### Old Company Name in Catalogs and Other Documents

On April 1<sup>st</sup>, 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 1<sup>st</sup>, 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.
- 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. 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; anticrime 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 majorityowned subsidiaries.
- (Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.



# M16C R8C FoUSB/UART Debugger

User's Manual

Renesas Microcomputer Development Environment System R8C Family R8C/2x Series Notes on Connecting R8C/2E, R8C/2F



Rev.1.00 2008.12

Active X, Microsoft, MS-DOS, Visual Basic, Visual C++, Windows and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

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

Intel and Pentium are registered trademarks of Intel Corporation.

Adobe and Acrobat are registered trademarks of Adobe Systems Incorporated.

All other brand and product names are trademarks, registered trademarks or service marks of their respective holders.

#### Keep safety first in your circuit designs!

• Renesas Technology Corporation and Renesas Solutions Corporation put the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

#### Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Renesas Technology
  product best suited to the customer's application; they do not convey any license under any intellectual property rights,
  or any other rights, belonging to Renesas Technology Corporation, Renesas Solutions Corporation or a third party.
- Renesas Technology Corporation and Renesas Solutions Corporation assume no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams, charts, programs and algorithms, represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation and Renesas Solutions Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation, Renesas Solutions Corporation or an authorized Renesas Technology product distributor for the latest product information before purchasing a product listed herein. The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corporation and Renesas Solutions Corporation assume no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors. Please also pay attention to information published by Renesas Technology Corporation and Renesas Solutions Corporation by various means, including the Renesas home page (http://www.renesas.com).
- When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation and Renesas Solutions Corporation assume no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Renesas Technology semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation, Renesas Solutions Corporation or an authorized Renesas Technology product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
- The prior written approval of Renesas Technology Corporation and Renesas Solutions Corporation is necessary to reprint or reproduce in whole or in part these materials.
- If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination. Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
- Please contact Renesas Technology Corporation or Renesas Solutions Corporation for further details on these materials or the products contained therein.

# **Table of Contents**

1.	Cor	nnecting the Target Board to the User System4
2.	Pre	paring the M16C R8C FoUSB/UART Debugger5
3.	Mer	mory Map When Using the R8C UART Debugger19
4.	Mor	nitor Program Occupied Area20
5.	Not	es on Using the R8C UART Debugger20
5	5.1.	Changing communication speed and restarting the R8C UART debugger after the R8C
		UART debugger is done 20
5	5.2.	User program ID code20
5	5.3.	User program download area21
5	5.4.	Frequency characteristics
5	5.5.	Limitations of SFR operations25
5	5.6.	Limitations on stop mode and wait mode25
5	5.7.	Watchdog timer
5	5.8.	Real-time operation of user program
5	5.9.	Executing anomalistic steps27
5	5.10.	Limitations on peripheral functions
5	5.11.	Limitations on the flag register
5	5.12.	Operation on peripheral I/O during a break

### 1. Connecting the Target Board to the User System



**Target Board** 





Figure 2 Circuit Using the RS-232C Cable (R8C/2E, 2F Groups)

### 2. Preparing the M16C R8C FoUSB/UART Debugger

The M16C R8C FoUSB/UART debugger (hereinafter R8C UART debugger) is used for debugging by connecting an RS-232 cable between the host computer and the target board.

A monitor program comes bundled and is automatically installed with the R8C UART debugger software.

For the R8C Family, the monitor program is automatically programmed when starting the R8C UART debugger. The monitor program does not need to be preprogrammed with an M16C FlashStarter or other flash programmer.

 a) To start the High-performance Embedded Workshop (HEW), go to "Start", "Programs", "Renesas", "High-performance Embedded Workshop", and "High-performance Embedded Workshop". You will see the following "Welcome!" dialog box.

Welcom	91	? 🛛
۶	C <u>C</u> reate a new project workspace	ОК
<i>_</i>		Cancel
	Open a recent project workspace:	
	C:\WorkSpace\sample\sample.hws	Administration
	C Browse to another project workspace	

- [Create a new project work space] Select this when creating a new workspace.
- [Open a recent project workspace] Select this when using an existing workspace. Previous workspaces are displayed in the pull-down menu.
- [Browse to another project workspace]
  - Select this when using an existing workspace, and there are no previous workspaces listed in the pull-down menu.

