

Be sure to read this note

## M3T-NC30WA V.5.20 Release 1 Release Note

First Edition

Renesas Solutions Corporation  
Microcomputer Tool Department  
Software Engineering Group 1  
Sep 1 2003

### Abstract

Welcome to M3T-NC30WA(abbreviated as NC30WA) V.5.20 Release 1. This document contains supplementary descriptions to User's Manual. When you read certain items in the User's Manual, please read this document as well.

Also, this document contains a License Agreement in the last. Please read it before using. By using the software, you are accepting and agreeing to such terms.

<b>1.</b>	<b>Contents</b> .....	<b>2</b>
<b>2.</b>	<b>The latest info and FAQs</b> .....	<b>2</b>
<b>3.</b>	<b>Precautions</b> .....	<b>2</b>
3.1.	<b>About this product</b> .....	<b>2</b>
3.2.	<b>About R8C/Tiny Series</b> .....	<b>2</b>
3.3.	<b>About the specification of Entry version</b> .....	<b>2</b>
3.4.	<b>Precaution about the real-time OS MR30</b> .....	<b>3</b>
3.5.	<b>Precaution M16C Family-Dependent Code</b> .....	<b>3</b>
3.6.	<b>Precaution for startup</b> . .....	<b>5</b>
3.7.	<b>Precautions about environment of operation</b> .....	<b>5</b>
3.8.	<b>Suggestions Concerning File Names</b> .....	<b>5</b>
3.9.	<b>Precautions about virus check programs</b> .....	<b>5</b>
3.10.	<b>Precautions regarding the preprocessing directive #define</b> .....	<b>5</b>
3.11.	<b>About specifying the interrupt control register</b> .....	<b>5</b>
3.12.	<b>Precautions to be taken when using #pragma ASM/ENDASM</b> .....	<b>6</b>
3.13.	<b>Regarding M16C/62 4M extended mode</b> .....	<b>6</b>
3.14.	<b>Searching for files to include</b> .....	<b>6</b>
3.15.	<b>Note of Debugging the program using _Bool type</b> .....	<b>7</b>
3.16.	<b>About specifying -fauto_over_255(-fAO2)</b> .....	<b>7</b>
<b>4.</b>	<b>Precautions to be taken when Using HEW</b> .....	<b>7</b>
4.1.	<b>Using the PDxx Debugger</b> .....	<b>7</b>
<b>5.</b>	<b>Installing M3T-NC30WA</b> .....	<b>10</b>
5.1.	<b>Before installing M3T-NC30WA</b> .....	<b>10</b>
5.2.	<b>Precaution about installing this product</b> .....	<b>10</b>
5.3.	<b>M3T-NC30WA Installer</b> .....	<b>10</b>
5.4.	<b>Setting environment after installation</b> .....	<b>10</b>
<b>6.</b>	<b>Operating environments that have been confirmed to be good</b> .....	<b>11</b>
<b>7.</b>	<b>Software version list of NC30WA V.5.20 Release 1</b> .....	<b>11</b>
<b>8.</b>	<b>About Added Options</b> .....	<b>12</b>
8.1.	<b>About the warning message</b> .....	<b>12</b>
8.2.	<b>About #pragma SECTION</b> .....	<b>13</b>
<b>9.</b>	<b>License Agreement</b> .....	<b>14</b>

# 1. Contents

## The product contains

- a) CD-ROM

## the CD-ROM contains

- a) NC30WA V.5.20 Release 1(Entry version)
- b) NC30 user's manual (PDF file)
- c) AS30 user's manual(PDF file)
- d) TM
- e) TM user's manual(PDF file)
- f) NC30WA V.5.20 Release 1, TM Release Note

\*Manual of NC30, AS30, and TM are PDF files only.

# 2. The latest info and FAQs

Please refer to the useful following sites:

Tool News:

<http://www.renesas.com/jpn/products/mpumcu/tools/index.html>

FAQs:

[http://www.renesas.com/jpn/products/mpumcu/toolhp/faq/m16c80/nc30wa/nc30wa\\_j.htm](http://www.renesas.com/jpn/products/mpumcu/toolhp/faq/m16c80/nc30wa/nc30wa_j.htm)

# 3. Precautions

Please observe the following precautions before using NC30WA.

## 3.1. About this product

The Entry Version is **not to be supported and warranted**. Accordingly, for the directions for use of Entry Version or the like, you cannot apply to our company. However, if you inform us of any idea that may flash upon your mind and your comments, if any, we are ready to take up your suggestions and hints for future improvements. However, please take note that we may not give our reply to you as to your idea.

## 3.2. About R8C/Tiny Series

When you need the compiler for R8C/Tiny series, please use the Professional version or R8C/Tiny Free version.

## 3.3. About the specification of Entry version

- Please be careful that Entry version has the following specification.
  - About -fansi option
 

This compiler always compile with "-fansi" option. So If you specify this option, this compiler disregards this option.
  - About describing of a program
 

If you describe the following words, append "underscore(\_)" before the word, please.

