To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

For M16C/60,30,Tiny,20,10,R8C/Tiny series C compiler Package

V.5.44 Release 00

Guidebook

(Rev.1.00)

Renesas Solutions Corporation

Apr. 1, 2008

Abstract

This document provides a guide to the introduction of M16C/60,30,Tiny,20,10,R8C/Tiny series C compiler Package V.5.44 Release 00. When you install this software package or create a project or want to know about the compiler, please refer to this guidebook.

A. Installation guide of C compiler package 3
A.1. Before installing C Compiler Package 3
A.2. Required System Configuration 3
A.3. Installation Procedure 3
A.3.1. When installing for the first time 4
A.3.2. When installing AutoUpdate 13
A.3.3. When checking of the existing environment 16
A.3.4. When installing two or more High-performance Embedded Workshop in a single PC. 17
A.4. Uninstalling programs 17
A.5. Startup or termination of program 17
A.5.1. Startup and termination of the High-performance Embedded Workshop 17
A.5.2. Start of Manual Navigator 18
A.5.3. Displaying the online manuals and attached documents 18
A.6. Setting when compiler is used on DOS prompt and command prompt 18
A.6.1. Environment Variables and Path 18
A.6.2. Batch file 18
B. Guide Book for V5.44 Release 00 19
B.1. Points to be noted when you upgrade from old versions of the compiler to V.5.44 Release 00 and use it in combination of Renesas real-time OS 19
B.1.1. Modify startup file 19
B.1.2. Change size of size_t, ptrdiff_t 21
B.1.3. Interrupt vector 24
B.2. Points to be maked new project when you use V.5.44 Release 00 24
B.2.1. Select CPU 24
B.2.2. To create a new workspace with a microcomputer that is not listed in CPU Group 24
B.2.3. When using an assembler startup 28
C. A Guide to Porting Projects Created with TM to High-performance Embedded Workshop V.4 29
C.1. Summary 29
C.2. Porting Procedure 29
C.3. Usage Notices 31
C.3.1. TM-to-High-performance Embedded Workshop Portable and Non-Portable Information 31
C.3.2. Cross Tools 31
C.3.3. High-performance Embedded Workshop Versions 31

C.3.4. Generated Project Workspace 32
C.3.5. Load Module Converter 32
C.3.6. Other Tools 33
C.3.7. Linkage order 37
C.3.8. Placing the Start Up program at the top of Linkage Order 37

A. Installation guide of C compiler package

A.1. Before installing C Compiler Package

Please confirm as follows before installing C Compiler Package in your computer.

- Please carefully read the "License Agreement" and “Release Note” included with your product before using C Compiler Package. If you've installed this product in your computer, it is assumed that you've agreed to the provisions stipulated in the License Agreement.
- In order that C Compiler Package operates comfortably, it requires at least 32Mbytes of memory and a hard disk having 20Mbytes or more of space.
- You need to input a license ID in the middle of installation. Before you start installing C Compiler Package, check your license ID.

A.2. Required System Configuration

Host Computer	IBM [1] PC/AT compatible
CPU	Pentium 4 (or Pentium M) or more is recommended.
OS	Windows 2000 and Windows XP
Memory	512MB or more is recommended.
Capacity of hard disk	200MB or more in empty capacity
Display	Resolution of SVGA or more
I/O device	CD-ROM drive
Else	Pointing device of mouse etc.

A.3. Installation Procedure

Insert the CD-ROM of the C Compiler Package into the CD drive of your computer, and High-performance Embedded Workshop Install Manager will start up automatically. Follow the messages displayed by High-performance Embedded Workshop Install Manager as you install the C Compiler Package.
If High-performance Embedded Workshop Install Manager does not start up automatically, execute "HewInstMan.exe" included in the CD-ROM.

```
D:¥> HewInstMan.exe

The drive name "D:" differs with each PC used.
```

Note, however, that before High-performance Embedded Workshop Install Manager starts up, all other applications must be closed.

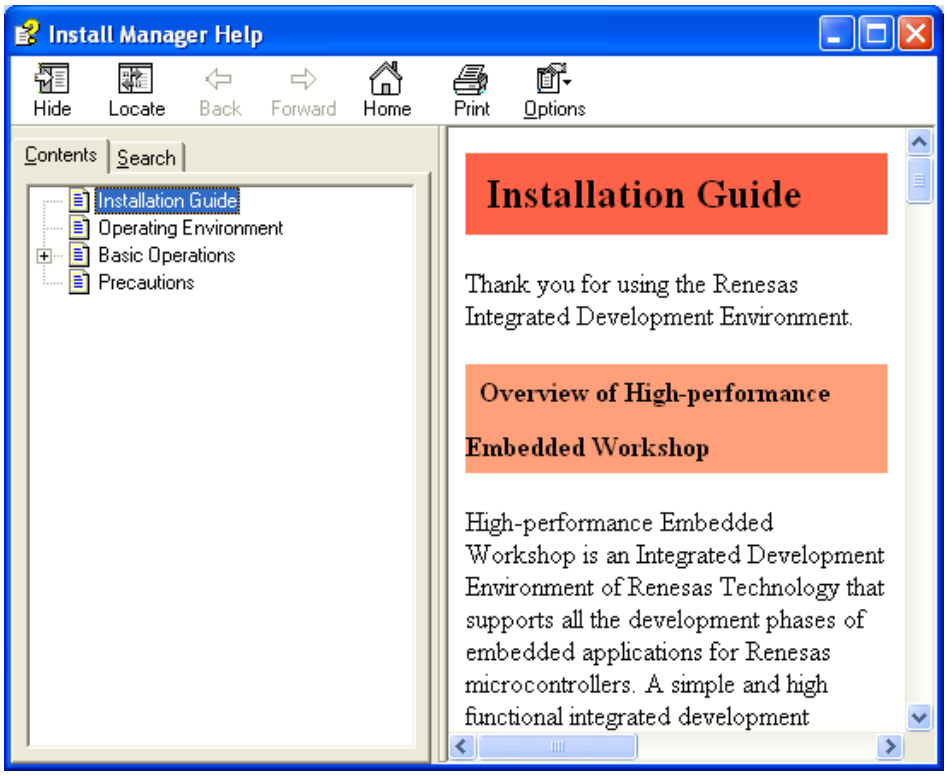
- Precautions
  [1] Make sure that High-performance Embedded Workshop V.4.04.01 is installed in a different directory than the one where the earlier version V.1.x is installed.
  [2] The projects created with High-performance Embedded Workshop V.1.0x cannot be used directly in V.4.04.01 To use any project created with V.1.0x in V.4.04.01, convert it to V.1.1x once (by opening a project in V.1.1 and then saving it).

The following describes the procedure for newly installing the C Compiler Package in your computer while High-performance Embedded Workshop Install Manager is up and running.

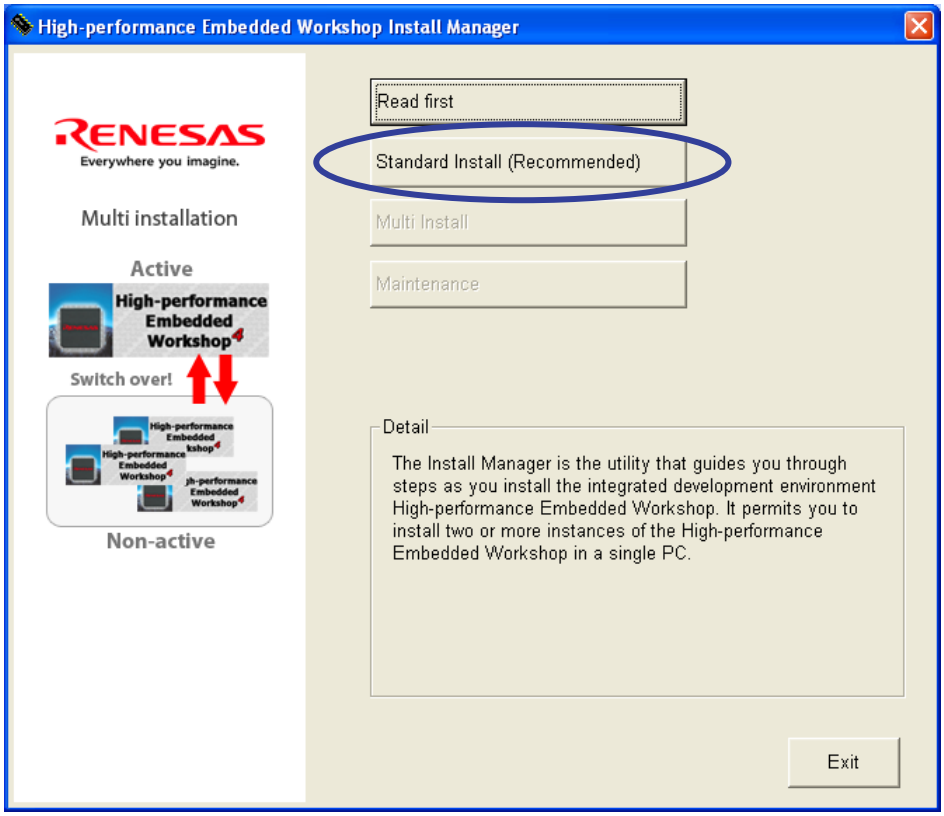
[1] IBM and AT are registered trademarks of International Business Machines Corporation.

A.3.1. When installing for the first time

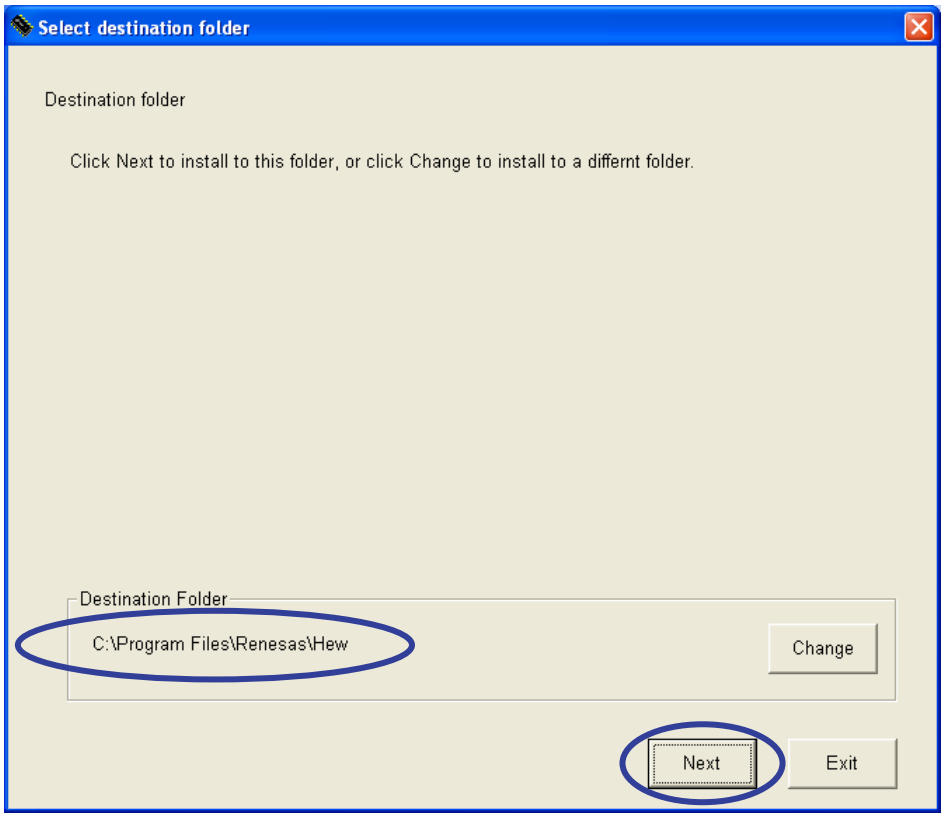
(1) When you are newly installing the C Compiler Package or when you click "Read first" in High-performance Embedded Workshop Install Manager, the Install Manager Help shown below is displayed on the screen. Please read it before you proceed.



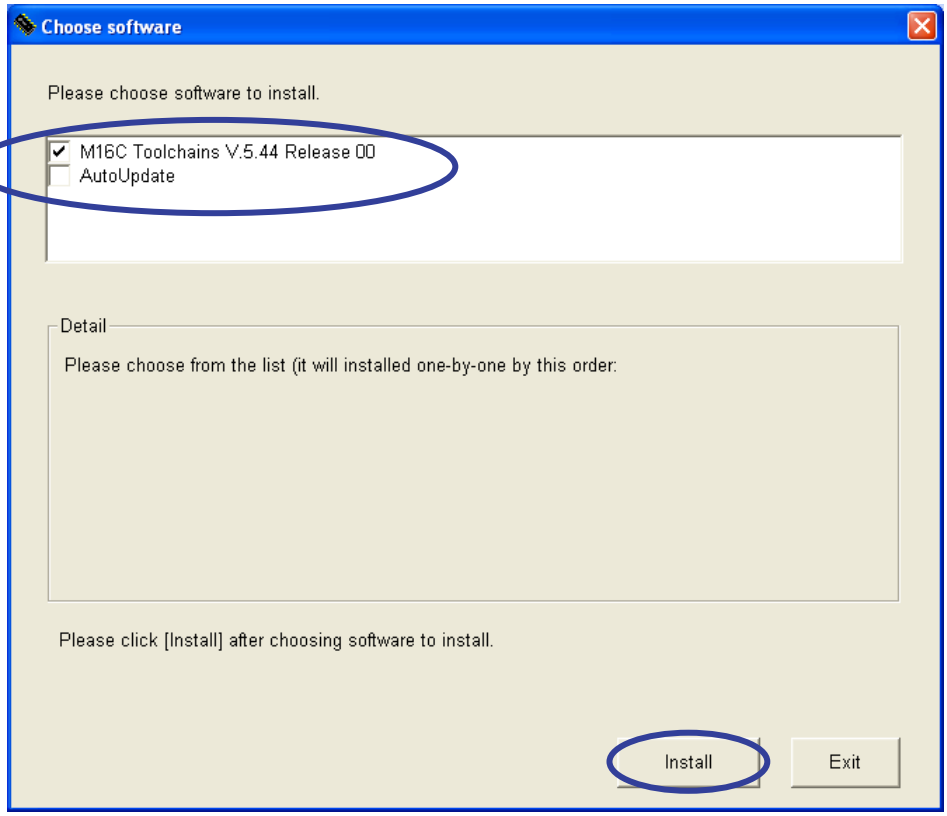
(2) Click the Standard Install(Recommended).



(3) Check the folder in which the High-performance Embedded Workshop will be installed and click Next.