After selecting an existing workspace and pressing [OK], the screen under letter t) (Page 17) is displayed.

b) Select "Create a new project workspace". Press [OK].

Welcom	el	? 🛛
<b>3</b>	Create a new project workspace	ОК
~	C Open a recent project workspace:	Cancel
$\bowtie$	C:\WorkSpace\sample\sample.hws	Administration
	Browse to another project workspace	

c) The Project Generator starts. When a toolchain is preinstalled, the following screen is displayed.

New Project Workspace		? 🔀
New Project Workspace Projects Project Types Project Types Application C source startup Application Empty Application Further and the startup Application Library Debugger only - M16C FoUSB Debugger only - M16C Simulat	Workspace Name:         sample         Project Name:         sample         Directory:         C:\WorkSpace\sample         CPU family:         M16C	? X Erowse
Properties	_ool chain: Renesas M16C Standard ▼ OK	Cancel

• [Workspace Name]

Enter a name to create a new workspace. "sample" has been input as an example.

• [Project Name]

Enter a project name. If the project name is to be the same as the workspace name, it is not necessary to input anything in the [Project Name] field.

• [CPU family]

Select the applicable CPU family. In this case, M16C has been selected for the R8C Family.

• [Toolchain]

When using a toolchain, select the applicable toolchain name. When not using a toolchain, select "None".

• [Project Types] Window

Select a project type. The project type which supports the C start-up has been selected as an example.

- ? New Project-1/5-Select Target CPU. Toolchain version Э Toolchain version : 5.44.00 Ŧ Which CPU do you want to use for this project? CPU Series: M16C/30 ~ M16C/20 M16C/10 M16C/Tiny R8C/Tiny CPU Group: 27 ~ 28 29 2A 2B If there is no CPU group to be selected, select the "CPU Group" that a similar to hardware specification or select "Other". < Back Next > Finish Cancel
- d) Set the toolchain version.

Select the "Toolchain Version", "CPU Series", "CPU Group" and then press [Next]. In this example, R8C/27 Group is selected.

7

e) Select the MCU ROM size and other related settings.

New Project-2/5-Setting the Contents of Files to be Generated 🛛 💽 🔀				
New Project-2/5-Setting the Col	What kind of initialization routine would you like to create? ROM 32K Use Standard I/O Library (UART1) Use Heap Memory Heap Size: 0x80 Generate main() Function C source file			
Back	Use OnChip Debugging Emulator None Firmware Address: WorkRAM Address: Size: Next > Finish Cancel			

Select the MCU ROM size and heap size. Select "None" for the on-chip debugging emulator. And press [Next].

f) Set the stack.

New Project-3/5-Setting the Stack Area ? 🛛		
	What are the stack settings? User Stack Pointer : Use User Stack Stack Size: 0x80 Interrupt Stack Pointer : Stack Size: 0x80	
< Back	Next > Finish Ca	incel

Set the "Stack Size" and press [Next].

g) When the toolchain setting is complete, the following screen is displayed.

New Project-4/5-Setting the Target System for Debugging 💦 🔀		
	Targets :         M16C R8C FoUSB/UART         M16C R8C Simulator         M16C R8C Simulator         External Debugger :         none         M16C Family KD 30 V.4.10 Release 1         Target type :       R8C/Tiny	
< Back	Next > Finish Can	cel

Select "M16C R8C FoUSB/UART" and press [Next].

If necessary, select other targets.

h) Set the configuration file name.

Configurations are the build option settings (e.g., output of debug information or optimization) having their own names. The term "configuration" can also be referred to as "build configuration".

New Project-5/6-Setting the Debugger Options				
	Target name :         M16C R8C FoUSB/UART         Configuration name :         Debug_M16C_R8C_FoUSB_UART         Detail options :         Item         Setting         Modify			
< Back	Next > Finish Cancel			



i) Verify the file names to be generated.

 j) The figure below displays the files generated by the High-performance Embedded Workshop based on the above settings. Pressing [OK] launches High-performance Embedded Workshop.