**inline**      →      **\_inline**

<b>near</b>	<b>_near</b>
<b>far</b>	<b>_far</b>
<b>asm()</b>	<b>_asm()</b>

- About NC30&AS30 option

You can not use the following options.

C compiler(NC30)	
Debug Option	-genter, -gno_reg
Optimization Option	-O[1-5],-OR,-OS,-Oconst(-OC),-Ono_bit(-ONB),-Ono_stdlib(-ONS), -Ono_break_source_debug(-ONBSD),-Ono_float_const_fold(-ONFCF), -Osp_adjust(-OSA),-Ostack_frame_align(-OSFA),-Oloop_unroll(-OLU), -Ono_asmopt(-ONA) ,-Ocompare_byte_to_word(-OCBTW), -Ono_logical_or_combine(-ONLOC)
Code generated Option	-finfo, -fuse_DIV(-fUD), -fansi, -fnear_ROM(-fNROM), -fsmall_array(-fSA), -fno_align(-fNA)
Assemble & Link Option	-as30, -ln30
Assembler(AS30)	
Option	-finfo, -P, -M

- About software tools and an utility

You can not use the following software tools and an utility.

Software Tools	StkViewer,MapViewer, Assemble optimizer(aopt30), Librarian(lb30),Structured description assembler(pre30), Standard Library source file
Utility	utl30(The SBDATA declaration & SPECIAL page Function declaration Utility)

- About TM ( Tool Manager )

- You can not use the inspector function of TM(Tool Manager).
- You can not create a library project.

### 3.4. Precaution about the real-time OS MR30

Please be sure to describe asm function as \_asm function. So, Please change asm function which is described in MR30's header file (mr30.h) to \_asm. (mr30.h file is in the inc30 subdirectory under the directory where MR30 has been installed.)

### 3.5. Precaution M16C Family-Dependent Code.

You may need to use specific instructions when writing to or reading registers in the SFR area. Because the specific instruction is different for each model, see the User's Manual for the specific Machine. These instructions should be used in your program using the asm function.

#### [Problem]

When -O5 optimizing options is used, the compiler generates in some cases BTSTC or BTSTS bit manipulation instructions. In M16C, the BTSTC and BTSTS bit manipulation instructions are prohibited from rewriting the contents of the interrupt control registers. However, the compiler does not recognize the type of any register, so, should BTSTC or BTSTS instructions be generated for interrupt control

registers, the assembled program will be different from the one you intend to develop. For detailed information about this, see Section 5, "Precautions for Interrupts" Described in Related Documents (Excerpts).

When using any of the products concerned, ensure that no incorrect code is generated.

Note:

In the products other than those concerned, this problem does not occur because BTSTC and BTSTS instructions are not generated if any of the above-mentioned optimizing options is used.

### [Workaround ]

This problem will be circumvented in either of the following ways:

1. Suppress the generation of the BTSTC and BTSTS instructions resulting from using an optimizing option by selecting the -ONA (or -Ono\_asmopt) option together with any of the above-mentioned optimizing options.
2. Add an asm function to disable optimization locally, as shown in the example below.

Example:

```
-----
while(TA0IC.IR==0)
{
    asm(); /* An asm function added to disable
           optimization for TAOIC.IR */
}
-----
```

Make sure that no BTSTC and BTSTS instructions are generated after these side-steppings.

### [Example]

When -O5 optimizing options is used in the program shown below, a BTSTC instruction is generated at compilation, which prevents an interrupt request bit from being processed correctly, resulting in the assembled program performing improper operations.

```
-----
#pragma ADDRESS TA0IC  0055h /* M16C/62 MCU's Timer A0 interrupt control register */
struct {
char  ILVL : 3;
char  IR   : 1; /* An interrupt request bit */
char  dmy  : 4;
} TA0IC;
void wait_until_IR_is_ON(void)
{
while (TA0IC.IR == 0) /* Waits for TA0IC.IR to become 1 */
{
;
}
TA0IC.IR = 0; /* Returns 0 to TA0IC.IR when it becomes 1 */
}
-----
```

== "Precautions for Interrupts" Described in Related Documents (Excerpts) ==

The excerpts of the descriptions on "Precautions for Interrupts" in data sheets and User's Manuals of M16C MCUs are shown below.

#### [Precautions for Interrupts]

When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to

change the register.

Instructions: AND, OR, BCLR, BSET

When accessing registers in the SFR area in C language, make sure that the same correct instructions are generated as done by using asm functions, regardless of the compiler's version and of whether optimizing options are used or not.

### 3.6. Precaution for startup .

For C-language programs to be written into ROM, NC30 comes with a sample startup program written in the assembly language to initial set the hardware (M16C), locate sections, and set up interrupt vector address tables, etc. This startup program needs to be modified to suit the system in which it will be installed. See the manual or data book for your machine for the address of the processor mode register and the bit settings and for your machine for the address of the interrupt vector.