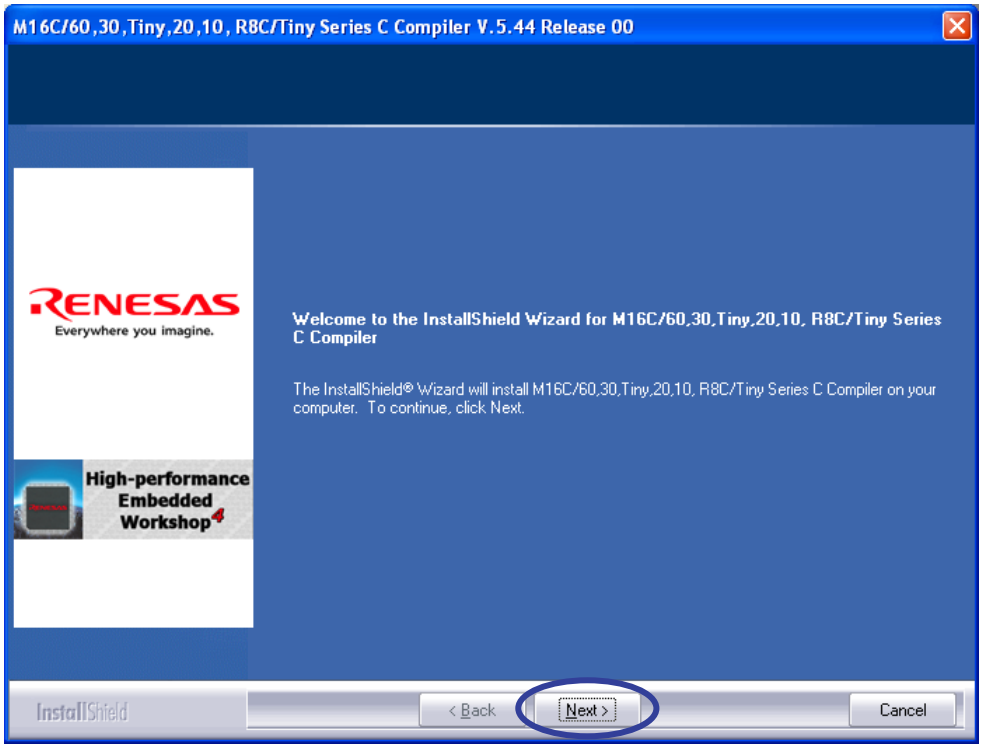


(4) Select check boxes and click Install.

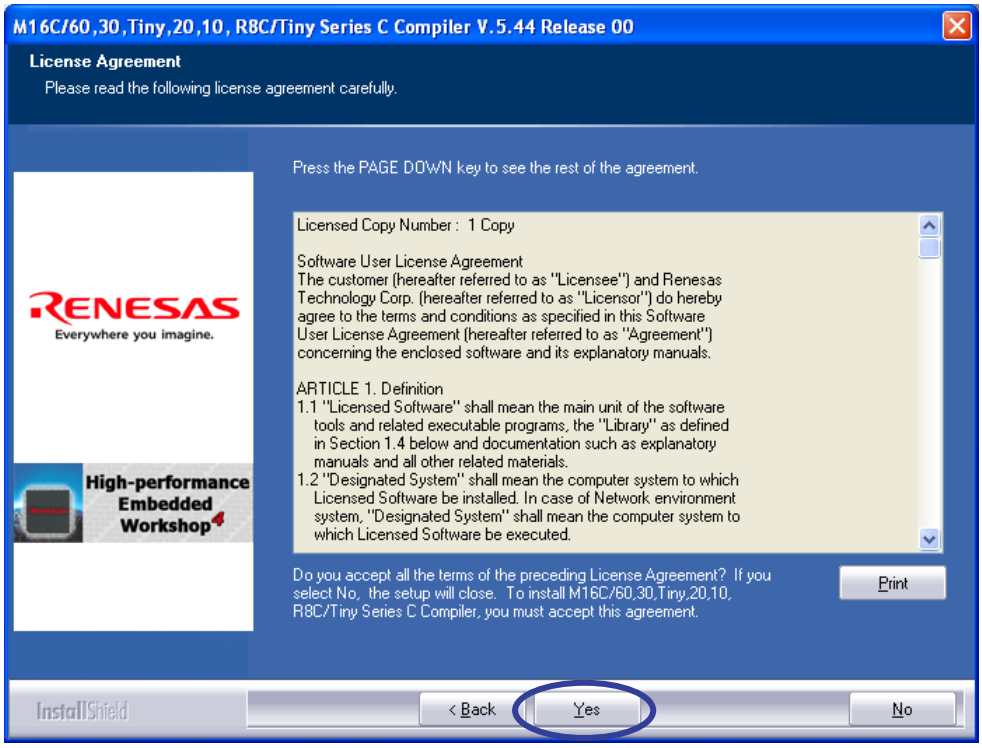


Only "M16C Toolchains" is checked. If you install AutoUpdate, check "AutoUpdate".

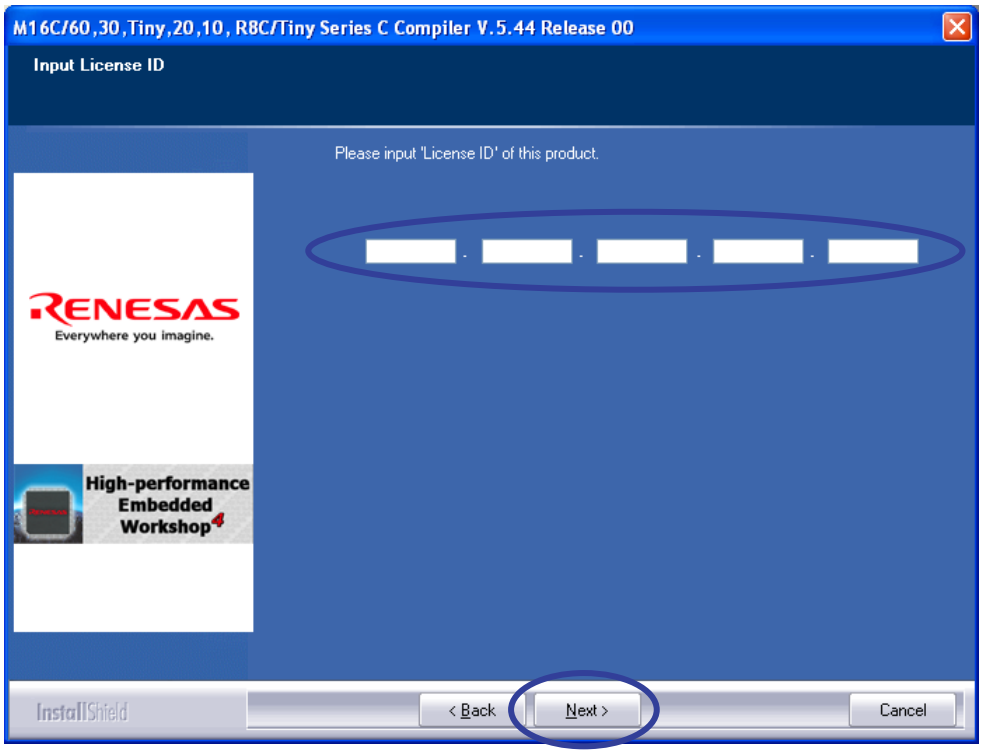
(5) Click Next.



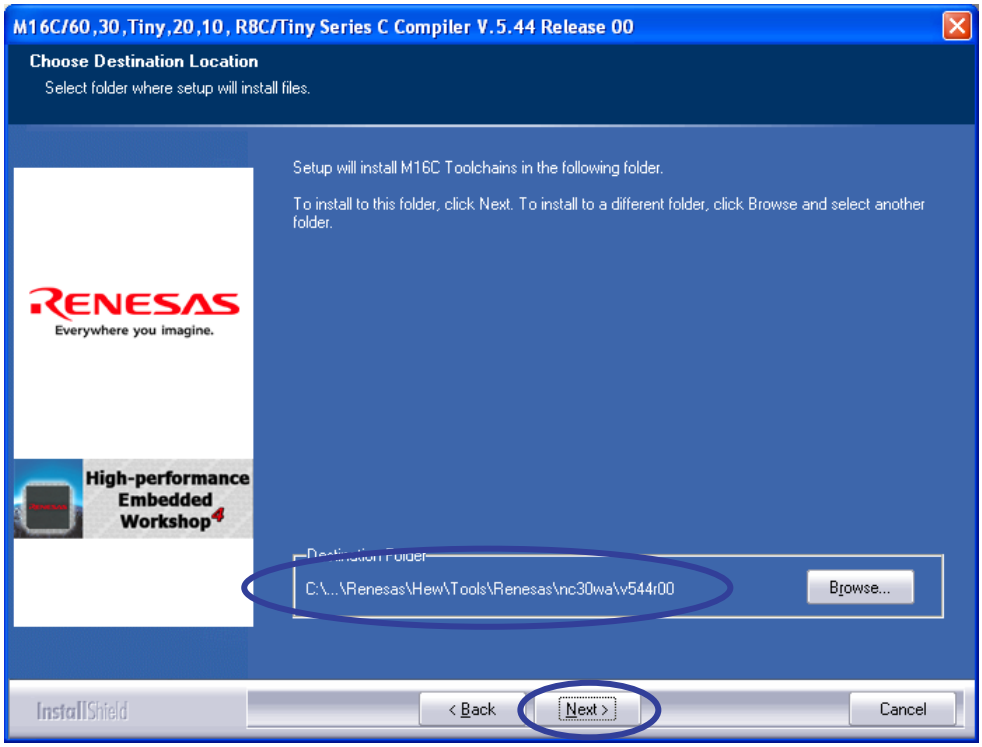
(6) Carefully read the Software User License Agreement and click Yes.



(7) Enter the license ID written in your license ID certificate and click Next.

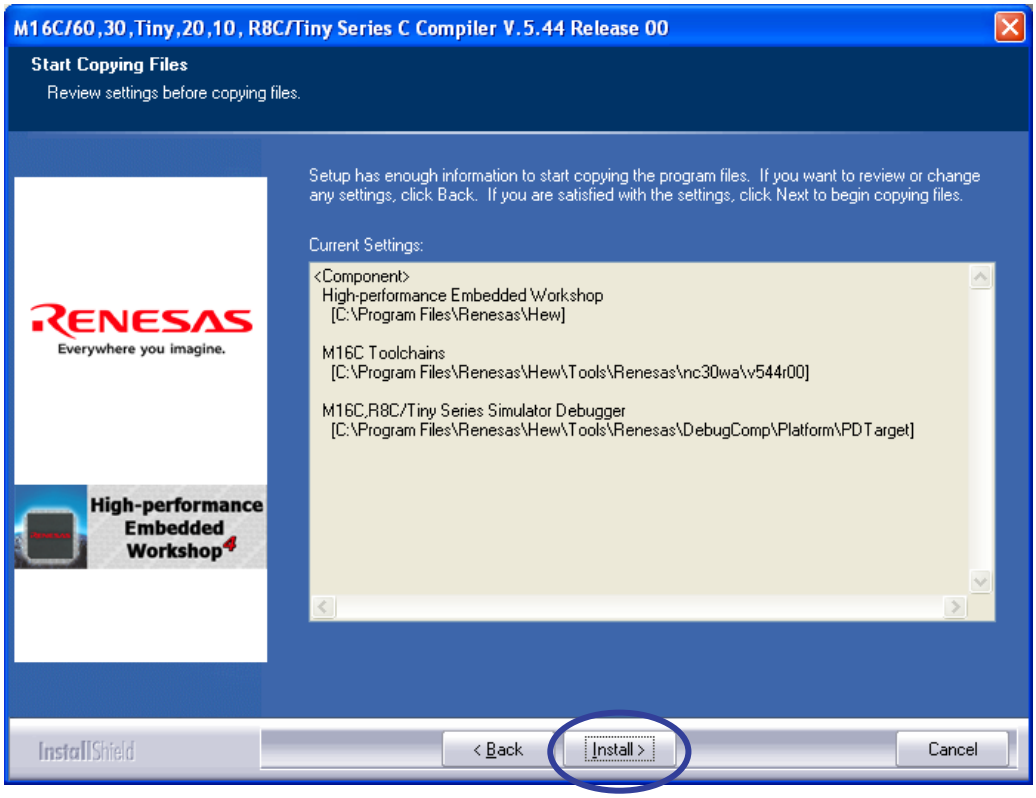


(8) Check the folder in which the C compiler will be installed and click Next.

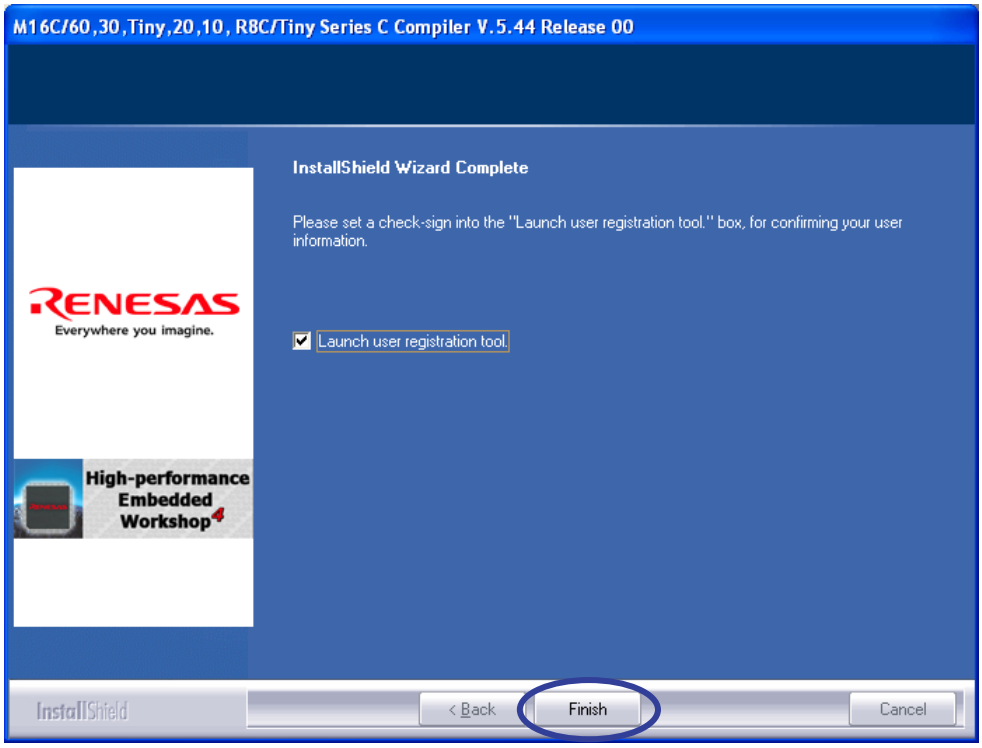


If changing the folder, click Browse.
Do not install the same folder with Hitachi Integration Manager or Hitachi Embedded Workshop.
Refer to Renesas Tool News about the latest information.

