

## Contents

Chapter 1. Changes .....	2
1.1 Support for double-precision floating-point processing instructions .....	2
1.2 Support for the register bank save function .....	2
1.3 Addition of checking source code across multiple files against MISRA-C:2012 rules [Professional edition] .....	2
1.4 Extensions to the checking of source code against MISRA-C:2012 rules [Professional edition] .....	3
1.5 Addition of the -truncated_address_initializer compiler option.....	3
1.6 Addition of a feature for changing a section name when a library file is input .....	4
1.7 Enhancement of optimization .....	4
1.8 Rectified points for caution .....	7
1.9 Other improvement.....	7
Chapter 2. Points for Caution .....	8
2.1 Note on a case of the W0523041 message [C/C++ compiler].....	8
2.2 Note on using MVTC or POPC instructions [Assembler] .....	8
2.3 Note on the delete option for linkage [Optimizing linkage editor].....	8
Chapter 3. Restrictions .....	9
3.1 Restriction on usage of math.h functions (frexp, ldexp, scalbn and remquo) in C++ language (including EC++).....	9
3.2 Restriction of PIC/PID function (pic and pid options) .....	11
3.3 Eliminated options (for the C/C++ compiler) .....	11
3.4 Notes on C/C++ source-level debugging (for the C/C++ compiler) .....	11
3.5 Note on using sections that include the address 0xffffffff (in assembler).....	12
3.6 Note on using -form and -output at the same time (in the linkage editor).....	12
3.7 Note on using function names that begin with _builtin (for the C/C++ compiler) .....	12
3.8 Restriction of -merge_files.....	12
3.9 Restriction of -cfi_ignore_module.....	13
3.10 Restriction on using fenv.h when -dpfpu is specified .....	13
Chapter 4. Standard Libraries Included.....	14
4.1 Library files .....	14
4.2 Using the library files .....	15

## Chapter 1. Changes

This section describes changes on CC-RX from V3.00.00 to V3.01.00. The features indicated as **[Professional edition]** can only be used if the compiler is registered under the professional license.

### 1.1 Support for double-precision floating-point processing instructions

Code that takes advantage of the double-precision floating-point processing instructions that have been included in the RXv3 instruction set architecture is generated for devices with that core. This improves the performance and code size of programs in which the double and long double types are used.

When generating code, the new `-dpfpu` option must be specified as compiler and assembler options.

### 1.2 Support for the register bank save function

The register bank save function that is included in the RXv3 instruction set architecture is now usable.

Writing `bank=<bank number>` with `#pragma interrupt` causes the saving of register values at the start of the interrupt handler and restoring them at the end of the interrupt handler, both at high speed. To take advantage of this feature, `-bank` must be specified as an assembler option.

### 1.3 Addition of checking source code across multiple files against MISRA-C:2012 rules **[Professional edition]**

The `-misra_intermodule` option has been added to check source code across multiple files against MISRA-C:2012 rules.

Although source code had previously only been checked within the individual files, specifying this option now enables the checking of source code across multiple files.

### 1.4 Extensions to the checking of source code against MISRA-C:2012 rules [Professional edition]

The following rule numbers have been added as arguments of the -misra2012 option for checking source code against MISRA-C:2012 rules.

Required rules: **8.5** and **8.6**

The following shows the number of MISRA-C:2012 rules which can be checked by each revision.

Classification of Rules: Number of Rules	V1.07.00	V1.08.00
Mandatory rules: 16	7	<b>7</b>
Required rules: 108	86	<b>88</b>
Advisory rules: 32	26	<b>26</b>
Total: 156	119	<b>121</b>

### 1.5 Addition of the -truncated\_address\_initializer compiler option

The -truncated\_address\_initializer compiler option has been added.

With this option specified, the E0520069 error code is not output in response to code for using addresses to initialize the values of 1- or 2-byte external variables or static variables. Instead, the W0520069 warning message is output.