Summary	? 🛛		
Project Summary:			
PROJECT GENERATOR     PROJECT NAME : sample     PROJECT DIRECTORY : C:\WorkSpace\sa     CPU SERIES : R8C/Tiny     CPU GROUP : 27     TOOLCHAIN NAME : Renesas M16C SI     TOOLCHAIN VERSION : 5.44.00     GENERATION FILES :         C:\WorkSpace\sample\sample\typedefine.h         define scalar types.         C:\WorkSpace\sample\sample\typedefine.h         define scalar types.         C:\WorkSpace\sample\sample\typedefine.h         define scalar types.         C:\WorkSpace\sample\sample\typedefine.h         initialize for C language.         C:\WorkSpace\sample\sample\typedefine.h         include some headder files.         C:\WorkSpace\sample\sample\typedefine.h         define the macro for initialization of sections.     } }	tandard Toolch		
<	>		
Click OK to generate the project or Cancel to abort.			
(OK)	Cancel		

 k) Check the section address. Under the "Build" menu, select "Renesas M16C Standard Toolchain".

🖗 sample - High-performance Embedded Workshop			∎₽⊠
File         Edit         View         Project         Build         Debug         Setup         Tools         Test         Window         He           Image: Image	100		
Build File Ctrl+F7	_ ▶ ♠ 爲 爲 ] ـ ـ 善 参 🕮 🚟 Ă   Debug	DefaultSession	- ≯ ↔
E- Sample 🛗 Build F7			
E G C source			
Gean Current Project			
initset ∰ Clean All Projects			
Build Multiple      Weather State     Street     Suid Multiple      Suid Mult			
Sampl Stop Tool Execution Ctrl+Break	_		
Cstart Include/Exclude Build     Initsct     Build Phases	_		
Build Phases			
sfr_r8 Build ⊆onfigurations			
Linkage Order			
Generate Makefile			
🕺 OJ OT AJ AT 😫 ST 🖉 🖻 🖬 ?			
Build ( Debug ) Find in Files 1 ) Find in Files 2 Macro ) T	st 👌 Version Control _/		
	💷 🔝 🔝 Default1 desktop		INS NUM

I) "Renesas M16C Standard Toolchain" is displayed. Click on the "Link" tab and select "Section Order" from the "Category" pull-down menu. Make sure the section start address does not overlap with the monitor program occupied area shown in Table 1. If there is any overlap, edit the address value.

Renesas M16C Standard Toolcha	in 🥐 🔀
Configuration	C Assembly Link Librarian Lmc RTOS CPL
Debug 🗨	Category : Section Order
All Loaded Projects C source file Assembly source file Assembly source file	Address       Section       Edit         0x00000400       data_SE       bss_SE         data_SO       bss_SO       data_ND         bss_NE       data_NO       bss_NO         stack       istack       istack         istack       heap_NE       0x00008000 rom_NE         rom_NO       data_SEI       ✓
	Options Link : -L "r8clib" -G -MS -O "\$(CONFIGDIR)\\$(PROJECTNAME).x30" -ORDER data_SE=0400,bss_SE,data_SO,bss_SO,data_NE,bss_NE,dat a_NO,bss_NO,stack,istack,heap_NE,rom_NE=08000,rom_NO, data_SEI,data_SOI,data_NEI,data_NOI,switch_table,program,i
	OK Cancel

m) Double-click the source program to launch the program editor.

🖗 sample - High-performance Embedded Workshop	- P 🗙
File Edit View Project Build Debug Setup Tools Test Window Help	
	2 🎘 🛞
Image: Sample         Image: Sample <td< td=""><td></td></td<>	
A OL OT AL AT 21 21 00 10 10 10 10 10 10 10 10 10 10 10 10	
Build ∧ Debug ∧ Find in Files 1 ∧ Find in Files 2 ∧ Macro ∧ Test ∧ Version Control	
Ready 🕅 🕅 🕅 🕅 Default 1 desktop	INS NUM

n) When the program is complete, under the "Build" menu, select "Build" or "Build All" to build the program.



o) The result of the build is displayed.



p) Connect with the target. A host computer can be connected with the target easily by switching to the session file in which the setting to use the R8C UART debugger is pre-registered.



q) The Init screen is displayed. Select the "Serial" radio button and press [Refer(ence)].