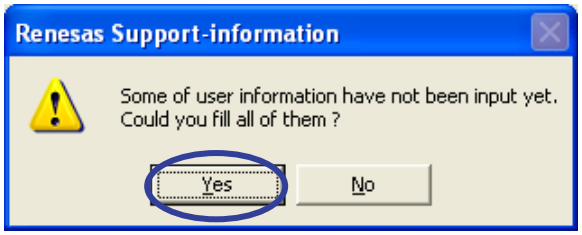
(9) Click Install.



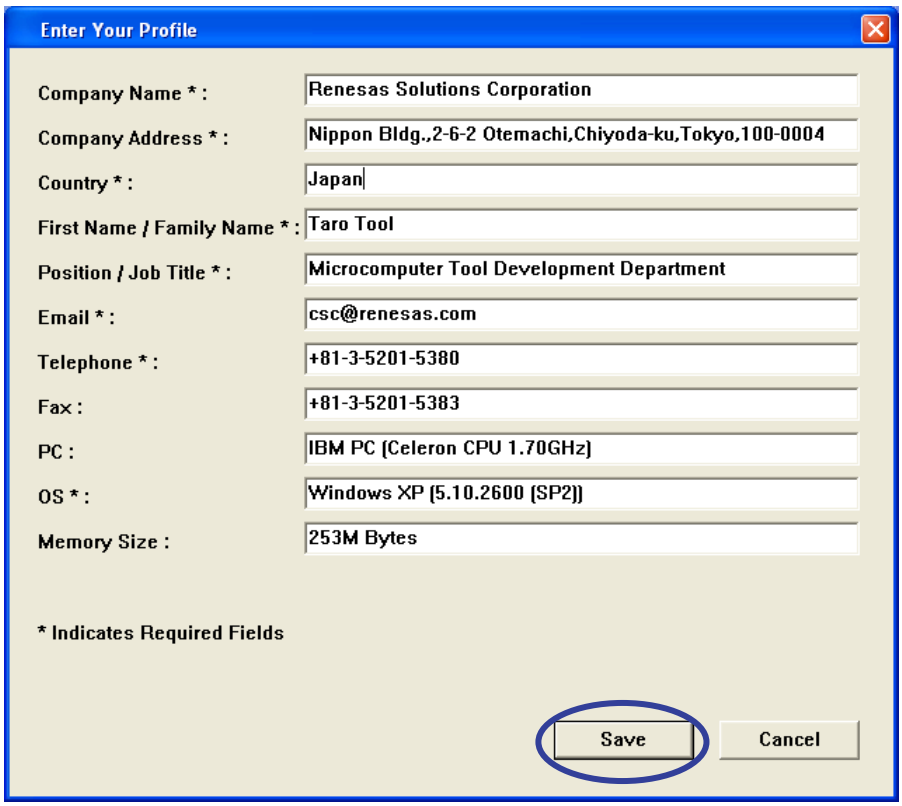
(10) Select the "Launch user registration tool" check box and click Finish.



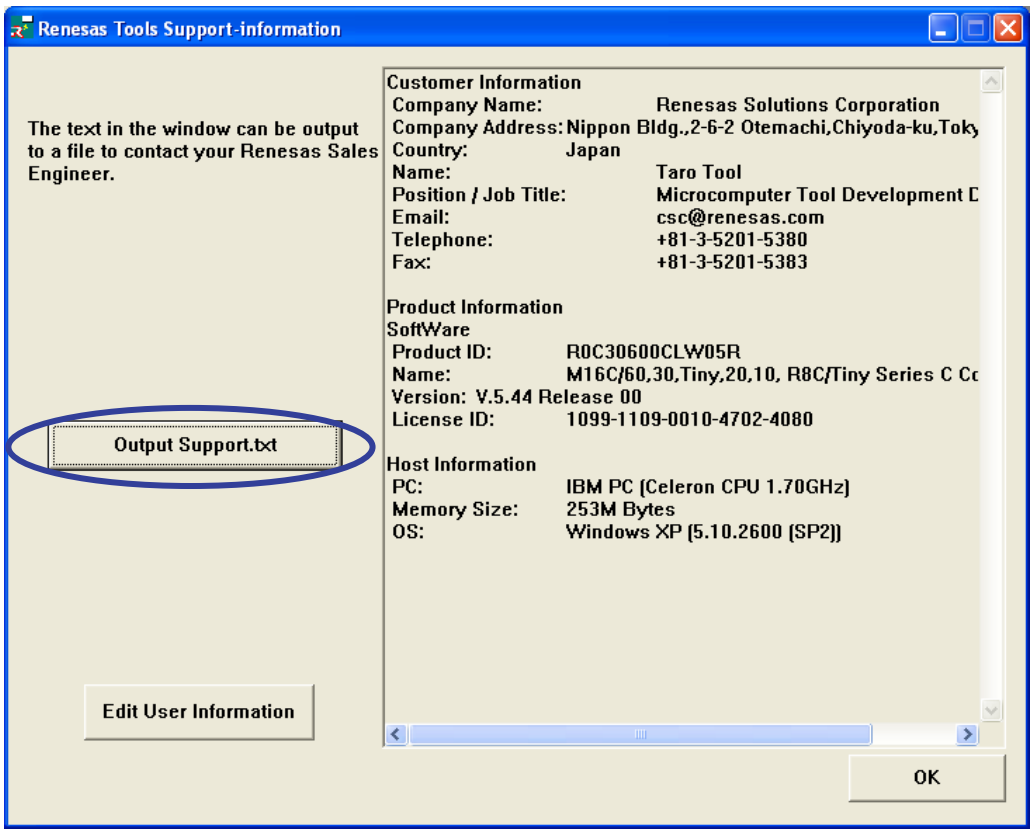
(11) Click Yes.



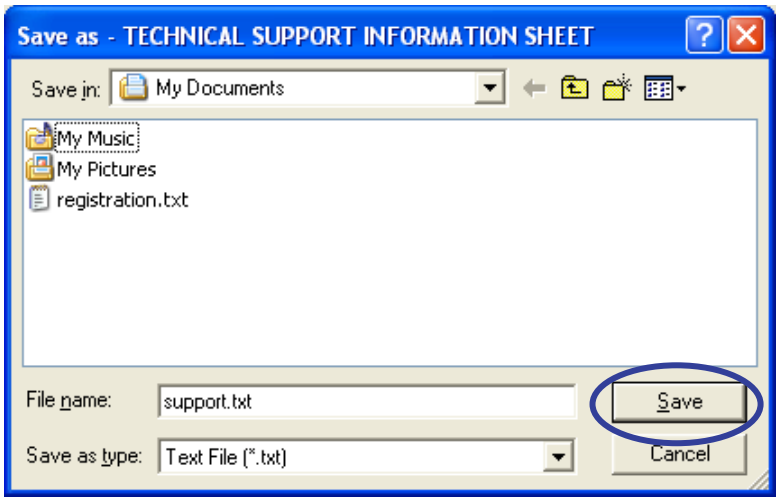
(12) Enter the name of your country in the Country text box and click Save.



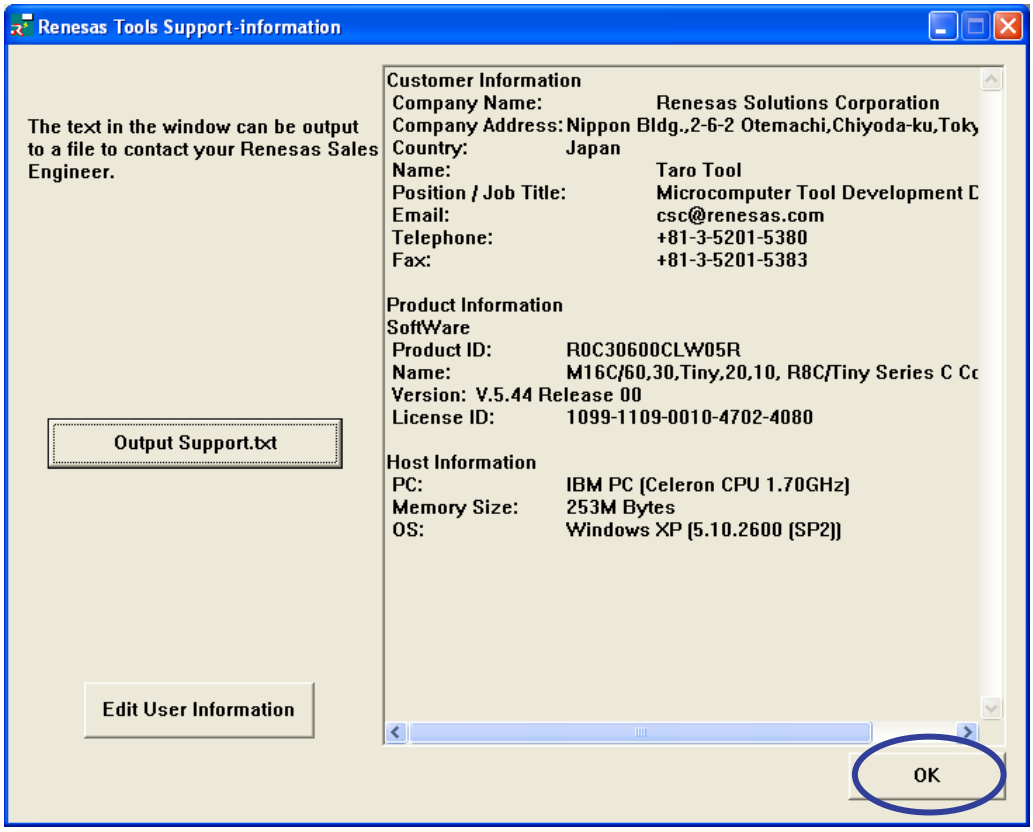
(13) Check the contents displayed in the Renesas Tools Support Information dialog box and click Output Support.txt. If you want to correct the displayed content, click Edit User Information and make corrections.



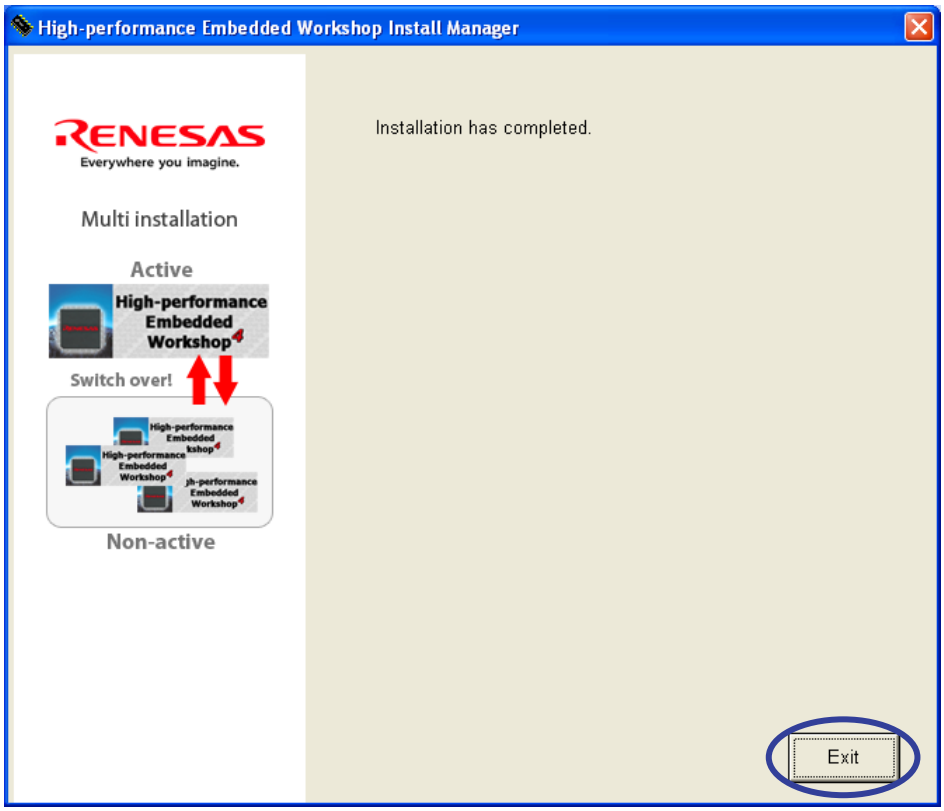
(14) Select any location in which you want the TECHNICAL SUPPORT INFORMATION SHEET file (support.txt) to be created and click Save.



(15) Click OK.



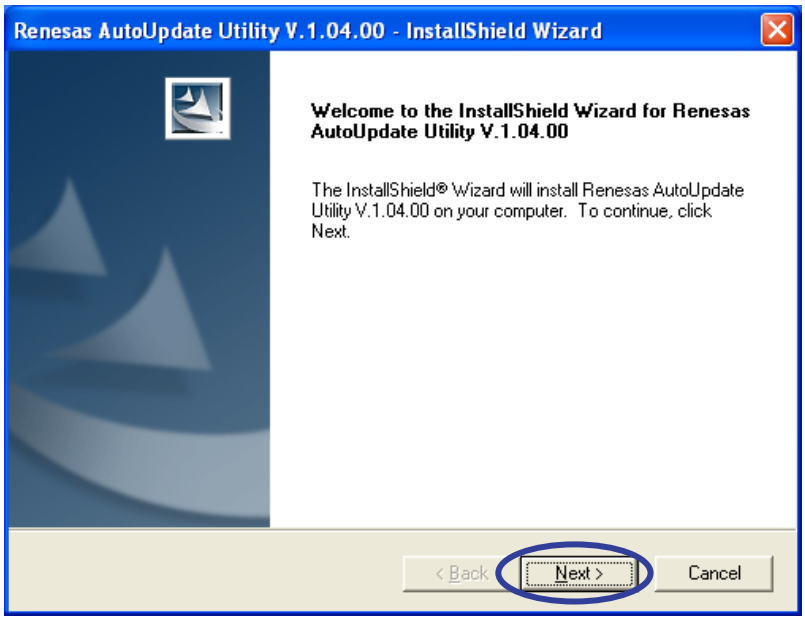
(16) Click Exit to close High-performance Embedded Workshop Install Manager.