Note, however, that effective from this version, the startup program is separated between the M16C use and the R8C/Tiny use by conditional assemble instructions (.if –, .else – .endif) with a conditional expression “\_\_R8C\_\_!=1.” When you alter the startup program, please alter the program on the .if side.

### 3.7. Precautions about environment of operation

- The MS-Windows version operates under Windows 95, Windows 98, Windows NT 4.0 or later. It does not work under Windows 3.1 and Windows NT 3.5x or earlier.
- If in Windows NT environment the command prompt size is set to other than “80 x 25,” the command prompt size will change frequently as you start the compiler. Make sure the command prompt size is set to “80 x 25.”

### 3.8. Suggestions Concerning File Names

The file names that can be specified are subject to the following restrictions:

- Directory and file names that contain kanji cannot be used.
- Only one period (.) can be used in a file name.
- Network path names cannot be used. Assign the path to a drive name.
- Keyboard shortcuts cannot be used.
- Directory and file names that contain a space character cannot be used.
- The “...” symbol cannot be used as a means of specifying two or more directories.
- A file name in length of 128 characters or more including path specification cannot be used.

### 3.9. Precautions about virus check programs

If the virus check program is memory-resident in your computer, NC30WA may not start up normally. In such a case, remove the virus check program from memory before you start M3T-NC30WA.

### 3.10. Precautions regarding the preprocessing directive #define

To define a macro which will be made the same value as the macro ULONG\_MAX, always be sure to add the prefix UL.

### 3.11. About specifying the interrupt control register

NC30WA supports the function that set or change the value of an interrupt priority level to conform to MESC TECHNICAL NEWS(No. M16C-14-9805).

- case of specifying the value

Please use SetLevel function. In this time, please be sure to include “intlevel.h” file.

```
SetLevel(char *adr, char val)
```

adr : Address of the interrupt control register

val : Specifying the value

- case of changing the value

Please use ChgLevel function. In this time, please be sure to include “intlevel.h” file.

```
ChgLevel(char *adr, char val)
```

adr : Address of the interrupt control register

val : Specifying the value

#### [Example]

```
#include <intlevel.h>
#pragma ADDRESS timerA 55H
char *timerA;
void func(void)
{
    SetLevel(timerA,2); // Specifying the value “2” to the interrupt priority level
    :
    ChgLevel(timer A,4); // Changing the value “4” to the interrupt priority level
}
```

### 3.12. Precautions to be taken when using #pragma ASM/ENDASM

- Regarding debug information when using #pragma ASM outside functions, if you write #pragma ASM anywhere outside functions, no C source line information will be output. for this reason, information regarding descriptions in #pragma ASM to #pragma ENDASM, such as error message lines when assembling or linking and line information when debugging, may not be output normally.
- Do not write branches (including conditional branches) in the register or variable areas, that would allow the asm functions to affect program flow during program flow analysis and processing by the compiler.
- C compilers generate code of arguments to be passed via registers and of register variables by analyzing their scopes. However, if manipulations of register values are described using inline assemble functions (such as #pragma ASM/#pragma ENDASM directives and asm function), C compilers cannot hold information on the scopes of the above-mentioned arguments and register variables. So, be sure to save and recover register contents on and from the stack when registers are loaded using inline assemble functions

### 3.13. Regarding M16C/62 4M extended mode

Make sure the program is located in the internal ROM.

### 3.14. Searching for files to include

If you give a file to include together with a drive name in the #include line, and attempt to compile the file from a directory different from the one in which the file to compile is present, instances may occur in which the file to include cannot be searched.

#### [Example]

```
#include "c:\user\test\sample.h"
main(){}
```

```
C:\user>nc30 \user\test2\sample.c -silent
\user2\tm_test\aa.c
```

```
[Error(cpp30.21):\user2\test2\sample.c, line 1] include file not found 'c:\user\test\sample.h'
```

### 3.15. Note of Debugging the program using `_Bool` type

When you debug the program which uses the `_Bool` type, please confirm whether the debugger is supporting the `_Bool` type.

In using the debugger which is not supporting the `_Bool` type, please use a debugging option

- “-gbool\_to\_char (-gBTC)” at the time of compile.

### 3.16. About specifying `-fauto_over_255(-fAO2)`

Specifying `-fauto_over_255(-fAO2)` option results in incorrect code being generated.

#### [Condition]

1. The optimizing options `-OS` is selected.
2. Auto variables is not used.
3. The size of parameter is more than 128 bytes.
4. Immediately before the ending process of a function A, another function B is called.