### 1.6 Addition of a feature for changing a section name when a library file is input

The `-lib_rename` option has been added as an optimizing linkage editor (`rlink`) option. With this option, section names or symbol names in a file within a library that is input at the time of linkage can be changed and linked.

### 1.7 Enhancement of optimization

In CC-RX V3.01.00, optimization has been further enhanced on the following points.

(a) Enhanced use of the MAX, MIN, and BFMOV instructions

<An example of source program>

```
int func(int x) {
    if (x >= 100) {
        x = 100;
    }
    if (x <= -100) {
        x = -100;
    }
    if (x <= 0) {
        return -x;
    }
    return x;
}
```

```
<V3.00.00>
_func:
    MIN #64H, R1
    CMP #0FFFFFF9DH, R1
    BGE L12
L11:
    MOV.L #0FFFFFF9CH, R1
L12:
    ABS R1
    RTS
```

```
<V3.01.00>
_func:
    MIN #64H, R1
    MAX #0FFFFFF9CH, R1
L11:
    ABS R1
    RTS
```

## (b) Improvement of instruction scheduling

```
<An example of source program>
int a[1024], b[1024], c[1024];
void func(int n) {
    int i;
    for (i = 0; i < 4; ++i) {
        a[i] = b[i] + c[i];
    }
}
```

```
<V3.00.00>
_func:
    MOV.L #00000002H, R5
    MOV.L #_c, R14
    MOV.L #_b, R15
    MOV.L #_a, R1
L11:
    MOV.L [R14+], R4
    MOV.L [R15+], R3
    ADD R3, R4
    MOV.L R4, [R1]
    MOV.L [R14+], R2
    MOV.L [R15+], R4
    ADD R4, R2
    MOV.L R2, 04H[R1]
    ADD #08H, R1
    SUB #01H, R5
    BNE L11
L12:
    RTS
```

```
<V3.01.00>
_func:
    MOV.L #00000002H, R5
    MOV.L #_c, R14
    MOV.L #_b, R15
    MOV.L #_a, R1
L11:
    MOV.L [R14+], R4
    MOV.L [R15+], R3
    MOV.L [R14+], R2
    ADD R3, R4
    MOV.L [R15+], R3
    MOV.L R4, [R1]
    ADD R3, R2
    MOV.L R2, 04H[R1]
    ADD #08H, R1
    SUB #01H, R5
    BNE L11
L12:
    RTS
```

(c) Enhanced processing for deleting redundant comparison instructions

```
<An example of source program>
void func2(void);

void func1(unsigned int a, unsigned int b, unsigned int c, unsigned int d){
  if (((a != 0) && (a > b)) && (c == d)){
    func2();
  }
}
```

```
<V3.00.00>
_func1:
    CMP R2, R1
    BLEU L14
L11:
    CMP #00H, R1
    BEQ L14
L12:
    CMP R4, R3
    BNE L14
L13:
    BRA func2_
L14:
    RTS
```

```
<V3.01.00>
_func1:
    CMP R4, R3
    BNE L13
L11:
    CMP R2, R1
    BLEU L13
L12:
    BRA func2_
L13:
    RTS
```

### 1.8 Rectified points for caution

The following points for caution no longer apply. For details, refer to Tool News.

- Point for caution when the -misra2012 option is specified. (CCR#050)

### 1.9 Other improvement

Other major improvement is described below.

- Correction of internal errors

Internal errors sometimes occurred in the build process in previous versions. These errors have been corrected.

## Chapter 2. Points for Caution

This section describes points for caution regarding CC-RX.

### 2.1 Note on a case of the W0523041 message [C/C++ compiler]

When the `int_to_short` option is specified and a file including a C standard header is compiled as C++ or EC++, the compiler may show the W0523041 message. In this case, simply ignore the message because there are no problems.

[NOTE]

In compilation of C++ or EC++, the `int_to_short` option will be invalid.