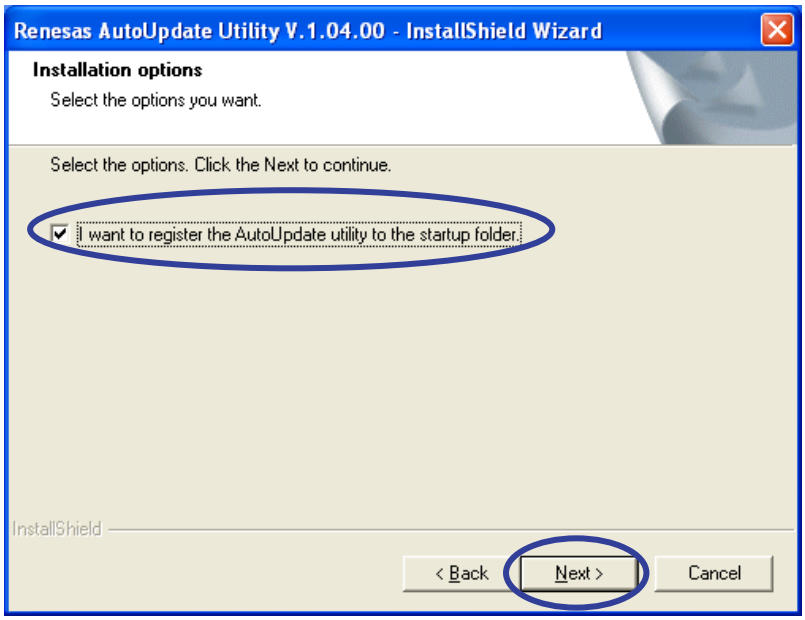
A.3.2. When installing AutoUpdate

Check AutoUpdate in “Choose software” dialog box. (Refer to Page 6).

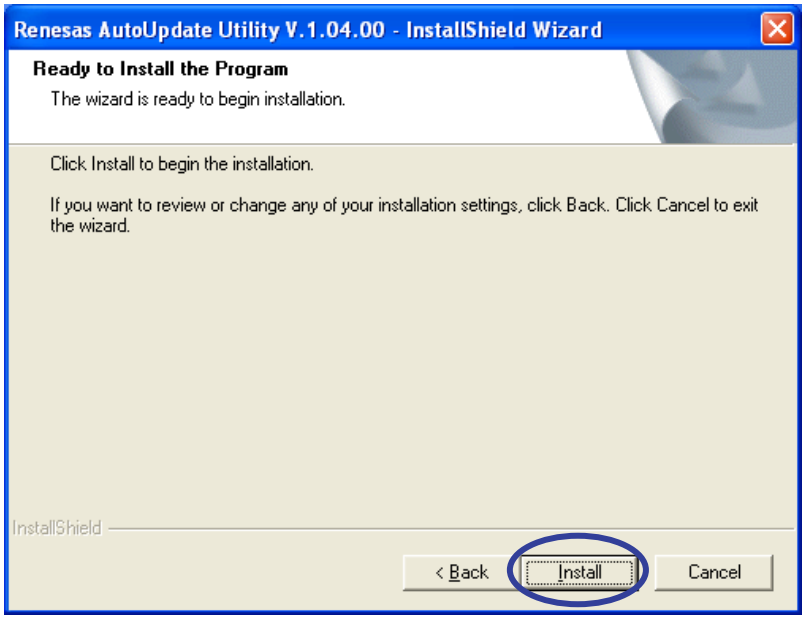
(1) Click Next.



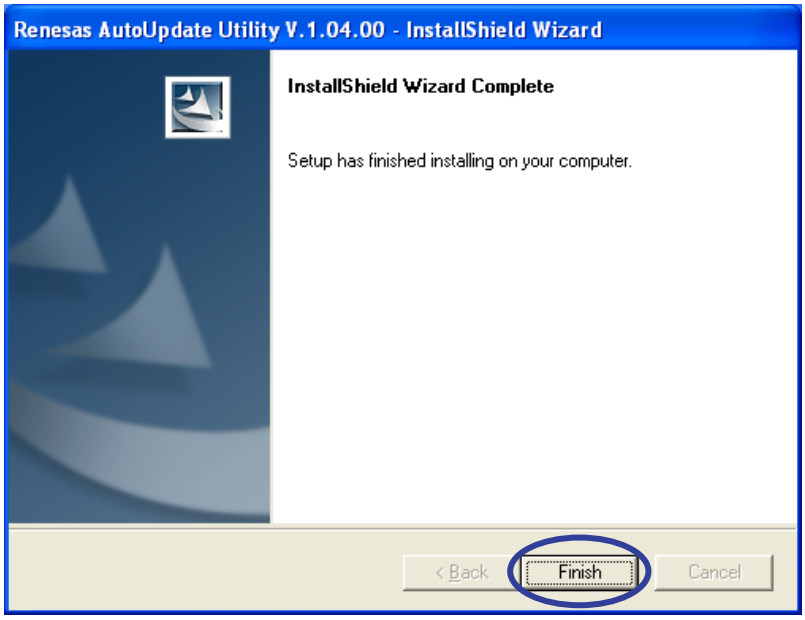
(2) Check the folder in which the " “I want to register the AutoUpdate utility to the startup folder. " will be installed and click Next.



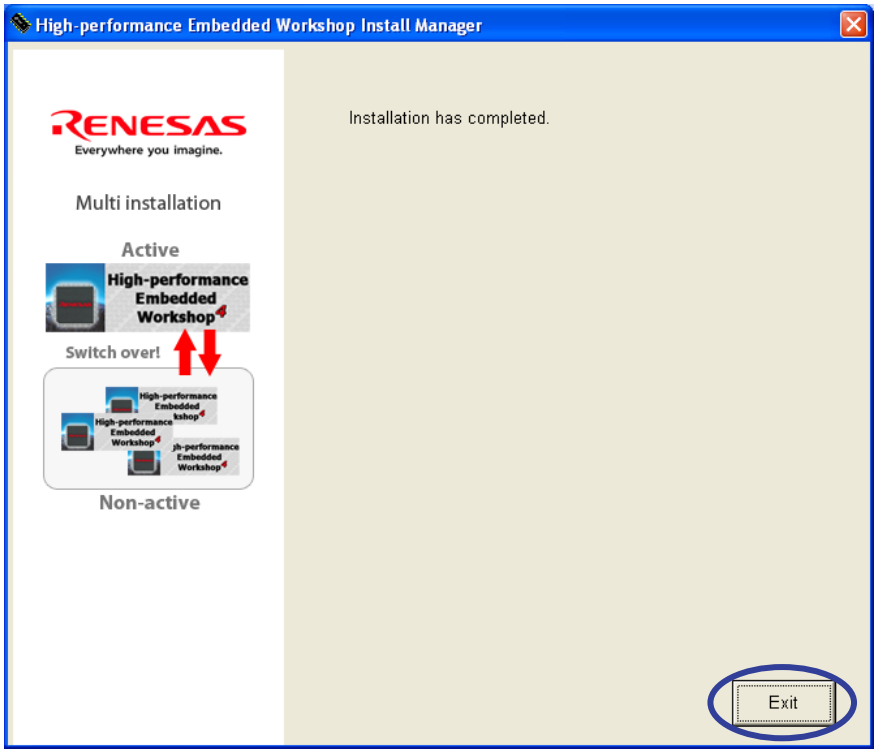
(3) Click Install.



(4) Click Finish.

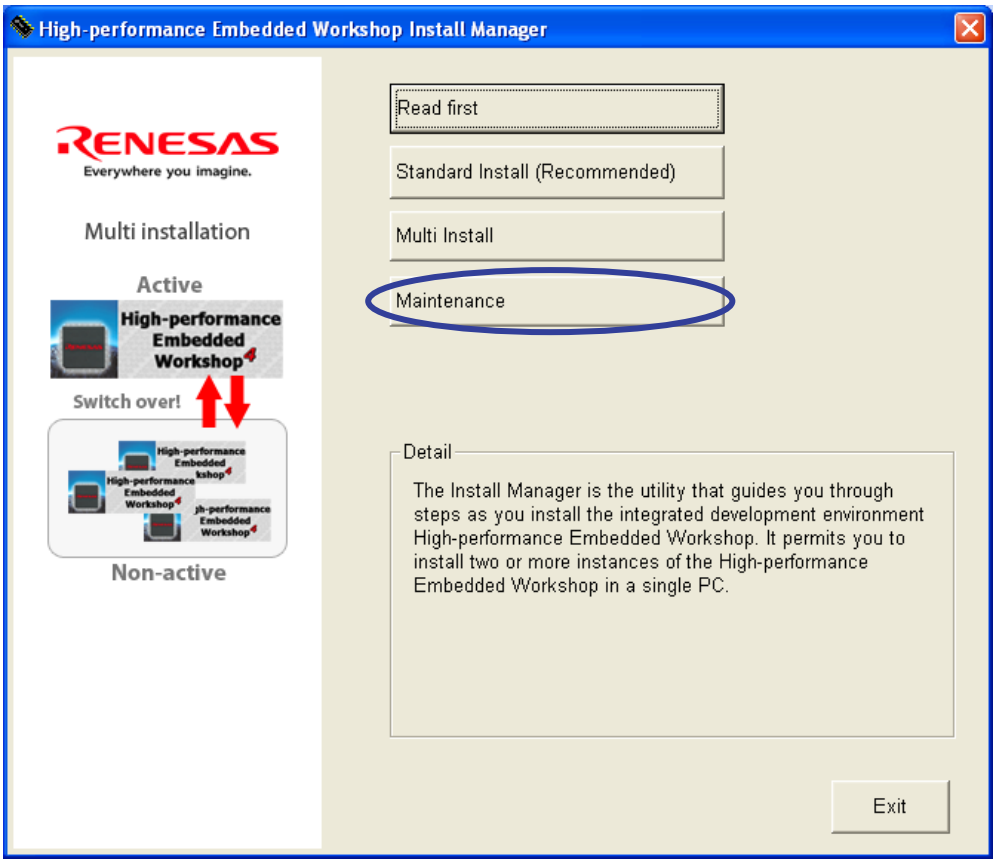


(5) Click Exit to close High-performance Embedded Workshop Install Manager.

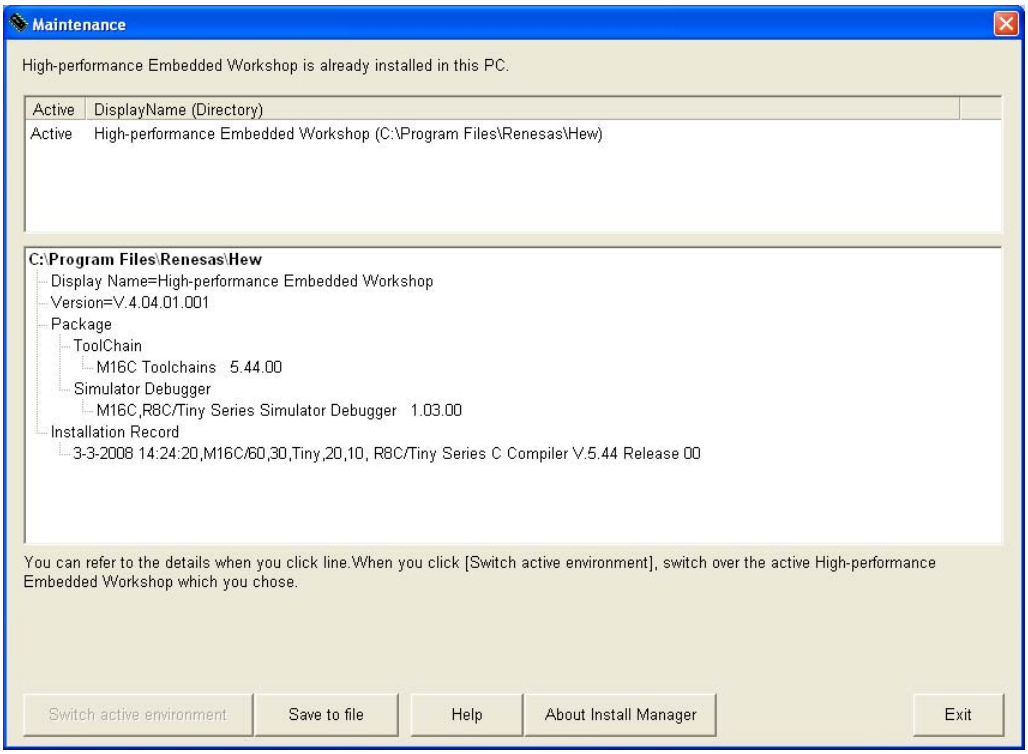


A.3.3. When checking of the existing environment

(1) Click Maintenance

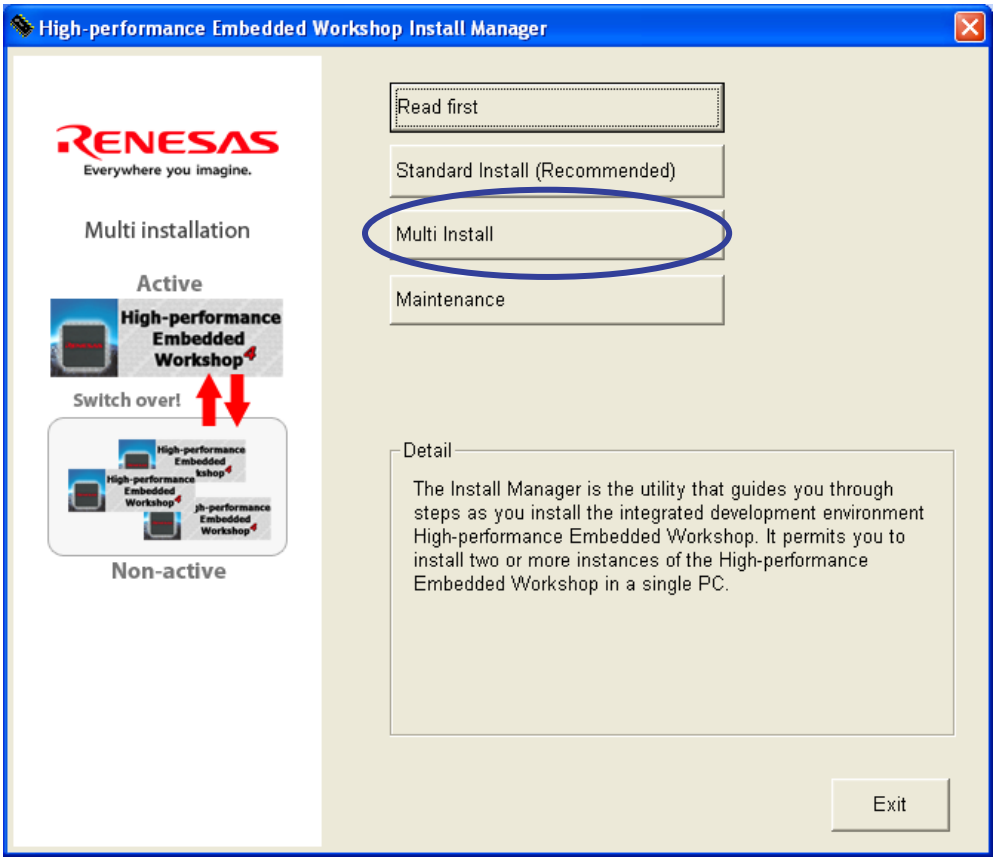


(2) Check of the environment.



