show entries for :] [Binary files]	-binary

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Input> [Show entries for :] [Defines:]	-define	Link/Library <Input> [Show entries for :] [Defines:]	-define
	Link/Library <Input> [Use entry point :]	-entry	Link/Library <Input> [Use entry point :]	-entry
	Link/Library <Input> [Prelinker control :]	-noprelink	Link/Library <Input> [Prelinker control :]	-noprelink
	Link/Library <Output> [Type of output file :]	-form	Link/Library <Output> [Type of output file :]	-form
	Link/Library <Output> [Data record header :]	-record	Link/Library <Output> [Data record header :]	-record
	Link/Library <Output> [Length of data record :]	-byte_count	Link/Library <Output> [Length of data record :]	-byte_count
	Link/Library <Output> [Debug information :]	-debug	Link/Library <Output> [Debug information :]	-debug
	Link/Library <Output> [Debug information :]	-sdebug	Link/Library <Output> [Debug information :]	-sdebug
	Link/Library <Output> [Show entries for :] [Output file path]	-output	Link/Library <Output> [Show entries for :] [Output file path]	-output
	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom	Link/Library <Output> [Show entries for :] [ROM to RAM mapped sections:]	-rom
	Link/Library <Output> [Show entries for :] [Divided output files:]	-output	Link/Library <Output> [Show entries for :] [Divided output files:]	-output
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-message
	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage	Link/Library <Output> [Show entries for :] [Messages] [Repressed information level messages:]	-nomessage
	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused	Link/Library <Output> [Show entries for :] [Notify unused symbol:]	-msg_unused
	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space	Link/Library <Output> [Specify value filled in unused area] [Output padding data]	-space
	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc	Link/Library <Output> [Show entries for :] [Generate CRC code]	-crc
	Link/Library <Output> [Show entries for :] [Vector] [Specific vector :]	-vectn	Link/Library <Output> [Show entries for :] [Vector] [Specific vector :]	-vectn
	Link/Library <Output> [Show entries for :] [Vector] [Empty vector :]	-vect	Link/Library <Output> [Show entries for :] [Vector] [Empty vector :]	-vect

Phase	Options before Conversion		Options after Conversion	
	Link/Library <List> [Generate list file]	-list	Link/Library <List> [Generate list file]	-list
	Link/Library <List> [Contents :]	-show	Link/Library <List> [Contents :]	-show
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-optimize
	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize	Link/Library <Optimize> [Show entries for :] [Optimize items] [Optimize :]	-nooptimize
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Elimination of dead code]	-symbol_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Section in]	-section_forbid
	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid	Link/Library <Optimize> [Show entries for :] [Forbid item] [Memory allocation in]	-absolute_forbid
	Link/Library <Section> [Show entries for :] [Section]	-start	Link/Library <Section> [Show entries for :] [Section]	-start
	Link/Library <Verify> [CPU information check :]	-cpu	Link/Library <Verify> [CPU information check :]	-cpu
	Link/Library <Verify> [CPU information check :]	-cpu=stride	Link/Library <Verify> [CPU information check :]	-cpu=stride
	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section	Link/Library <Verify> [Not divide the specified section :]	-contiguous_section
	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack	Link/Library <Other> [Miscellaneous options :] [Stack information output]	-stack
	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size	Link/Library <Other> [Miscellaneous options :] [Displays total section size]	-total_size
	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9	Link/Library <Other> [Miscellaneous options :] [Always output S9 record at the end]	-s9
	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory	Link/Library <Other> [Miscellaneous options :] [Low memory use during linkage]	-memory

Phase	Options before Conversion		Options after Conversion	
	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress	Link/Library <Other> [Miscellaneous options :] [Compress debug information]	-compress
	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand	Link/Library <Subcommand file> [Use external subcommand file]	-subcommand
RTOS	RTOS <Mkritbl> [Search directory for MRC files]		RTOS <Mkritbl> [Search directory for MRC files]	

5. Compatibility with Microsoft® Windows® Vista and Windows® 7

5.1 With running High-performance Embedded Workshop V.4.05.00 or later on Windows Vista® or Windows® 7 by user rights

Please take note of the following problem in using High-performance Embedded Workshop V.4.05.00 or later: With running High-performance Embedded Workshop V.4.05.00 or later on Windows Vista® or Windows® 7 by user rights

5.1.1 Description

Even if your High-performance Embedded Workshop and all the software-development tools managed by it are compatible with Windows Vista® or Windows® 7, the OS may requests you to be logged on as a member of the Administrators group; that is, you cannot run them on the OS by user rights.

This problem arises when you are logged on as a member of the Administrators group and make the settings for running High-performance Embedded Workshop V.4.04.01 or earlier on Windows Vista® or Windows® 7 by using the Compatibility tab, and afterward you update the IDE to the one compatible with Windows Vista® (High-performance Embedded Workshop V.4.05.00 or later) or Windows® 7 (High-performance Embedded Workshop V.4.08.00 or later).

NOTICE:
High-performance Embedded Workshop can run on Windows Vista® or Windows® 7 by user rights when both of the following conditions are satisfied:
(1) The IDE is compatible with Windows Vista® (i.e. V.4.05.00 or later) or Windows® 7 (i.e. V.4.08.00 or later).
(2) All the software-development tools managed by the IDE are also compatible with Windows Vista® or Windows® 7.

Otherwise, you must be logged on as a member of the Administrators group to run the IDE on Windows Vista® or Windows® 7 as well as installing any software-development tool.

5.1.2 Workarounds

To avoid the problem, use either of the following ways:
Re-setting items in the Compatibility tab of the program's Properties dialog box
a. Right-click the shortcut of High-performance Embedded Workshop to open the Properties dialog box.
b. In the Compatibility tab, clear the following two check boxes:
- Run This Program in Compatibility Mode
- Run the Program as an Administration

RENESAS

Renesas Electronics Corporation

http://www.renesas.com

SALES OFFICES

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141