Data that are shared between C and C++ (EC++) program must be declared as the long or short type rather than as the int type.

### 2.2 Note on using MVTC or POPC instructions [Assembler]

In the assembly language, the program counter (PC) cannot be specified for MVTC or POPC instructions.

### 2.3 Note on the delete option for linkage [Optimizing linkage editor]

When a function symbol is removed by the delete option, its following function in the source program is not allowed to have a breakpoint at its function name on the editor in your debugging. If you would like to set a breakpoint via the Label window at the function entrance, set the breakpoint via the Label window or at the program code of the function.



## Chapter 3. Restrictions

This chapter describes restrictions on CC-RX.

### 3.1 Restriction on usage of math.h functions (frexp, ldexp, scalbn and remquo) in C++ language (including EC++)

An object is generated which will be an infinite-loop at execution when the actual argument of some function (frexp, ldexp, scalbn or remquo) of math.h is int-type, at compiling C++ or EC++ program.

Conditions:

This problem occurs when both (1) and (2) are satisfied.

(1) This program is in C++ or the lang=cpp option is effective.

(2) math.h is included and any of the following functions is called.

- (a) frexp(double, long\*) with 'int \*' type second argument (except when the first argument is float-type and the dbl\_size=8 option is effective).
- (b) ldexp(double, long) with 'int \*' type second argument (except when the first argument is float-type and the dbl\_size=8 option is effective).
- (c) scalbn(double, long) with 'int \*' type second argument (except when the first argument is float-type and the dbl\_size=8 option is effective).
- (d) remquo(double, double, long\*) with 'int \*' type third argument (except when the both the first and second arguments are float-type and the dbl\_size=8 option is effective).

Examples:

file.cpp:

```
// Example of compiling C++ source that generates an infinity-loop
#include <math.h>
double d1,d2;
int i;
void func(void)
{
    d2 = frexp(d1, &i);
}
```

Command Line:

```
ccrx -cpu=rx600 -output=src file.cpp
```

file.src: Example of the generated assembly program

```
_func:
    ; ... (Omitted)
    ; Calling substitute function of frexp
    BSR __$frexp__tm__2_f__FZ1ZPi_Q2_21_Real_type__tm__4_Z1Z5_Type
    ; ... (Omitted)

__$frexp__tm__2_f__FZ1ZPi_Q2_21_Real_type__tm__4_Z1Z5_Type:
L11:
    BRA L11 ; Calls itself ==> infinity-loop
```

### Countermeasures:

Select one of the following ways to avoid the problem.

- (1) Compile the program with the lang=c or lang=c99 option.
- (2) Change int or int \* into long or long \*.
- (3) Append the following declarations to each function that is being used.

```
/* For the frexp function */
static inline double frexp(double x, int *y)
{ long v = *y; double d = frexp(x,&v); *y = v; return (d); }
/* For the ldexp function */
static inline double ldexp(double x, int y)
{ long v = y; double d = ldexp(x,v); return (d); }
/* For the scalbn function */
static inline double scalbn(double x, int y)
{ long v = y; double d = scalbn(x,v); return (d); }
/* For the remquo function */
static inline double remquo(double x, double y, int *z)
{ long v = *z; double d = remquo(x,y,&v); *z = v; return (d); }
```

### Example of (2):

Change in file.cpp:

```
#include <math.h>
double d1,d2;
int i;
void func(void)
{
    long x = i; /* Accept as long type temporary */
    d2 = frexp(d1, &x); /* Call with long type argument */
    i = x; /* Set the result for variable 'i' */
}
```

### Example of (3):

Change in file.cpp:

```
#include <math.h>
/* Append declaration */
static inline double frexp(double x, int *y)
{ long v = *y; double d = frexp(x,&v); *y = v; return (d); }
double d1,d2;
int i;
void func(void)
{
    d2 = frexp(d1, &i);
}
```

### 3.2 Restriction of PIC/PID function (pic and pid options)

When a standard library is created by the library generator (lbgrx) with the pic or pid option specified, the following warning may appear once or more.

```
W0591301:"-pic" option ignored (When the pic option has been specified)
```

```
W0591301:"-pid" option ignored (When the pid option has been specified)
```

Despite the warning, the created standard library has no problems.