A.3.4. When installing two or more High-performance Embedded Workshop in a single PC.

(1) Click Multi Install.



Refer to the Install Manager Help about Multi Install.

A.4. Uninstalling programs

The installed programs can be uninstalled according to the following instructions. Begin the un-installation after closing all the applications.

(1) Select [Control Panel] on the Windows [Start] menu.
(2) Select the [Add or Remove Programs] icon.
(3) Click [High-performance Embedded Workshop(Multiple Install)] on the [Add or Remove Programs] tab and click the [Remove] button.
(4) Then, Uninstall dialog of Install Manager is displayed. Choose [High-performance Embedded Workshop] in the list and click [Uninstall] button.
(5) Follow the instruction displayed on the screen.
(6) After removing the program, restart Windows.

A.5. Startup or termination of program

A.5.1. Startup and termination of the High-performance Embedded Workshop

- Startup
  Click [High-performance Embedded Workshop] in the [High-performance Embedded Workshop] folder in the [Renesas] folder in the [Program] folder of the Windows [Start] menu.
- Termination
  Click [Exit] on the [File] menu.

A.5.2. Start of Manual Navigator

- Startup
  Click [Manual Navigator] in the [High-performance Embedded Workshop] folder in the [Renesas] folder in the [Program] folder of the Windows [Start] menu.
- Termination
  Termination: Click [Exit] on the [File] menu.

A.5.3. Displaying the online manuals and attached documents

Displaying the online manuals and attached documents

- Note
  [1] Manual Navigator requires Adobe Reader [2] .
  [2] If Manuals folder is moved, Manual Navigator cannot show them.

A.6. Setting when compiler is used on DOS prompt and command prompt

Please execute setnc30.bat when you use the compiler on the DOS prompt and the command prompt.

A.6.1. Environment Variables and Path

Environment variable	Usage
BIN30	Directory in which the C compiler execution files (e.g., *exe) are stored
INC30	Directory in which the standard include files of the C compiler are stored
LIB30	Directory in which the standard library files of the C compiler are stored
TMP30	Directory in which the temporary files generated by the C compiler are stored
PATH	Directory in which the C compiler generates temporary files Select the directory for which you have access rights.

A.6.2. Batch file

A batch file named "setnc30.bat" will be generated in the directory in which you've installed the C compiler. This file has written in it the environment variables that the C compiler uses.
To use the C compiler from the DOS or the command prompt, execute setnc30.bat.

- Contents written in the batch file

```
REM ***** Environment variable for M16C Toolchains *****
SET BIN30=C:¥Program Files¥Renesas¥Hew¥Tools¥Renesas¥nc30wa¥v544r00¥BIN
SET LIB30=C:¥Program Files¥Renesas¥Hew¥Tools¥Renesas¥nc30wa¥v544r00¥LIB30
SET INC30=C:¥Program Files¥Renesas¥Hew¥Tools¥Renesas¥nc30wa¥v544r00¥INC30
SET TMP30=C:¥Program Files¥Renesas¥Hew¥Tools¥Renesas¥nc30wa¥v544r00¥TMP
SET PATH=%BIN30%;%PATH%
```

[2] Adobe and Acrobat are registered trademarks of Adobe Systems Incorporated.

B. Guide Book for V5.44 Release 00

This section describes the precautions to be taken when you upgrade the projects you created with old versions of the compiler to V.5.44 Release 00, and the points to be noted when you create new projects with V.5.44 Release 00.

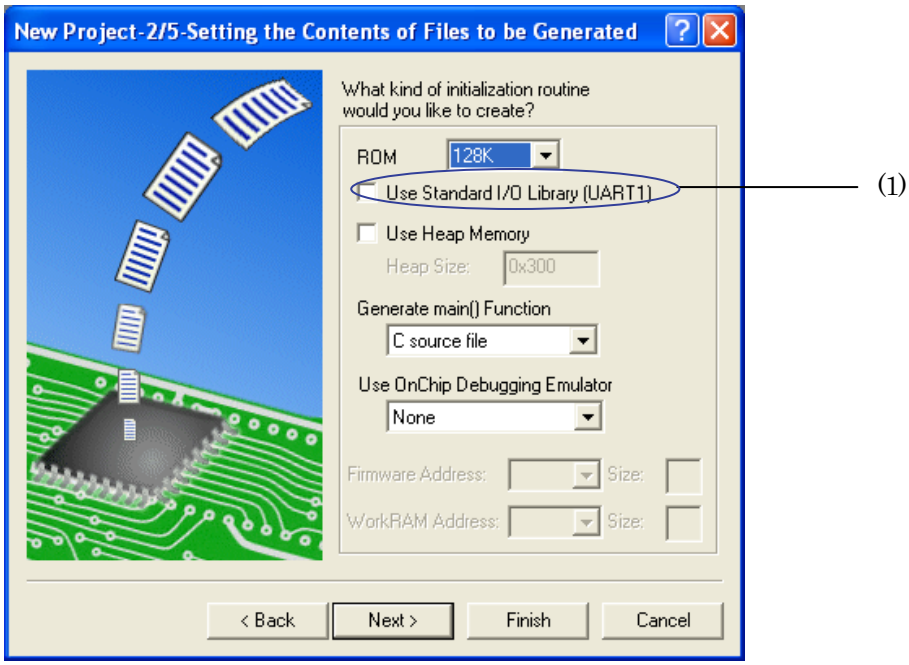
B.1. Points to be noted when you upgrade from old versions of the compiler to V.5.44 Release 00 and use it in combination of Renesas real-time OS.

B.1.1. Modify startup file

・_init() function

Beginning with V.5.40 Release 00(A), the name of the library function init() has been changed to _init().
Therefore, if you attempt to build without modification, an error message *'_init' value is undefined* may be generated during a link process. This error occurs in the following cases:

・If you created the project with an old version prior to V.5.40 Release 00, select the check box (1)
・When init function calls are enabled by altering ncrt0.a30 directly



If this error occurs, alter a part of ncrt0.a30 that is shown below

[When you are using the startup file (ncrt0.a30) supplied with the compiler]

Before modification

```
;======================================================================
; Initialize standard I/O
;----------------------------------------------------------------------
.if __STANDARD_IO__ == 1
        .glb    _init
        .call   _init,G
        jsr.a   _init
.endif
```

After modification

```
;======================================================================
```

```
; Initialize standard I/O
;------------------------------------------------------------------
.if __STANDARD_IO__ == 1
        .glb    __init
        .call   __init,G
        jsr.a   __init
.endif
;------------------------------------------------------------------
```

[When you are using the startup file (crt0mr.a30) supplied with the Real Time OS]

For M3T-MR308

Before modification

```
; +-------------------------------------------------+
; |    User Initial Routine ( if there are )        |
; +-------------------------------------------------+
; Initialize standard I/O
        .GLB    _init
        JSR.A   _init
```

After modification

```
; +-------------------------------------------------+
; |    User Initial Routine ( if there are )        |
; +-------------------------------------------------+
; Initialize standard I/O
        .GLB    __init
        JSR.A   __init
;-------------------------------------------------
```

For M3T-MR30

* In several versions including the latest version (V.3.30 Release 2), the jump process shown below is 'commented out.'

Before modification

```
;===================================================================
; Initialize standard I/O
;-------------------------------------------------------------------
        .glb    _init
        jsr.a   _init
```

After modification

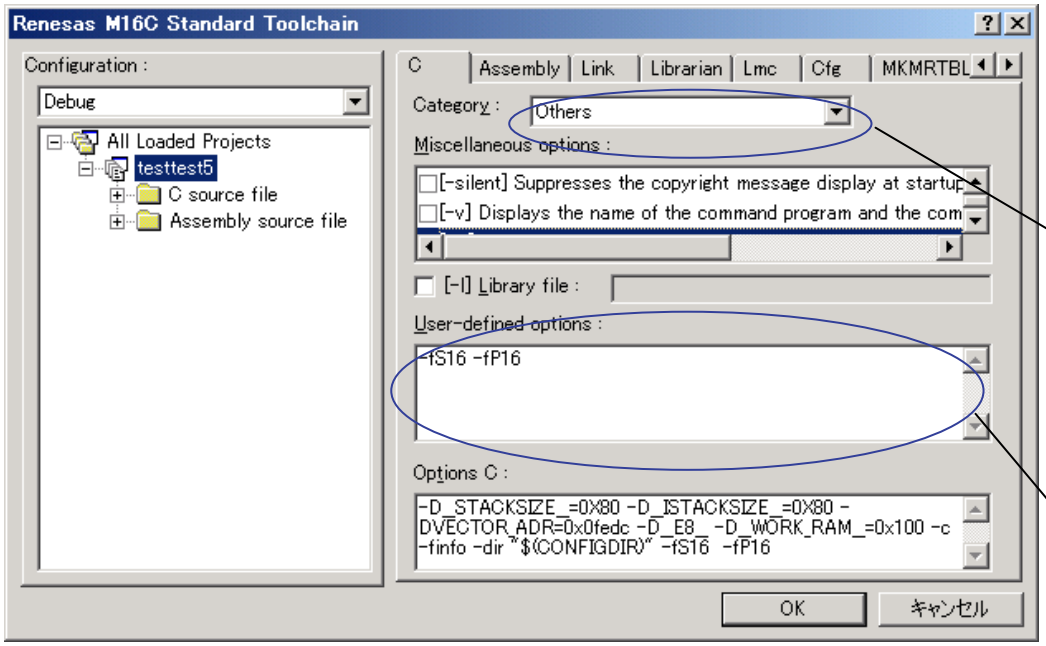
```
;===================================================================
; Initialize standard I/O
;-------------------------------------------------------------------
        .glb    __init
        jsr.a   __init
```

B.1.2. Change size of size_t, ptrdiff_t

Beginning with this version, size_t and ptrdiff_t have been changed in size from 16 bits to 32 bits.
If you need to use size_t and ptrdiff_t in 16 bits because you are using a size_t and ptrdiff_t type based user library created with an old version, for example, make the following settings.

- Set the compile options -fsize_t16 (-fS16) and -fptrdiff_t (-fP16).
- Change the libraries to be linked from nc30lib.lib and r8clib.lib to nc30s16.lib and r8cs16.lib when you're using NC30WA, or from nc308lib.lib and nc382lib.lib to nc308_16.lib and nc382_16.lib when you're using NC308WA.

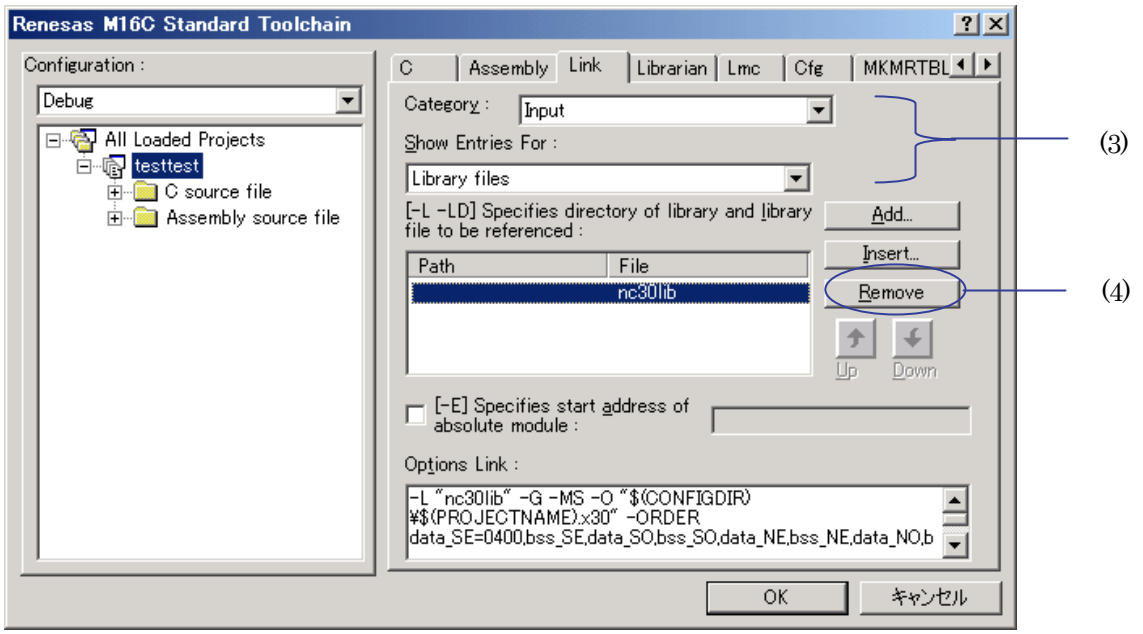
[Procedure for setup in HEW]
Set the compile options -fsize_t16 (-fS16) and -fptrdiff_t (-fP16).
From the Build menu of HEW, select [Renesas M16C Standard Toolchain] -> C tab.



(1)For Category,select Other
(2)For User-Defined Options, enter -fsize_t16 (or -fS16) and -fptrdiff_t16 (or -fP16).

Change the libraries to be linked.

From the Build menu of HEW, select [Renesas M16C Standard Toolchain] -> Link tab.



(3) For Category,select Input

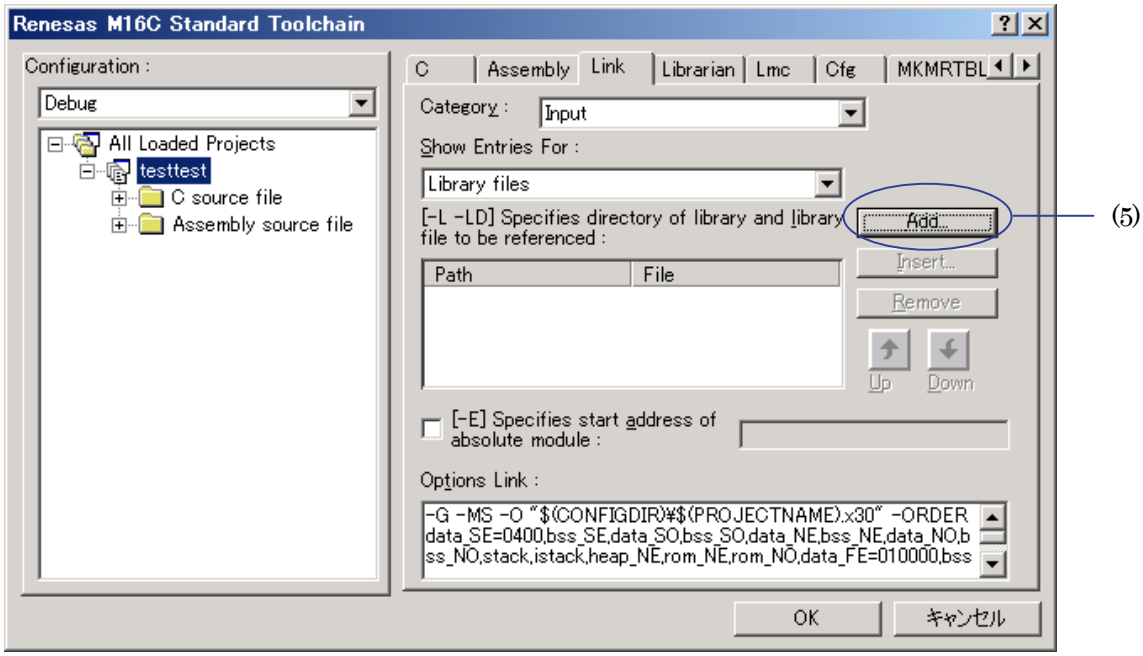
For Show Entries-For select Library files.