Init (M16C R8C FoUSB/UART)
MCU Debugging Information Run Mode Script
MCU: Refer
O Parallel 💿 Serial O LAN O LPT O USB
Port: COM1 💌
Baud Rate: 38400 💌
Monitor Debug
Start up for monitor debug
OK Cancel Help
🦳 Do not show this dialog box again.

r) Select "R8C-Tiny Series".

Select MCU Fi	le		? 🛛
Look <u>i</u> n: 🗀 M	IcuFiles	- + 1	-111 *
M16C-2D Gro M16C-6N Gro M16C-6S Gro M16C-6V Gro M16C-10 Ser M16C-24 Gro	up CM16C-26A Group up CM16C-28 Group up CM16C-29 Group ies CM16C-62A Group	R8C-Tiny Series	
File <u>n</u> ame: 🛛	.mcu		<u>O</u> pen
Files of <u>type</u> :	MCU Files (*.mcu)	•	Cancel

s) Select one of the following MCU files depending on the group.

R8C/2E Group: R5F212E4UART.MCU R8C/2F Group: R5F212F4UART.MCU

Select MCU	File			1	2×
Look jn: 🔀	R8C-Tiny S	eries	- 🗢 🗈 🕯	* 🎟 •	
<ul> <li>R5F21144</li> <li>R5F21144</li> <li>R5F21154</li> <li>R5F21154</li> <li>R5F21164</li> <li>R5F21164</li> <li>R5F21164</li> </ul>	UART.MCU .MCU UART.MCU .MCU	<ul> <li>R5F21174.MCU</li> <li>R5F21174UART.MCU</li> <li>R5F21184UART.MCU</li> <li>R5F21194UART.MCU</li> <li>R5F21237UART.MCU</li> <li>R5F21237UART.MCU</li> <li>R5F21256UART.MCU</li> </ul>	R5F21266	BUART.MCL SUART.MCL SUART.MCL SUART.MCL SUART.MCL	
<b>K</b> File <u>n</u> ame:	R5F21276	UART.MCU		<u>O</u> pen	Type: Date Size:
Files of <u>t</u> ype:	MCU Files	(*.mcu)	•	Cance	

In this example, R8C/27 Group is selected.

t) Select the appropriate "Port" and "Baud Rate" from the pull-down menus. Press [OK] and a monitor program is downloaded.

Init (M16C R8C	FoUSB/UART)				
MCU Debugging Information Run Mode Script					
MCU: R5F21276UART.MCU					
C Parallel 💿	Serial C LAN C	LPT C USB			
Port:	COM1 💌				
Baud Rate:	38400 💌				
Monitor Debug					
Start up for	monitor debug				
	OK Cance	el Help			
🔽 Do not show this dialog box again.					

u) To download a user program, select "Download File (X30 file)" in the "Download Modules" submenu under the "Debug" menu.

🖗 sample - High-performance Er	mbedded Workshi	op - [sample.c			
A File Edit View Project Build D	ebug Setup Tools	Test Window	Help _ Ə' >		
] D 🛩 E 🕼   👌   X 🗠	Debug Sessions		💌 🛝 🖓 🛗 📇 🤌 🛗 📇 📥 Debug 💽 SessionM16C_R8C_FoU 👤 🥕 🙊		
0 👿 16 10 8 2 🛒 !	Debug Settings		┣ ᠿ  ᠮᢧᡄ ╞ᢏ ]]   झ     驛  ] 疑   疑  ] ₯		
	EÎReset CP <u>U</u> E↓Go EûReset Go	F5 Shift+F5			
	E⊈ Reset Go E Free Go	Shirt+PS	*/		
initsct.c lintprg.c lintprg.c lintprg.c lintprg.c lintprg.c lintprg.c lintprg.c lintprg.c lintprg.c	Go To Cursor For Set PC To Cursor Run	Ctrl+Shift+Y	E :sample.c */ E :Ftp://May 30, 2008 */ CRIPTION :main program file. */ GROUP :27 */		
	Pc Display PC → Step In → Step Oyer	F11 F10	*/ s file is generated by Renesas Project Generator (Ver.4.12). */ */		
initsct.h¶ resetprg.h sfr_r827.h	Step Out Step Step Mode	Shift+F11	in (void)		
typedefine.h	Lalt Program				
	Gonnect Disconnect				
	S <u>a</u> ve Memory Verify Memory				
	Do <u>w</u> nioad Modules Unioad Modules	Þ	C:(WorlSpace sample sample Cebug sample.x30 - 00000000 All Download Modules		
Communication ERROR. Can't accept data. (160 Disconnected Connected	)14)				
Disconnected Connected					
Build Debug Find in Fi	Market A Buld A Debug A Find in Files 1 A Find in Files 2 A Macro A Test A Version Control /				
			🔝 🔝 🔝 Default1 desktop Read-write 4/16 20 INS NUM		