#### [Example]

```
void sub(void);
struct x {
    char    mm[200];
};

void func(struct x m)
{
    cc1 = m.mm[0];
    sub();
}
```

#### [Workaround]

Place a dummy asm function immediately after calling a.

```
void func(struct x m)
{
    cc1 = m.mm[0];
    sub();
    asm();
}
```

## 4. Precautions to be taken when Using HEW

### 4.1. Using the PDxx Debugger

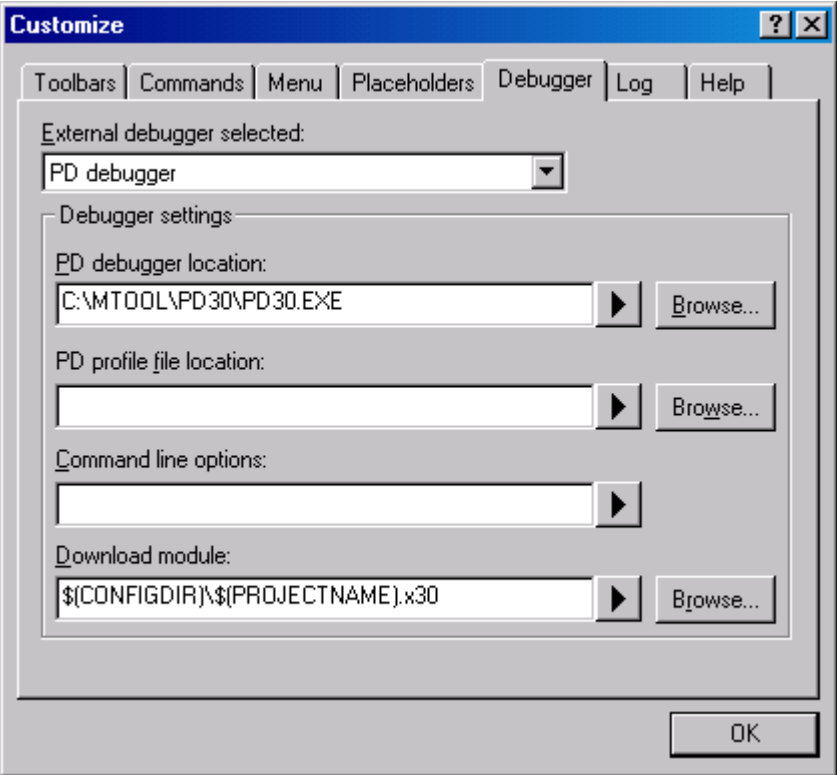
The HEW can launch the PDxx debugger. If you want to use the PDxx debugger then you must add it to the “Tools” menu.

The “Debugger” tab of the “Customize” dialog is where the HEW related information is configured.

How to launch the PDxx debugger is following.

- (1) Select “PD debugger” for the [External debugger selected].
- (2) Specify the location of the executable file PDxx.exe for the [PD debugger location].
- (3) Input “-LOAD=\$(CONFIGDIR)\\$(PROJECTNAME).x30” to the [Download module].

The following is an example for the setting.



The following debuggers are available to use with HEW.

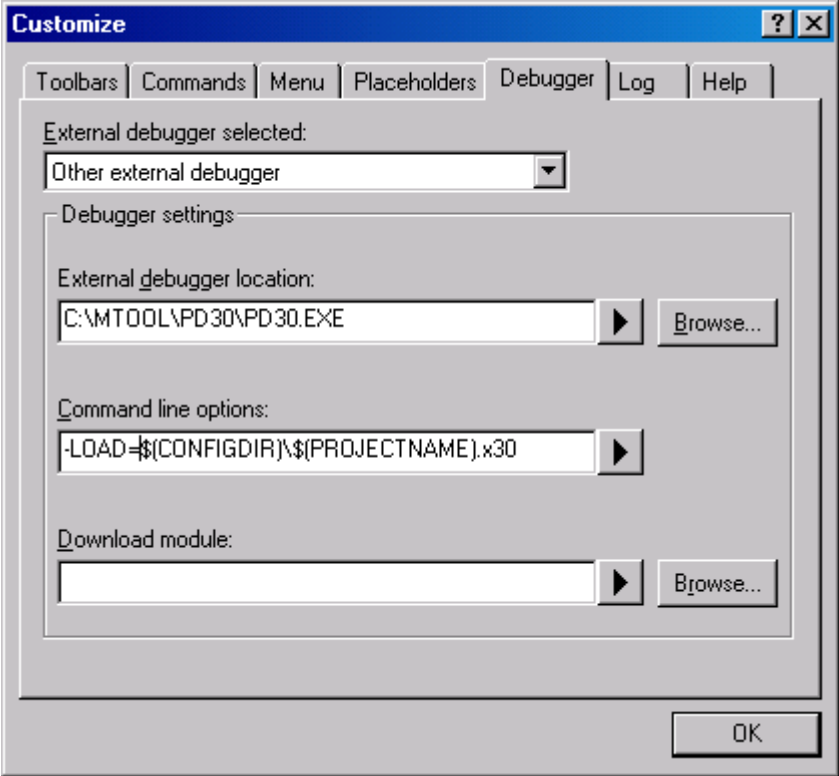
PD30	V.8.10 or later
PD30F	V.2.10 or later
PD308	V.5.10 or later
PD308F	V.3.10 or later

How to register a debugger of which version is former than the above debuggers is following.