(4) Click the Remove button to remove nc30lib.lib temporarily.

Remove r8clib.lib when you're using R8C/Tiny.

Remove nc308lib.lib when you're using M16C/8X

Remove nc382lib.lib when you're using M32C/8X



(5) Click the Add button to select [Library files].

(6)Enter a library name usable for size_t and ptrdiff_t in 16-bit size.
Input nc30s16.lib when you're using M16C
Input r8cs16.lib when you're using R8C/Tiny.
Input nc308_16.lib when you're using M16C/8X
Input nc382_16.lib when you're using M32C/8X
Click the OK button to finish.

[When you're using the makefiles generated by the configurator of the real-time OS]
Correct the following part of statements in the makefile. The following shows an example for the case where nc308_16.lib is linked.

Before modification

```
# Use the following macro when you use C-libraries for M32C/80 series.
#NEWLIB          = -l nc382lib
```

After modification

```
# Use the following macro when you use C-libraries for M32C/80 series.
#NEWLIB          = -l nc308_16.lib
```

[when you're using M3T-MR30]
Since the "LIBS" macro in the makefile is rewritten to "nc30lib.lib" by the configurator, the problem cannot be solved by correcting the "LIBS" macro.
Therefore, correct a process during "$(LINKLIST)" generation to solve the problem.
Correct the following part of statements in the makefile.

Before modification

```
$(LINKLST): makefile
    @mrecho "-o $(PROGRAM)" $(LINKLST)
    @mrecho -a "-ld $(LIB30)" $(LINKLST)
    @mrecho -a "-l $(LIBS)" $(LINKLST)
```

After modification

```
$(LINKLST): makefile
    @mrecho "-o $(PROGRAM)" $(LINKLST)
    @mrecho -a "-ld $(LIB30)" $(LINKLST)
    @mrecho -a "-l nc30s16.lib -l $(LIBS)" $(LINKLST)
```

B.1.3. Interrupt vector

Beginning with this version, a vector table will be automatically generated when interrupt functions are declared by specifying vector numbers.
In old versions, it was necessary to specify the option -fmake_vector-table(-fMVT). But this is unnecessary.
When projects are upgraded, this option is not inherited. Therefore, when you build a project, an error "Can't generate automatically the variable interrupt vector table" may be generated when linking. If this error occurs, correct sec30.inc following the method B.2.3.

B.2. Points to be maked new project when you use V.5.44 Release 00

B.2.1. Select CPU

When creating a new project you can select the type of microcomputer in CPU Group. However, the selection of microcomputer types is enabled when

- Registration of the sfr header file
- Registering the variable vector interrupt table entry function registration file (intprg.c) to the workspace
- Link address settings

B.2.2. To create a new workspace with a microcomputer that is not listed in CPU Group

[In the case of R8C/Tiny series]

(1) Choose R8C/Tiny from CPU Series.
(2) Choose Other from CPU Group.

When you create a new project using V.5.44 Release 00, please check whether the compile option and the library file suit the ROM size of the microcomputer type you use on Renesas M16C Standard Toolchain Dialog Box of HEW.
From the Build menu of HEW, select [Renesas M16C Standard Toolchain]

ROM size	compile option	library
Less than 64 Kbytes	-R8C	r8clib.lib
64 Kbytes or more	-R8CE	nc30lib.lib

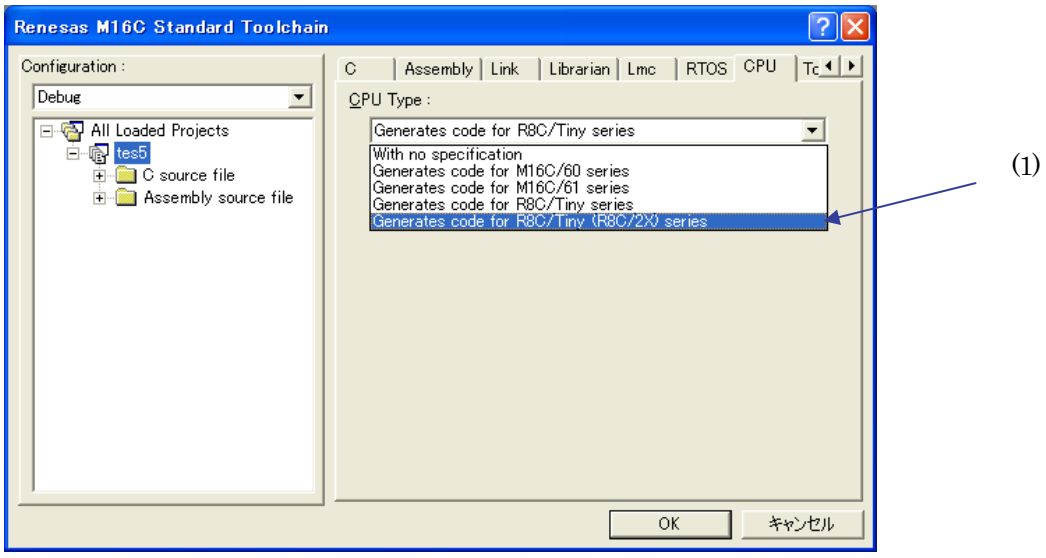
When you create a new project using V.5.43 Release 00 or earlier, please make a change shown below.

If the ROM space of the microcomputer type you use exceeds the 64 Kbyte boundary, make the following settings.

(3) Change the compile option from -R8C to -R8CE.
(4) Change the library to link from r8clib.lib to nc30lib.lib.

Change the compile option from -R8C to -R8CE.

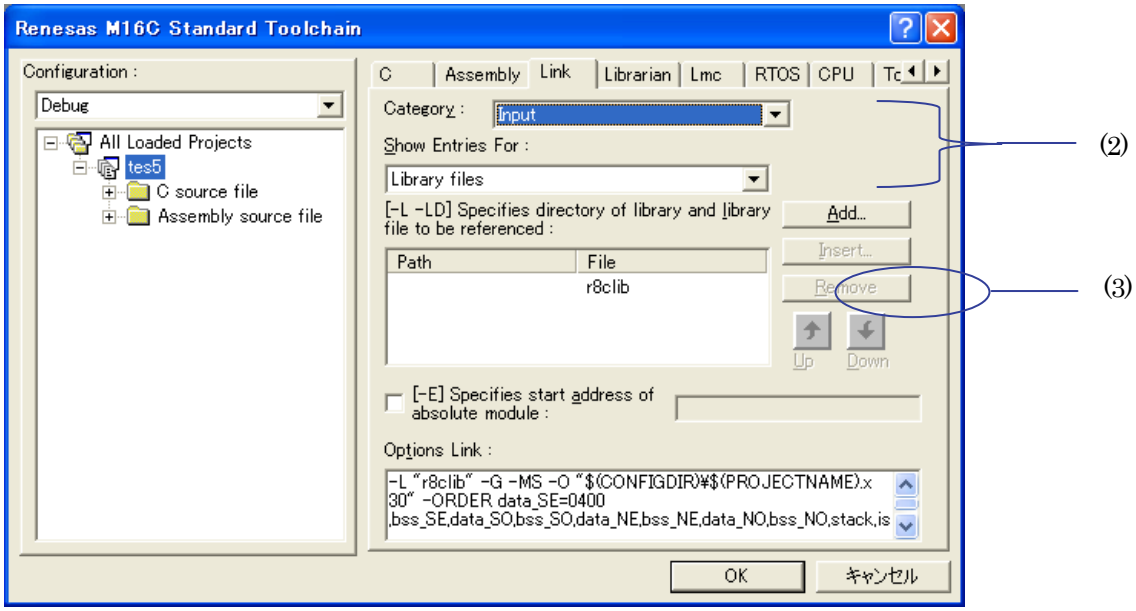
From the Build menu of HEW, select [Renesas M16C Standard Toolchain] -> CPU tab.



(1) From the CPU Type pulldown menu, select Generates Code for R8C/Tiny(R8C/2X) Series.

Change the linked library from r8clib.lib to nc30lib.lib

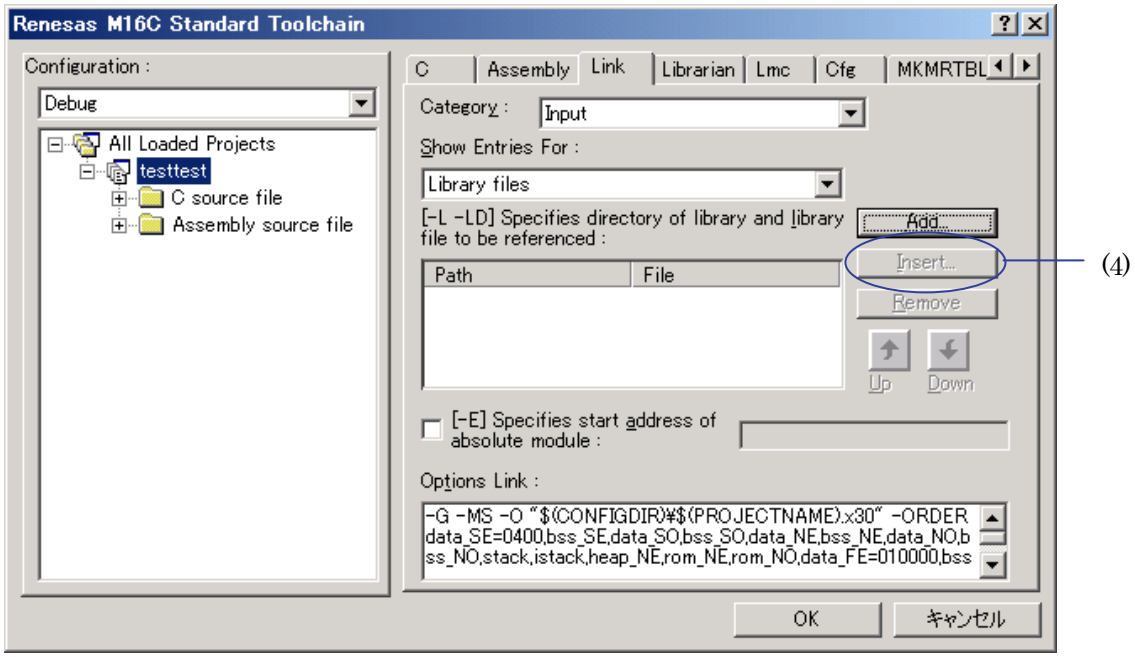
From the Build menu of HEW, select [Renesas M16C Standard Toolchain] -> Link tab.



(2) For Category,select Input

For Show Entries-For select Library files.

(3) Click the Remove button to remove r8clib.lib temporarily.



(4) Click the Add button to select [Library files].

(5)Input nc30lib.lib and click the OK button to finish.

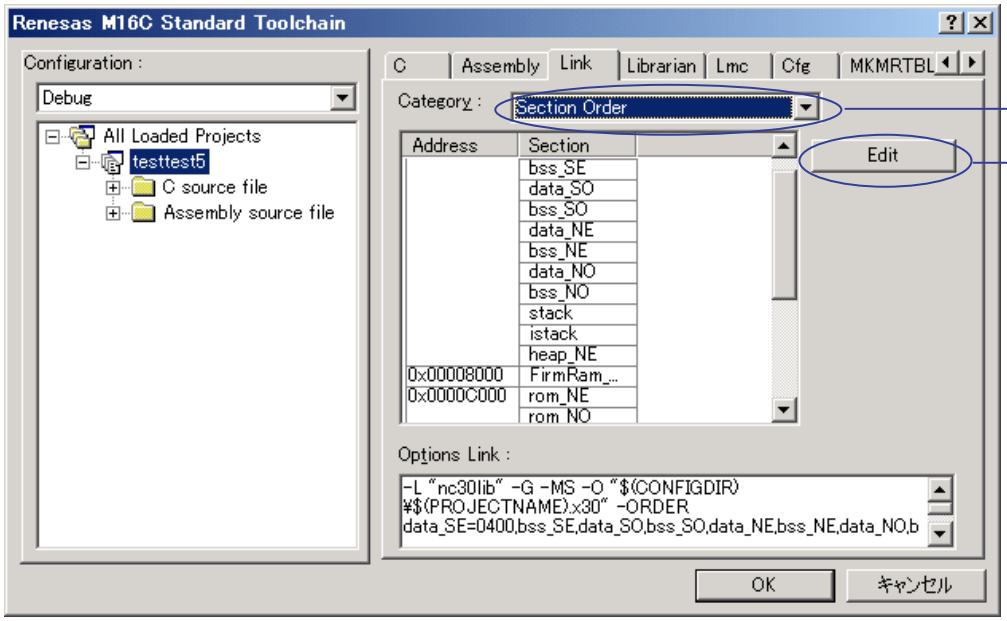
[To create a new workspace using microcomputers other than the R8C/Tiny]

- Choose the relevant CPU series from CPU Series.
- Choose Other from CPU Group.

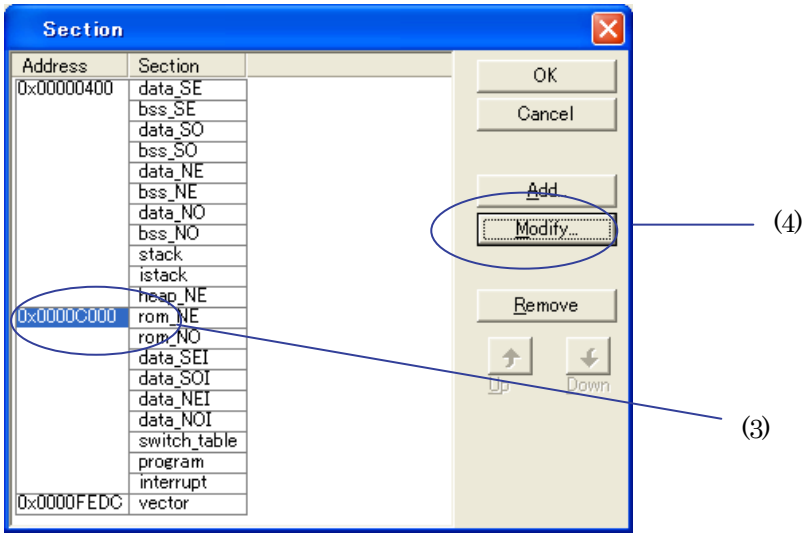
In addition to the above selections, the following requires caution.
・No sfr header files are registered.
Acquire the sfr header file corresponding to the microcomputer you use from the Web site, or create one if necessary.
・The order of sections is inaccurate.
Change the address of each section according to the ROM and RAM spaces of the microcomputer you use.
・No variable vector interrupt entry functions (intprg.c) are registered.
The interrupt functions should be created in a user file.