### 3.3 Eliminated options (for the C/C++ compiler)

(a) `-file_inline`, `-file_inline_path`

Specifying these options has no effect and the compiler will output a warning. Instead of `-file_inline` or `-file_inline_path`, write `#include` in the source code. In case of C and C99, `merge_files` can be used instead.

(b) `-enable_register`

This option is simply ignored and does not affect the generated code.

### 3.4 Notes on C/C++ source-level debugging (for the C/C++ compiler)

Even when `-debug` is specified, you may not be able to set a breakpoint or stop stepped execution on lines that

contain a dynamic initialization expression for a global variable (in C++),

are the first lines of functions that begin with a loop statement (e.g. `do` or `while`) and do not have an `auto` variable or of functions for which `#pragma inline_asm` has been specified, or

contain the control section and body of a loop statement (e.g. `for`, `while`, or `do`) written as a single line.

(a) The values of members of union type and of dummy variables that are to be passed via registers may be displayed incorrectly (e.g. in the [Watch] window).

### 3.5 Note on using sections that include the address 0xffffffff (in assembler)

If two or more `.section` directives in the assembly source code contain `.org` directives, the sections have the same name, and the sections overlap at 0xffffffff, the assembler outputs an internal error message (C0554098).

Example)

```
.section SS,ROMDATA
.org 0fffffffh
.byte 1
.byte 2 ; 0xffffffff
.section SS,ROMDATA
.org 0fffffffh
.byte 3; ; 0xffffffff
.end
```

### 3.6 Note on using `-form` and `-output` at the same time (in the linkage editor)

When `-form=rel` and `-output=<filename>` are specified for the linkage editor (`rlink`) at the same time, the filename extension given as `<filename>` is ignored and replaced with `.rel`.

Example)

```
rlink -form=relocate -output=DefaultBuild\lib_test.lib
```

The filename specified for output, `test.lib`, is changed to `test.rel`.

### 3.7 Note on using function names that begin with `_builtin` (for the C/C++ compiler)

Declaration of a function with a name that begins with `_builtin` and for which the definition is in `machine.h` in the `include` directory may lead to an internal error. In general, do not use any names that begin with an underscore (`_`) in your source code, since such names are reserved.

### 3.8 Restriction of `-merge_files`

Under certain conditions, compilation with `-merge_files` or `-whole_program` specified as the translation unit of code that includes union-type variables will produce error code F0530800 or warning code W0530811.

[Conditions]

If all the following conditions are satisfied, error code F0530800 or warning code W0530811 will be produced.

(1) `-merge_files` or `-whole_program` is specified.

(2) A union-type external variable having two or more members has been initialized outside any function, and, other than the member that have been initialized, a member has an alignment and size larger than the other member or members.

(3) The variable described in (2) above is declared as extern for reference by either of the following.

(3-1) Source files other than the definition of external variable described in (2) exists.

(3-2) Header files included directly or indirectly by the source files other than the definition of external variable described in (2) exists.

[Workarounds]

Take any of the following steps.

(1) Specify neither of the options in condition (1).

(2) Initialize the union-type external variable described in condition (2) within a function.

(3) Refer to the variables corresponding to condition (2) only in the source file that includes the definition of the external variable.

### 3.9 Restriction of `-cfi_ignore_module`

When compiling C/C++ source file using `-output=abs`, it isn't possible to specify the intermediately generated object files by `-cfi_ignore_module`.

Please generate an object file using `-output=obj` and indicate a generated object file by `-cfi_ignore_module`.