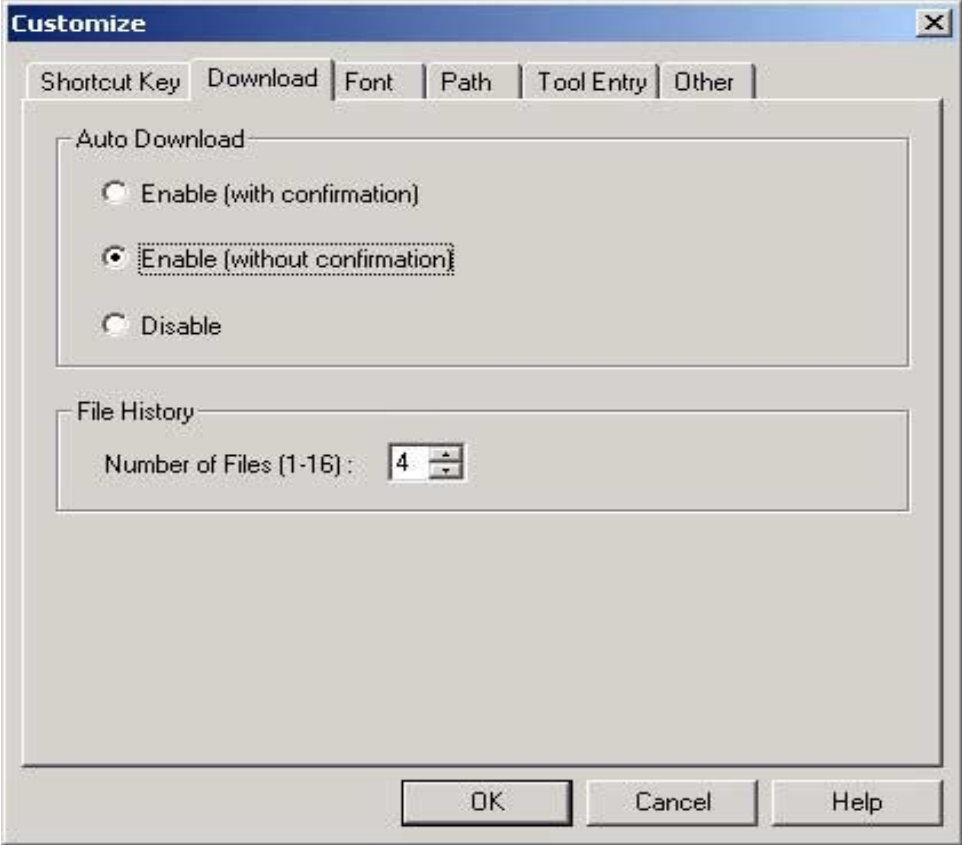
- (1) Select "other external debugger" for the [External debugger selected].
- (2) Specify the location of the executable file PDxx.exe for the [External debugger location].
- (3) Input "-LOAD=\$(CONFIGDIR)\\$(WORKSPNAME).x30" to the [Command line options].



The following is an example for the setting.



If you check the “Enable” for Auto Download in the Download tab of the **PDxx’s** Customize dialog box, the current object module will be downloaded automatically when it was modified after build.(Refer to PDxx user’s manual for the detail.)



## 5. Installing M3T-NC30WA

### 5.1. Before installing M3T-NC30WA

Please confirm as follows before installing M3T-NC30WA in your computer.

- Please carefully read the "License Agreement" and "Release Note" included with your product before using M3T-NC30WA. If you've installed this product in your computer, it is assumed that you've agreed to the provisions stipulated in the License Agreement.
- Use the dedicated installer to install M3T-NC30WA.

### 5.2. Precaution about installing this product

- When you install this product, this installer is sure to install HEW (High-performance Embedded Workshop), too.
- If HEW or NC30WA is already installed, be sure to uninstall it beforehand.
- If the dialog described the following message is output during installing, please select "Yes to All".

Dialog Message:

The following file is already on your computer.

C:\WIN98\TEMP\NC30WA\...

Do you wish to overwrite this file?

:

- You can not specify a name that contains a blank for the directory in which AS30 will be installed. For other points to note, see "Suggestions Concerning File Names" in this note.
- When changing the item "Installation directory" from the "Select Components" screen by using "Browse..." during installation, a trouble sometimes occurs that you cannot select some of the drives displayed by "Drives:" on the "Chose Directory" screen. In such a case, return to the "Select Components" screen and use "Disk Space..." on that screen to specify a drive. If you still have difficulty, reexecute the installer after restarting Windows 95.