Change the section's address at linking

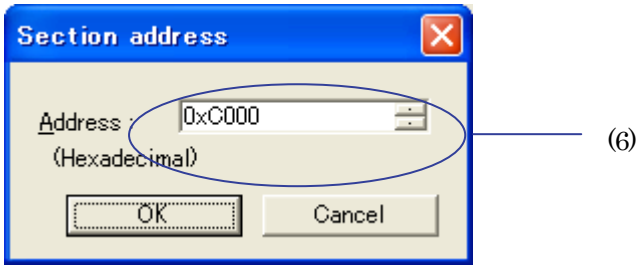
From the Build menu of HEW, select [Renesas M16C Standard Toolchain] -> Link tab.



(2) For Category,select Section-Order
Click the Edit button



(3) select address
(4) click the Modify botton



(6)input the address

B.2.3. When using an assembler startup

Instead of using a C language startup program, if you want to use the startups written in assembler ncrt0.a30 , sect30.inc (for NC308WA, sect308.inc) or nc_define.inc to create a project and the following applies, you need to make corrections.

・No vector numbers are specified when interrupt functions are declared.

If you build a project without making this correction, an error "Can't generate automatically the variable interrupt vector table" may be output when linking.

Content of correction

When using NC30WA

[sect30.inc]

```
.if        __MVT__==0

;-----------------------------------------------------------
; variable vector section
;-----------------------------------------------------------
        .section   vector,ROMDATA   ; variable vector table
        .org       VECTOR_ADR
.if  0  <------------------------------------------------------  Deletes .if 0 to enable .lword.
        .lword     dummy_int                ; vector 0 (BRK)
        .lword     dummy_int                ; vector 1
        ……
        .lword     dummy_int                ; vector 63
.endif  <------------------------------------------------------  Deletes .endif
```

When using NC308WA

[sect308.inc: near the 428th line]

```
;-----------------------------------------------------------
; variable vector section
;-----------------------------------------------------------
        .section   vector,ROMDATA           ; variable vector table
        .org       VECTOR_ADR
.if  0  <------------------------------------------------------  Deletes .if 0 to enable .lword.
        .lword     dummy_int                ; BRK (software int 0)
        .lword     dummy_int                ;
        ……..
        .lword     dummy_int                ; software int 63
.endif  <------------------------------------------------------  Deletes .if 0 to enable .lword.
```

C. A Guide to Porting Projects Created with TM to High-performance Embedded Workshop V.4

This document explains how to port projects created with TM V.2.xx or V.3.xx into High-performance Embedded Workshop V.4.

C.1. Summary

To port projects created using TM V.2.xx or V.3.xx into High-performance Embedded Workshop V.4, the Import Makefile function of High-performance Embedded Workshop is used. This function can create projects from such items of information as source files and build options described in the specified makefile files.
In TM, project files are created in the makefile format executable in GNU make format. When project files created with TM are selected as makefile files using High-performance Embedded Workshop Import Makefile function, they are converted to files that can run in High-performance Embedded Workshop. In addition to TM project files, the Import Makefile function can also convert files in the makefile formats for hmake, nmake, and gmake to High-performance Embedded Workshop projects.

C.2. Porting Procedure

To port projects created using TM into High-performance Embedded Workshop, perform the following steps:

1. Open the File menu and select the New Workspace command.
2. The New Project Workspace dialog box opens.

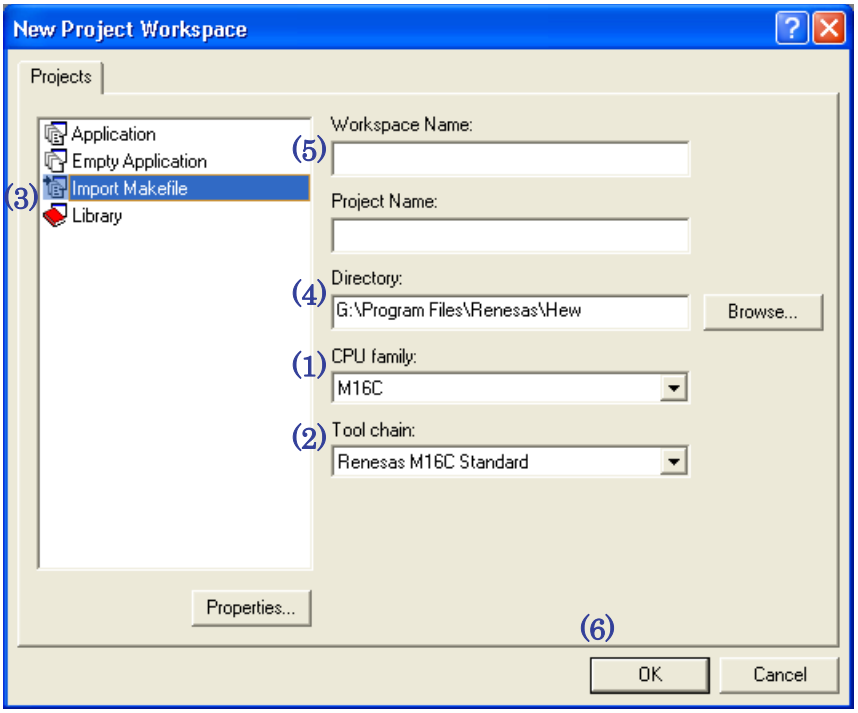


Figure 1 New Project Workspace Dialog Box

- Select the type of CPU used in the TM project from the Type of CPU drop-down list.
- Select the tool chain (cross tool) used for the TM project from the Toolchain drop-down list. The names of tool chains and corresponding cross tools are shown in Table 1.

Table 1 Tool Chains and Corresponding Cross Tools

Tool Chain	Cross Tool
Renesas M16C Standard	NC30WA
Renesas R8C Standard	NC8C
Renesas M32C Standard	NC308WA
Renesas M32R Standard	CC32R

- Select Import Makefile from the Project list.
- Type the directory path in the Directory text box.
- Type the workspace name in the Workspace Name text box. The same name will be automatically entered as the project name in the Project Name text box.
- Click **OK**.

3. You should now be able to see the New Project-1/4-Import Makefile wizard.

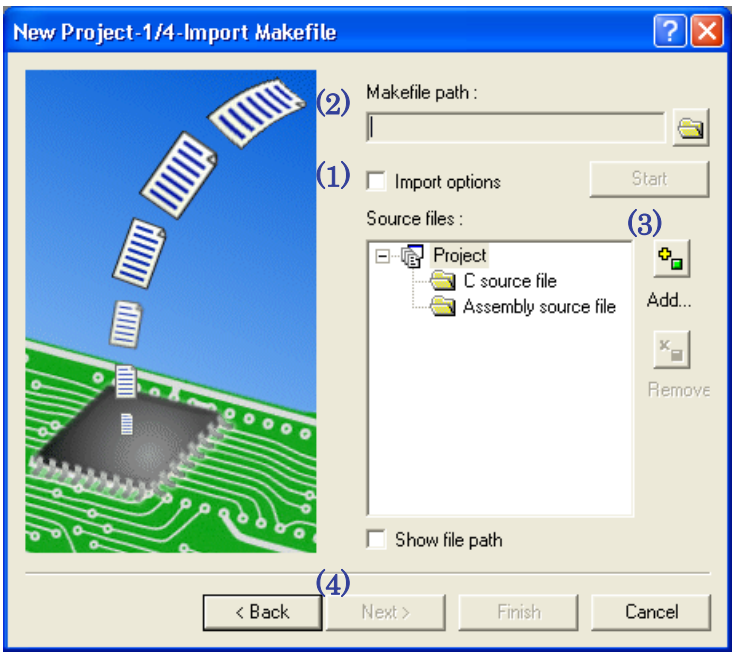


Figure 2 New Project-1/4-Import Makefile Wizard

- Select the Import options check box; this will enable information on build options (compiling and assembling options etc.) to be used to create High-performance Embedded Workshop projects. If you clear the Import options check box, the above information is neglected and not used in High-performance Embedded Workshop.
- Type the name of the TM project file (with extension .tmk) in the Makefile path text box. As soon the name is input, the specified file is analyzed, and upon analysis completion, the analyzed source files are displayed in a tree structure in the Source files box. Click the Start button to analyze the specified file again.
- If there are any errors in the analysis results (tree structure in the Source files box), rectify the tree structure with the Add and Remove buttons.
- Click **Next**.

4. Follow the instructions according to the Wizard as it continues in the procedure.

C.3. Usage Notices

C.3.1. TM-to-High-performance Embedded Workshop Portable and Non-Portable Information

When you port a project created using TM into High-performance Embedded Workshop, not all the components of the project can be ported.

Portable information is as follows:

- Paths of assembler source files
- Paths of C-language source files
- Assembling options
- C-compiling options
- Linking options (except linkage order)

Non-Portable Information:

- Linkage order
- Tool configurations, dependencies, and options other than Assembler, C Compiler, Linker

To transfer these items, edit the High-performance Embedded Workshop project as described in Section 3.4 and further after processing the Import Makefile.

C.3.2. Cross Tools

Import Makefile cannot enable all cross tool versions for use in High-performance Embedded Workshop projects regardless of whether they are used with TM or not; only the following cross tools versions are valid for High-performance Embedded Workshop projects:

NC30WA :	V.5.20 Release1 or later
NC8C :	V.5.30 Release1 or later
NC308WA :	V.5.20 Release1 or later
CC32R :	V.4.20 Release1 or later

C.3.3. High-performance Embedded Workshop Versions

When TM projects are ported into High-performance Embedded Workshop, information portable to High-performance Embedded Workshop varies according to the High-performance Embedded Workshop version. The information that can be ported from each cross tool to various High-performance Embedded Workshop versions are shown in Table 2.

Table 2 Portable Information and Corresponding High-performance Embedded Workshop Versions

		High-performance Embedded Workshop				
		~V.3.01.02	V.3.01.04	V.3.01.05	V.3.01.06	V.4.00
NC30WA	V.5.20 Release1	B	B	B	B	A
	V.5.30 Release1	B	B	B	B	A
NC8C	V.5.30 Release1	B	B	B	B	A
NC308WA	V.5.20 Release1	B	B	B	B	A
CC32R	V.4.20 Release1	B	B	B	B	A
	V.4.20 Release1A	B	B	B	B	A
	V.4.30 Release 00(A)	B	B	B	B	A

A: All the items of information listed in Section 3.1 are portable.
B: Only the paths of assembler and C-language source files are portable.

C.3.4. Generated Project Workspace

Because the project workspace created for a TM project ported to the High-performance Embedded Workshop environment is simply the contents of the makefile itself, its configuration (object output directory) will be different than that of a newly generated project workspace in High-performance Embedded Workshop.
To validate the configuration, modify the output directory file names for the compiler, assembler and linker as follows:

Output Directory (compiler, assembler):	$(CONFIGDIR)
Output Directory (linker):	$(CONFIGDIR)¥$(PROJECTNAME).x30

C.3.5. Load Module Converter

Import Makefile cannot port the information contained in any load module converter (for example, information on options, command executions, or dependencies) into the High-performance Embedded Workshop project. If using a load module converter to create projects in TM, change the settings of the load module converter as follows after completing the Makefile processing:

1. Open the Build menu and select the Build Phases command.
2. The Build Phases dialog box will open.

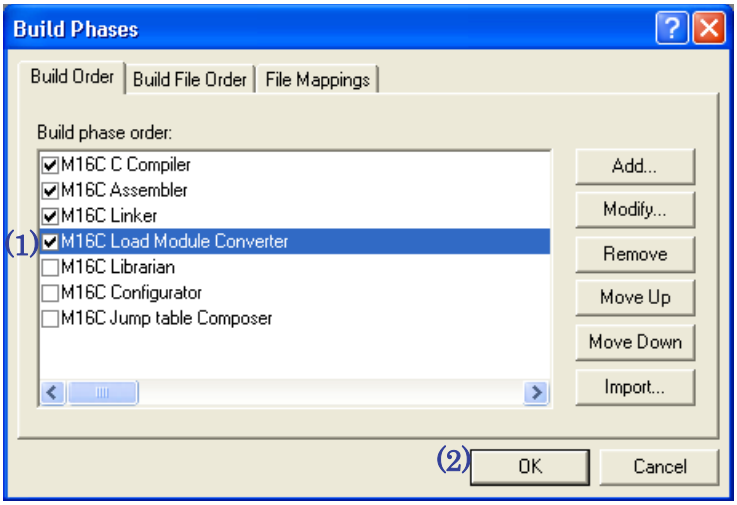


Figure 3 Build Phases Dialog Box

- Select the Mxxx Load Module Converter check box from the Order of Build Phases list.
- Click **OK**.

3. Open the Build menu and select Renesas Mxxx Standard Toolchain.
4. The Renesas Mxxx Standard Toolchain dialog box appears.

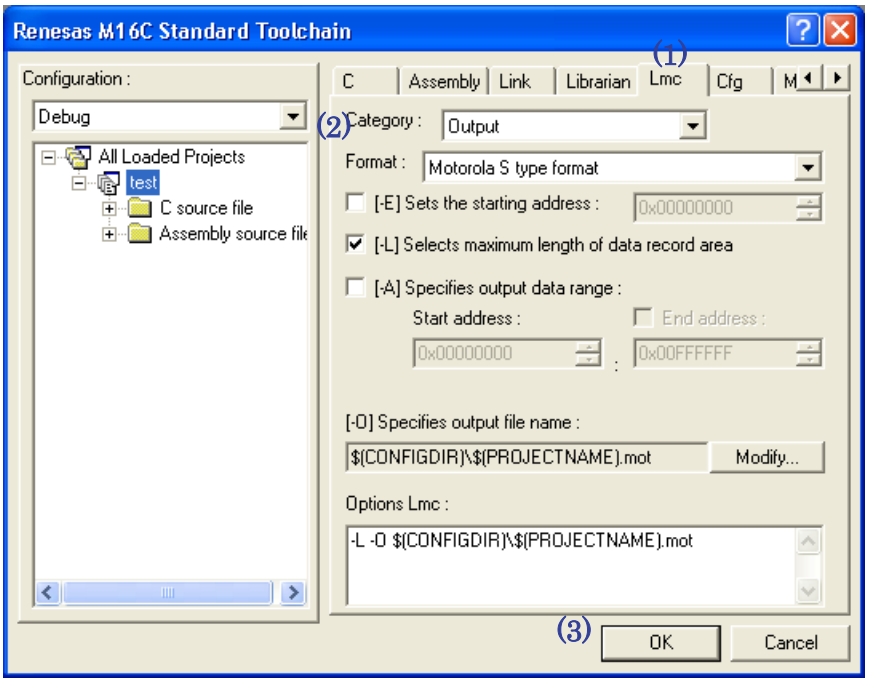


Figure 4 Renesas M16C Standard Toolchain Dialog Box

- Click the Lmc tab.
- Select the Category type from the Category drop-down list.
- Click **OK**.