v) To reset the user program, select "Reset CPU" under the "Debug" menu.

🖗 sample - High-performance Embedded Workshop - [sample.c	1	- 7 🗙			
A File Edit View Project Build Debug Setup Tools Test Window	Help	_ 8 ×			
Debug Sessions	▼ 🏘 斎 斎 🛗 参  益 Debug 💽 SessionM16C_R8C_FoU 👤 🗡	<del>@</del>			
	9 9 🚳 🛼 🐂 🔲 🖳 🖓 🖗 🖓 🖓 👘				
Dependencies     Dependencies     Cstartdef.h     initsct.h     Or Step Over     F10     Or Step Out     Shift+F11	<pre># #/ E :sample.c #/ E :Ftp. May 30, 2008 #/ CRIPTION :main program file. #/ GROUP :27 #/ s file is generated by Renesas Project Generator (Ver.4.12). #/ in (void) </pre>				
Communication ERROR.	1	^			
Can't accept data. (16014) Disconnected					
Connected Disconnected Connected					
End A Debug / Find in Files 1 / Find in Files 2 / Macro / Test / Version Control /					
Reset CPU	📧 🔝 🛃 Default1 desktop 🛛 Read-write 🛛 4/16 🛛 20 🛛 INS	NUM			

w) The cursor moves to the top of the user program and debugging starts.



### 3. Memory Map When Using the R8C UART Debugger

Figure 3 shows a memory map (16 KB).



Figure 3 Memory Map (32 KB)

#### 4. Monitor Program Occupied Area

ROM / RAM	Occupied Area for Monitor Program
8 KB / 512 B	Vector FFE8h to FFEBh, FFECh to FFEFh, FFF4h to FFF7h
16 KB / 1 KB	RAM 6FFh to 7FFh Flash memory C000h to C9FFh Vector FFE8h to FFEBh, FFECh to FFEFh, FFF4h to FFF7h

Table 1 Monitor Program Occupied Area

### 5. Notes on Using the R8C UART Debugger

# 5.1. Changing communication speed and restarting the R8C UART debugger after the R8C UART debugger is done

The target MCU holds the baud rate value after the R8C UART debugger is done. Therefore, when changing communication speed and restarting the R8C UART debugger, a communication error occurs. (The R8C UART debugger can be started when using the previous communication speed). When changing communication speed, turn off the target power and turn on the power again.

#### 5.2. User program ID code

Set the ID code of the user program to <u>all FFh</u> when using the R8C UART debugger.

		-
Address	ID No.	Vector Table
0FFDFh – 0FFDCh	ID1	Undefined instruction
0FFE3h – 0FFE0h	ID2	Overflow
0FFE7h – 0FFE4h		BRK instruction
0FFEBh – 0FFE8h	ID3	Address match
0FFEFh – 0FFECh	ID4	Single step
0FFF3h – 0FFF0h	ID5	Watchdog timer, oscillation stop detection, voltage monitor 2
0FFF7h – 0FFF4h	ID6	Reserved
0FFFBh – 0FFF8h	ID7	Reserved
0FFFFh – 0FFFCh	(See Note)	Reset

 Table 2 ID Code Storing Address

Note: Refer to the hardware manual for the value set to address 0FFFFh.