### 5.3. M3T-NC30WA Installer

The installer is provided for each of the environments (supported host, supported OS and language) listed below. Check the product you've purchased to find the appropriate installer.

Supported host	Supported OS	Installer name	Directory on CD-ROM
IBM PC/AT compatible	Microsoft Windows 95	SETUP.EXE	\NC30WA\W95E
	Microsoft Windows 98		
	Microsoft Windows Me		
	Microsoft Windows NT		
	Microsoft Windows 2000		
	Microsoft Windows XP		

### 5.4. Setting environment after installation

After you finished installing M3T-NC30WA, set environment variables next.

The environment variables marked by "Auto" in the tables below do not need to be set because the Windows installer automatically rewrites AUTOEXEC.BAT.

Environment variable	Example of setting
BIN30	Auto (SET BIN30=C:\MTOOL\BIN)
INC30	Auto (SET INC30=C:\MTOOL\INC30)
LIB30	Auto (SET LIB30=C:\MTOOL\LIB30)
TMP30	Auto (SET TMP30=C:\MTOOL\TMP)
NCKIN	SET NCKIN=SJIS
NCKOUT	SET NCKOUT=SJIS
Command path	Auto (C:\MTOOL\BIN is added)

## 6. Operating environments that have been confirmed to be good

The table below lists the host computers and OS versions with which NC30WA has been confirmed to be operating normally.

Host computer	OS version	CD-ROM directory
IBM <sup>1</sup> PC/AT	Microsoft <sup>2</sup> Windows 95	W95E
	Microsoft Windows 98	
	Microsoft Windows NT 4.0	
	Microsoft Windows Me	
	Microsoft Windows XP	
	Microsoft Windows 2000	

Whether NC30WA operates normally with other host computers and OSs depends on the manufactures of those host computers and OSs. Therefore, please contact the manufactures of your host computer and OS to see if the software will work properly with your computer system.

## 7. Software version list of NC30WA V.5.20 Release 1

The following lists the software items their versions included with NC30WA V.5.20 Release 1

nc30	V.5.10.03 –Entry	Compile driver
cpp30	V.4.04.00	Preprocessor
ccom30	V.5.20.03	Compiler
as30	V.5.00.00 –Entry	Assemble driver
mac30	V.3.40.00	Macro processor
asp30	V.5.00.00 -Entry	Assemble Processor
ln30	V.5.00.00	Linkage Editor
lmc30	V.4.00.00	Load Module Converter
xrf30	V.2.01.00	Cross Referencer
abs30	V.2.11.00	Absolute listor

<sup>1</sup> IBM and AT are registered trademarks of International Business Machines Corporation.

<sup>2</sup> Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation in the U.S.and othercountries.

## 8. About Added Options

**-fauto over 255**

**-fAO2**

Changes generated code

**Function:**

- Changes the stack frame size per function that can be reserved to 64 Kbytes.

**Note:**

- This option cannot be used in combination with #pragma SBDATA. If a file that contains a description of #pragma SBDATA is compiled, the warning shown below is output, with the description of #pragma SBDATA ignored.

[Warning(ccom):XX.c,line XX] compile option -fauto\_over\_255 is specified,  
#pragma SBDATA was ignored.

==> #pragma SBDATA xxx;

- Specify this option for the files described below.
  - a. When a function exists that requires a stack frame of 255 bytes or more (hereafter referred to as function A)  
==> Files in which function A is written
  - b. When an interrupt occurs while processing function A (hereafter referred to as interrupt A) and a variable declared by #pragma SBDATA is accessed from interrupt A  
==> Files in which interrupt A is written

### 8.1. About the warning message

Warning Messages	Description and countermeasure
no const in previous declaration.	The function or variable declaration without const qualification is const-qualified on the entity definition side. ==> Make sure the function or variable declaration and the const qualification on the entity definition side are matched.

## 8.2. About #pragma SECTION

### #pragma SECTION

Change section name

**Function:** Changes the names of sections generated by NC30

**Syntax:** #pragma SECTION  $\Delta$ section name  $\Delta$ new section name

**Description:** Specifying the program section, data section and rom section in a #pragma SECTION declaration changes the section names of all subsequent functions.

Specifying a bss section in a #pragma SECTION declaration changes the names of all data sections defined in that file.

If you need to add or change section names after using this function to change section names, change initialization, etc., in the startup program for the respective sections.

- You can specify “#pragma SECTION data” and “#pragma section program” two or more times in one file.

**Example:** **[C source program]**  
 #pragma SECTION program pro1 /\* Changes name of program section to pro1 \*/  
 void func( void );

:  
 (remainder omitted)

**[Assembly language source program]**

```

;### FUNCTION func
    .section pro1          /* Maps to pro1 section */
    ._file 'smp.c'
    ._line 9
    .glob _func
_func:
  
```

```

[Change name of data section from data to data1 ]
#pragma SECTION data data1
int i;          /* Maps to data1_NE section */
func()
{
  (remainder omitted)
  
```

```

#pragma SECTION data data2
int j;          /* Maps to data2_NE section */
sub()
{
  (remainder omitted)
  
```

Figure B.65 Example Use of #pragma SECTION Declaration

**Supplement:** When modifying the name of a section, note that the section's location attribute (e.g., NE or \_NEI) is added after the section name.