C.3.6. Other Tools

Import Makefile cannot port any information (options, command executions, dependencies) contained in tools other than the assembler, C compiler, and linker. If any tools other than the assembler, C compiler, linker, and load module converter are used to create projects in TM, custom build phases must be created in High-performance Embedded Workshop. Custom build phases are specifically for operating other tools before, after, or during standard builds (in the assembler, C compiler, and linker).

For more details, see Section 3.2 "Creating Custom Build Phases" in the High-performance Embedded Workshop 4 User's Manual. The following is provided as an example of how to register the cross-reference generation tool xrf30 with High-performance Embedded Workshop.

1. Open the Build menu and select the Build Phases command.
2. The Build Phases dialog box appears; click **Add**.

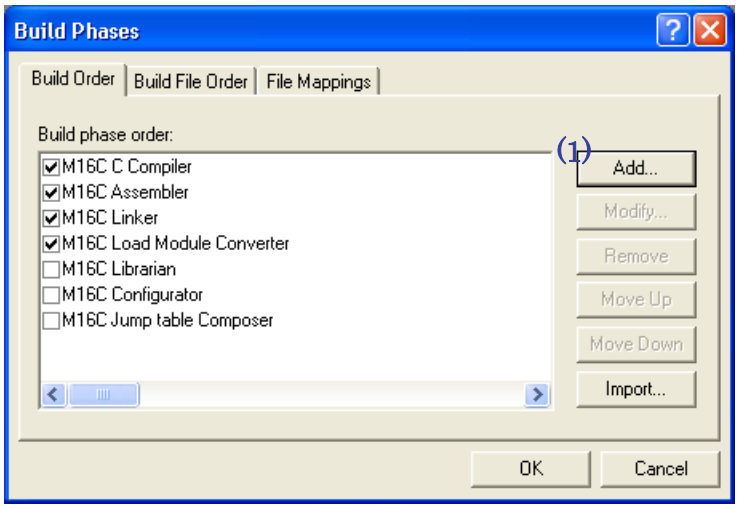


Figure 5 Build Phases Dialog Box

3. The New Build Phase- Step 1/4 wizard opens. Follow the instructions to register the tool as follows:

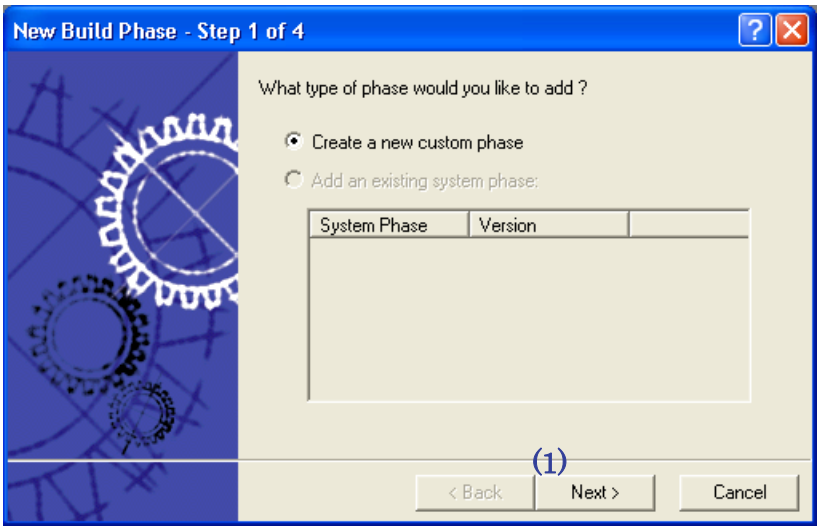


Figure 6 New Build Phase- Step 1/4 Wizard

- Click **Next** (the Create a New Custom Phase check box is selected by default); the New Build Phase-2/4 Step wizard opens.

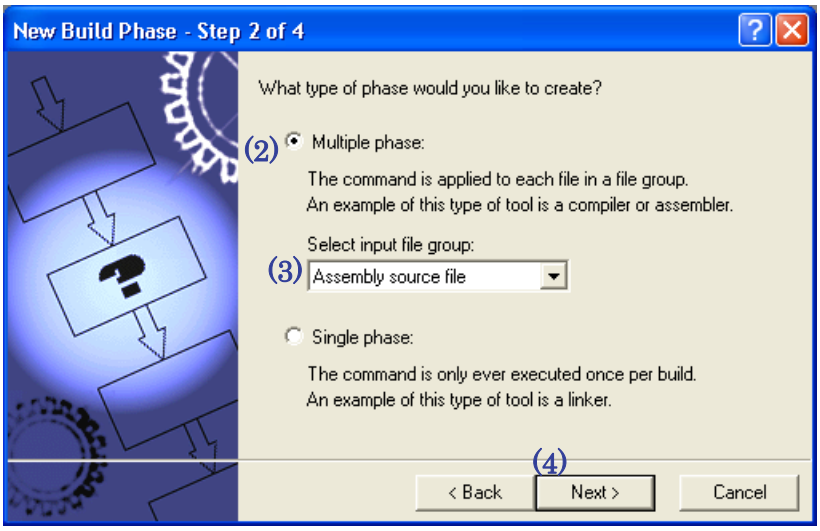


Figure 7 New Build Phase- Step 2/4 Wizard

- In this wizard, select the Multiple Phase check box.
- Select Assembly Source file from the Select input file group.
- Click **Next**; the New Build Phase- Step 3/4 wizard opens.

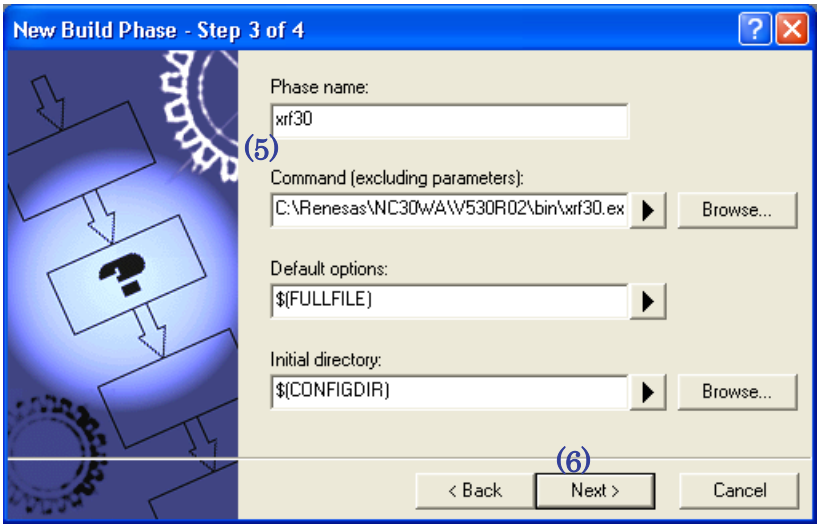


Figure 8 New Build Phase- Step 3/4 Wizard

- Type xrf30 and its fullpath name in the Phase Name and the Command text box.
- Click **Next**; the New Build Phase- Step 4/4 wizard opens.

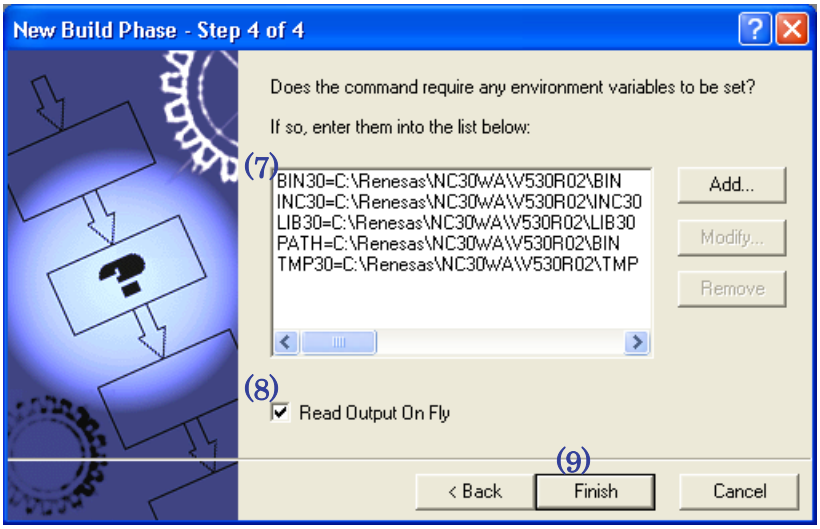


Figure 9 New Build Phase- Step 4/4 Wizard

- In this wizard, enter the necessary environment variables in the list.
- Select "Read Output On Fly" check box.
- Click **Finish**.

4. You return to the Build Phases dialog box at this point, where you can see that xrf30 has been registered as a build phase at the end of the Order of Build phase order.

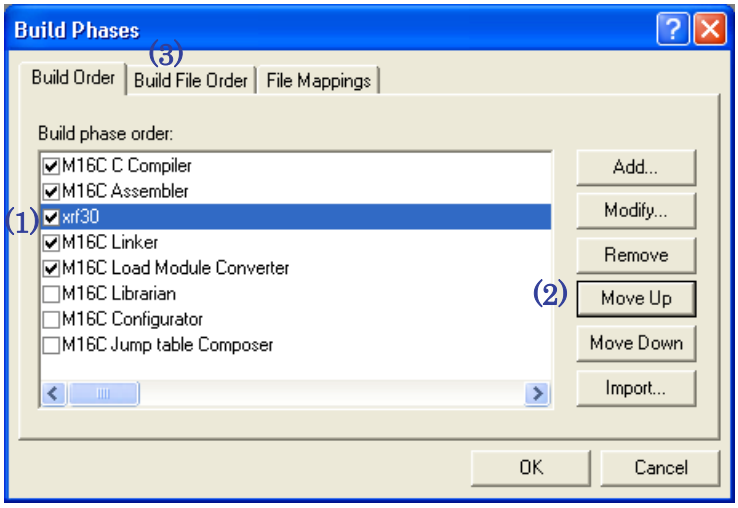


Figure 10 Build Phases Dialog Box (Build Order Tab)

- Select xrf30 from the Order of Build phase order.
- Click **Move Up** to move xrf30 next to the assembler name (see Figure 10).
- Click the Build File Order tab.

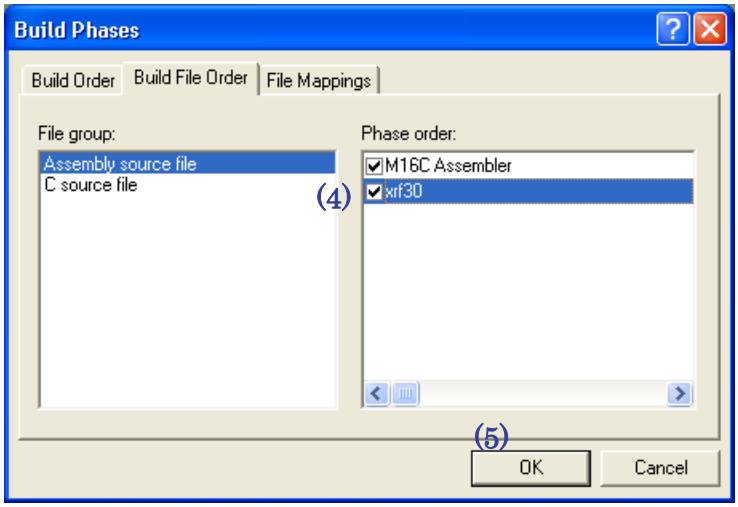


Figure 11 Build Phases Dialog Box (Build File Order Tab)

- Select the xrf30 check box in the Order of Phase order.
- Click **OK**.

5. Open the Options menu and select the xrf30 command.

6. The xrf30 Options dialog box appears; select options as necessary. This setting executes xrf30 for all assembler source files after assemble is completed at a build (before linking files).

C.3.7. Linkage order

Import Makefile cannot port the linking order information to High-performance Embedded Workshop. High-performance Embedded Workshop arranges the linking order alphabetically. To change this order, go through the following steps:

1. Open the Build menu and select the Linkage Order command.
2. The Linkage Order dialog box opens.

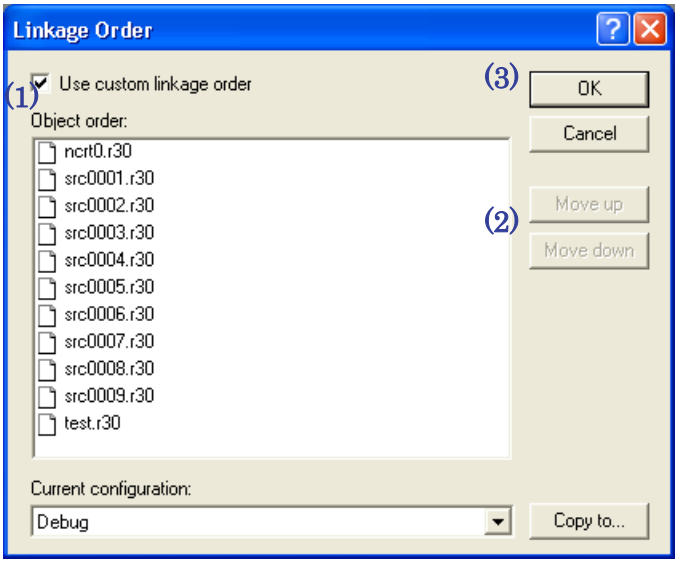


Figure 12 Linkage Order Dialog Box

- Select "Use custom linkage order" check box.
- Select a file from the Object order list, and click **Move up** or **Move down** to move the file. Repeat this step for all files that need to be rearranged.
- Click **OK**.

C.3.8. Placing the Start Up program at the top of Linkage Order

As the Import Makefile cannot port linking order information to High-performance Embedded Workshop, and links are order alphabetically, the start up program may not be placed at the top of the linking order. To place it at the top, follow the steps described previously in Section C.3.7 "Linkage Order."