#### 5.3. User program download area

As shown in Figure 3, a monitor program uses a part of RAM or flash memory when using the R8C UART debugger. The R8C UART debugger does not download a user program only in the area which overlaps with a monitor program. Note that the R8C UART debugger does not output an error at this time. When outputting an error, set the following.

a) Make "firm.c" and enter the following:

#include "typedefine.h"
#ifdefUART
#pragma section bss FirmArea
_far _UBYTE _firmarea[0xA00];
#endif

b) Add the file created above to the project. Under the "Project" menu select "Add Files...".When the file selection screen is displayed, select "firm.c".



c) To add the compile option, select "Renesas M16C Standard Toolchain..." under the "Build" menu.

🖗 sample - High-perforn	nance Embedded Workshop - [fiem.c]			
ile Edit View Project	Build Debug Setup Tools Test Window Help			_ 8 ×
0 🛩 🖬 🖉 🖂 🕺	Renesas M16C Standard Toolchain	해 윩 윩 🛛 🚟 🧇 🕮 🖽 🝝 🕞	ebug SessionM16C_R8C_FoU	- 🔭 💮
16 10 8 2	Build File Ctrl+F7	IPC YPC		
	🛗 Build F7			
	🛗 Build All			
🖻 🚯 sample	Build Multiple	. h″		
E 🔄 C source file	Clean Current Project			
- ≝ fiem.c - ≝ fvector.c	🚠 Clean All Projects	FirmArea		
— 🖆 heap.c	Update All Dependencies	a[OxA00];		
initset.c	Stop Tool Execution Ctrl+Break			
	Include/Exclude Build			
≚ sample.c				
- 🔄 Download mo				
Dependencie				
initset.h	Linkage Order			
sfr_r827.h	Generate Makefile			
		am.c		
	t 🖉 🖻 🖬 💡			
Phase M16C Lo	ad Module Converter finished			<u> </u>
Build Finishe O Errors, O W				
0 511013, 0 0				
				~
				>
	Find in Files 1 $\lambda$ Find in Files 2 $\lambda$ Macro $\lambda$ Test $\lambda$	Version Control /		
		💶 📰 🖪 Default1 desktop	Read-write 7/7 7	INS NUM

d) "Renesas M16C Standard Toolchain" is displayed. Select "Defines" from the "Show Entries For" pull-down menu, and then press [Add...].

Renesas M16C Standard Toolcha	in 🥐 🔀
Configuration Debug All Loaded Projects Sample C source file Assembly source file	C Assembly Link Librarian Lmc RTOS CPL   Category: Source  Show Entries For: Defines  [-D] Define a identifier:  Define Value Add  Remove
	Options C : -c -finfo -dir "\$(CONFIGDIR)" -R8C
	OK Cancel

e) "Set defined macro" is displayed. Enter "\_UART\_" in the "Macro" field and press [OK[.

Set defined	macro	? 🗙
<u>M</u> acro	_UART_	
<u>R</u> eplacement		
	ОК	Cancel

f) To set the link option, in the "Renesas M16C Standard Toolchain" window, click on the "Link" tab and select "Section Order" from the "Category" pull-down menu. Add "FirmArea\_NE" to the start address of the monitor program occupied area shown in Table 1.

Renesas M16C Standard Toolcha	in	? 🛛
Configuration	C Assembly Link	Librarian Lmc RTOS CPL
Debug 💌	Category : Section Order	<b>_</b>
All Loaded Projects	-ORDER data_SE=0400,bss_SE,data a_N0,bss_N0,stack,istack,	NFIGDIR)\\$(PROJECTNAME).x30" a_SO.bss_SO.data_NE.bss_NE.dat heap_NE.rom_NE=08000.rom_NO. E.data NDI.switch table.program=
		OK Cancel

When the user program overlaps with the monitor program occupied area due to the above settings, an error is output.

#### 5.4. Frequency characteristics

The monitor program operates in the range of the main clock (Xin) frequency shown below. The monitor program may not run with frequencies not listed below, so use an oscillator that has this frequency range.

1 MHz (minimum) to 20 MHz (maximum)

Table 3 lists each frequency and communication available speed. However, note that operation may not be possible when dividing the main clock and using it with less than 1 MHz even in the range of the above frequency. Do not select low-speed on-chip oscillator or Xcin clock as a main clock.