## 9. License Agreement

### IMPORTANT

DO NOT INSTALL THE SOFTWARE UNTIL YOU HAVE READ AND AGREED TO THE TERMS AND CONDITIONS SPECIFIED IN THE SOFTWARE USER LICENSE AGREEMENT BELOW

Permission to use this software is subject to your agreeing to the terms and conditions specified in the agreement entitled "Software User License Agreement" below. Please read carefully the Software User License Agreement before you open the package.

Product Name : M3T-NC30WA

### Software User License Agreement

The customer (hereafter referred to as "Licensee") and Renesas Solutions Corporation (hereafter referred to as "Licensor") do hereby agree to the terms and conditions as specified in this Software User License Agreement (hereafter referred to as "Agreement") concerning the enclosed this software and its explanatory manuals.

#### ARTICLE 1. Definition

- 1.1 "Licensed Software" shall mean the main unit of the C compiler, Assembler and related executable programs, the "Library" as defined in Section 1.3 below and documentation such as explanatory manuals for the Licensed Software and all other materials provided to Licensee in connection with use of the Licensed Software.
- 1.2 "Designated System" shall mean the single computer system to be installed licensed software.
- 1.3 "Library" shall mean a group of function programs that enable operations including, but not limited to input/output and character recognition and that are offered in the forms of a re-locatable program and a source program.
- 1.4 "Developed Software" shall mean software created by Licensee incorporating all or part of a Library (including a Library that have been modified or adapted by Licensee) in accordance with the licenses granted to Licensee under the terms and conditions of this Agreement.

#### ARTICLE 2. LICENSES

- 2.1 Licensor grants Licensee a personal, non-exclusive, non-transferable license:
  - (a) to use the Licensed Software solely on the Designated System for Licensee's business purpose of creating and copying Developed Software to be incorporated in the Licensee's products installed with Licensor's RENESAS microcomputer (hereafter referred to as "Licensee's Products");
  - (b) to make only one copy by installer program of Licensed Software on the single unit of Designated System for the purposes of using the Licensed Software in accordance with subparagraph (a) above of this Section 2.1 to develop Developed Software in accordance with subparagraphs (c) and (d) below of this Section 2.1;
  - (c) to merge the Library (including Library modified in accordance with subparagraph (c) above in this Section 2,1) with one or more other programs to develop Developed Software, provided any portion of the Library so merged continues to be subject to the terms and conditions of this Agreement; and

#### ARTICLE 3. Restrictions

- 3.1 Licensee shall not transfer the rights granted herein or allow any third party to use the Licensed Software. Licensee agrees not to reverse compile or reverse assemble all or any portion of the Licensed Software.
- 3.2 Licensee shall not remove the copyright notice contained in the Licensed Software.
- 3.3 Licensee shall not sublicense, rent, lease, or assign the Licensed Software.
- 3.4 Licensee shall not reverse assemble or reverse compile the Licensed Software.

#### **ARTICLE 4. Title to Licensed Software**

- 4.1 All right, title and interest in and to the Licensed Software are shall remain with Licensor. The copyrights in the Licensed Software owned by Licensor are protected by Japanese copyright laws, other applicable copyright laws and international treaty provisions. Licensor retains all rights not expressly granted.
- 4.2 The copyrights in the Developed Software as a derivative work of Library shall be owned by Licensee, provided, however, that copyrights in any part of Library contained in such Developed Software shall remain the property of Licensor.

#### **ARTICLE 5. CONFIDENTIALITY**

Licensee acknowledges that the Licensed Software contains valuable proprietary information and trade secrets of Licensor. Licensee shall preserve the Licensed Software in confidence, and shall not provide or disclose the contents of all or any part of the Licensed Software to any third party without written authorization from Licensor. Licensee will take appropriate action, by instruction, agreement or otherwise, with any persons permitted access to the Licensed Software so as to enable them to hold the Licensed Software in confidence and otherwise to satisfy their obligations under this Agreement.

This Agreement shall not restrict disclosure or use of any information which:

- (a) can be demonstrated by Licensee to be in the public domain when received, or thereafter enters the public domain through no fault of Licensee;
- (b) is known to Licensee at the time of the disclosure by Licensor;
- (c) is received from a third party without any restrictions on disclosure and use; or
- (d) is independently developed by Licensee through persons who have not had access to the Licensed Software.