### 3.10 Restriction on using `fenv.h` when `-dpfpu` is specified

For the following standard library functions provided by `fenv.h`, even if `-dpfpu` is specified when compilation proceeds, these functions only specify and refer to the relevant values of the FPSW register; and not to the values of the DPSW register.

- \* `feclearexcept`
- \* `fegetexceptflag`
- \* `feraiseexcept`
- \* `fesetexceptflag`
- \* `fetestexcept`
- \* `fegetround`
- \* `fesetround`
- \* `fegetenv`
- \* `feholdexcept`
- \* `fesetenv`
- \* `feupdateenv`

To specify and refer to the values of the DPSW register, use the `__set_dpsw` and `__get_dpsw` intrinsic functions.

## Chapter 4. Standard Libraries Included

This chapter describes restrictions on standard libraries included in RX Family C/C++ Compiler.

This compiler package includes four library files (\*.lib) for the RX600. You can use any of the library files if they correspond to the options that you wish to specify. Using these files shortens the time required for building.

### 4.1 Library files

Table 4.1 shows the standard library files and compiler options.

Note:

The compiler options you specify should be the same as the microcontroller options defined for each of the library files listed in Table 4.1. Otherwise these library files are not usable, so specify your compiler options in the library generator to generate your own library file.

Library File	Purposes	Optimize <sup>*2</sup> Options	Microcontroller Options <sup>*1 *2</sup>		
			-endian	-cpu -rtti -exception -noexception	Others <sup>*3</sup>
<b>rx600lq.lib</b>	For the RX600 Optimization type:Speed Little endian	-speed -goptimize	-endian=little	-cpu=rx600	-round=nearest -denormalize=off -dbl_size=4 -unsigned_char -unsigned_bitfield -bit_order=right -unpack -fint_register=0 -branch=24
<b>rx600ls.lib</b>	For the RX600 Optimization type:Size Little endian	-size -goptimize			
<b>rx600bq.lib</b>	For the RX600 Optimization type: Speed Big endian	-speed -goptimize	-endian=big	-rtti=on -exception	
<b>rx600bs.lib</b>	For the RX600 Optimization type: Size Big endian	-size -goptimize			

**Table 4.1 Library Files**

\*Notes:

\*1 For details on microcontroller options, please see the “Microcontroller Options” columns of the “(1) Compile Options” of “section A.1.3 Options”, in the Integrated Development Environment User’s Manual: RX Build.

\*2 These option selections are same from the each default of them.

### 4.2 Using the library files

Copy the library file(s) included in the package from "lib" directory into a desired directory.

Then specify one of the copied library files for the Library option and start the linkage processing.

All trademarks and registered trademarks are the property of their respective owners.

## Notice

1. 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
  2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
  3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
  4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
  5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.  
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.  
Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
  6. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
  7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
  8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
  9. Renesas Electronics products and technologies shall 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. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
  10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
  11. This document shall not be reprinted, 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.
- (Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.  
(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)



### SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

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

**Renesas Electronics Corporation**  
TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

**Renesas Electronics America Inc.**  
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.  
Tel: +1-408-432-8888, Fax: +1-408-434-5351

**Renesas Electronics Canada Limited**  
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3  
Tel: +1-905-237-2004

**Renesas Electronics Europe Limited**  
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K  
Tel: +44-1628-651-700

**Renesas Electronics Europe GmbH**  
Arcadiastrasse 10, 40472 Düsseldorf, Germany  
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**  
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**  
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China  
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

**Renesas Electronics Hong Kong Limited**  
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: +852-2265-6688, Fax: +852 2886-9022

**Renesas Electronics Taiwan Co., Ltd.**  
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan  
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**  
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949  
Tel: +65-6213-0200, Fax: +65-6213-0300

**Renesas Electronics Malaysia Sdn.Bhd.**  
Unit 1207, 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 India Pvt. Ltd.**  
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India  
Tel: +91-80-67208700, Fax: +91-80-67208777

**Renesas Electronics Korea Co., Ltd.**  
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5338