#### **ARTICLE 6. Term and Termination**

- 6.1 This Agreement shall become effective upon Licensee's breaking the packaging seal of the Licensed Software and shall remain in force until terminated by any of the provisions in Sections 6.2 and 6.3 below.
- 6.2 Licensee may terminate this Agreement at any time on thirty (30) day prior written notice to Licensor.
- 6.3 In the event that Licensee shall be in default under this Agreement, Licensor may terminate this Agreement and all licenses granted hereunder by giving written notice of termination to Licensee, effective upon its sending.

#### **ARTICLE 7. POST-TERMINATION PROCEDURES**

- 7.1 In the event of any termination of this Agreement, Licensee shall cease all use of the Licensed Software and shall remove any and all copies of the Licensed Software from the Designated Site, shall remove any Libraries contained in the Developed Software from such Developed Software (except for Libraries incorporated in the Licensee's Products already sold), and shall destroy all such copies of Licensed Software.

- 7.2 Licensee shall complete all the obligations described in Section 7.1 above and shall submit written proof to that effect to Licensor within thirty (30) days from the termination of this Agreement.

## **ARTICLE 8. PROPRIETARY RIGHTS INDEMNIFICATION**

- 8.1 Licensor shall defend or, at its option, settle any claim or proceedings brought against Licensee to the extent that it is based on an assertion that the Licensed Software, used within the scope of this Agreement, constitute direct infringement of any copyright, patent or trademark of a third party and shall indemnify Licensee against all costs, damages and expenses finally awarded against Licensee which result from any such claim, provided that Licensee notifies Licensor promptly in writing of such claim or proceedings and gives Licensor full and complete authority, information and assistance to defend such claim or proceedings, and further provided that Licensee gives Licensor sole control of the defense of any such claim or proceedings and all negotiations for its compromise or settlement.
- 8.2 Should the Licensed Software become, or, in Licensor's opinion, be likely to become the subject of a claim of infringement, Licensor may procure for the Licensee the right to continue using the Licensed Software or replace or modify it to make it noninfringing. If neither of the foregoing alternatives is reasonably available to Licensor, then Licensor's sole and exclusive remedy shall be to terminate this Agreement, and to return to Licensee refund of Licensefees and other expenses paid to Licensor.
- 8.3 Licensor's obligations under this Article 8 shall not apply to any claims of infringement based upon (i) use of other than the latest upgrades of the Licensed Software made available to Licensee by Licensor according to Section 9.2 hereunder if such infringement would have been avoided by the use of such upgrades of the Licensed Software, (ii) combination, operation or use of the Licensed Software with any programs, data or products not furnished by Licensor if such infringement would not have occurred without such combination, operation or use, or (iii) use of the Licensed Software after receiving written notice from Licensor that the Licensed Software infringes any patent, copyright or trademark of a third party. Licensor's obligation under this Article 8 constitute Licensee's sole and exclusive remedy for a claim or proceedings for and intellectual property infringement of a third party.

## **ARTICLE 9. LIMITED WARRANTY, INDEMNIFICATION AND LIMITATION OF LIABILITY**

- 9.1 Licensor warrants that it has the right to grant to Licensee a license to use the Licensed Software and that Licensor has the right to enter into this Agreement.

EXCEPT AS SPECIFICALLY PROVIDED HEREIN, LICENSOR DISCLAIM ALL WARRANTIES, REPRESENTATION, PROMISE OR GUARANTEE, EITHER EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL LICENSOR BE LIABLE FOR ANT INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND INCLUDING WITHOUT LIMITATION, DAMAGES RESULTING FROM LOSS OF USE, LOSS OF DATA, LOSS OF PROFITS OR LOSS OF BUSINESS. THE LICENSED SOFTWARE IS PROVIDED "AS IS", WITH ALL FAULTS.

Licensee agrees that Licensor's liability hereunder for damages, regardless of the form of action, shall not exceed the charges paid by Licensee for the Licensed Software. Notwithstanding the foregoing, in the event Licensee find what it believes to be an error or failure of the Licensed Software, Licensor shall make reasonable efforts to correct any such error or failure by taking measures which Licensor deems appropriate provided that Licensee has returned the Customer Registration Card to Licensor.

However, Licensor does not guarantee that any measures described above will correct all of the errors or failures.

- 9.2 The warranty coverage under this Agreement does not apply to any error or failure of the Developed Software or any error or failure caused by a modification of the Licensed



Software by Licensee or by any negligent act or omission of Licensee.

**ARTICLE 10. GENERAL**

10.1 This Agreement constitutes the entire agreement between the parties concerning Licensee's use of the Licensed Software and no waiver, alteration, modification or cancellation shall be binding unless made in writing and signed by the parties.

10.2 This Agreement is made under and shall be construed in accordance with the Laws of Japan. Licensee agrees that any disputes related to this Agreement shall be settled by the jurisdiction of the court determined by Licensor.

LICENSEE ACKNOWLEDGES THAT LICENSEE HAS READ THIS AGREEMENT AND AGREES TO ALL TERMS AND CONDITIONS STATED HEREIN.