Frequency	Communication Speed (bps)					
Frequency	1200	2400	4800	9600	19200	38400
20 MHz	Х	Х	0	0	0	0
16 MHz	Х	Х	0	0	0	0
14 MHz	Х	Х	0	0	0	0
12 MHz	Х	Х	0	0	0	0
10 MHz	Х	0	0	0	0	0
8 MHz	Х	0	0	0	0	0
6 MHz	Х	0	0	0	0	0
4 MHz	0	0	0	0	0	Х
2 MHz	0	0	Ó	0	Х	Х
1 MHz	0	Ó	Ó	Х	Х	Х

Table 3 Communication Available Speed of Each Frequency

O: Communication available

X: Communication not available

Note: Communication may not be possible depending on temperature and voltage. If communication is not possible, lower the communication speed.

#### 5.5. Limitations of SFR operations

Table 4 lists the limitations of register operations. Changing registers that are disabled will cause the monitor program to malfunction.

		-	
Register	Default Value	Limitation	Change
Processor Mode Register 0	Reset to 00h	Single-chip mode only	Partially enabled
Processor Mode Register 1	Reset to 00h		Enabled
System Clock Control Register 0	Reset to 08h	Set the CM05 bit to 0.	Partially enabled
System Clock Control Register 1	Reset to 28h	Set bits CM13 and CM15 to 1. Set the CM14 bit to 0.	Partially enabled
High-Speed On-Chip Oscillator Control Register 0	Reset to 03h	Set the FRA00 bit to 1.	Partially enabled
High-Speed On-Chip Oscillator Control Register 1		Do not change this register.	Disabled
High-Speed On-Chip Oscillator Control Register 2	Reset to 03h		Enabled
Oscillation Stop Detection Register	Reset to 00000x00b	Set this register to 00h.	Partially enabled
Protect Register			Enabled
Flag Register		Writing to the D flag is ignored. Do not set the D flag to 1.	Partially enabled
ISP (Interrupt Stack Pointer)	Reset to 057Fh	Set an area not used by the monitor program.	Partially enabled

#### Table 4 Limitations on SFR Operation

#### 5.6. Limitations on stop mode and wait mode

When using stop mode or wait mode, start the R8C UART debugger in free-run mode, and close the RAM window, C watch window, and ASM window in advance. Also, do not operate the R8C UART debugger until the program stops at the break point by setting the break point after exiting stop mode or wait mode.

#### 5.7. Watchdog timer

The watchdog timer is refreshed while the monitor program is running. When using the watchdog timer while running the user program, note that by referring to or changing memory content, the monitor program intervenes and the watchdog timer is refreshed.

#### 5.8. Real-time operation of user program

• Sampling run mode (also known as sampling mode)

In sampling mode, execution status of the user program will be regularly monitored when executing Go and Come. Therefore, it is possible to detect when the user program is stopped by a break or other command. Select this mode when performing a normal debug.

#### • Free run mode

In free run mode, execution status of the user program will not be monitored when executing Go and Come. Although real-time operation of the user program is secure, it is not possible to detect if the user program is stopped by a break or other command. Therefore, even when the user program stops, the R8C UART debugger does not stop executing Go and Come. Press STOP to stop the R8C UART debugger.

Note: In free run mode, use the R8C UART debugger while the RAM window, C watch window, and ASM window are closed.

#### 5.9. Executing anomalistic steps

• Software interrupt instruction

The step execution cannot be performed continuously to the instruction internal process of the instructions (undefined instruction, overflow, BRK instruction, and INT instruction) which generate the software interrupts.

Example: INT instruction



• INT instruction

To debug the program using the INT instruction, set the software break for the INT instruction process and use the Go command.

Example:



#### 5.10. Limitations on peripheral functions

UART1 is used for communication between the monitor program and the host computer. Do not use UART1 in the user program. Do not connect the pins below to other pins as they are used for communication with the host computer.

R8C/2E,2F Groups
 P3\_7/TRAO/TRFO11 (2 pin), P4\_5/INT0 (9 pin)

#### 5.11. Limitations on the flag register

When operating the flag register in a user program, execute the **FSET instruction and FCLR instruction** not to change the debug flag (D flag).

#### 5.12. Operation on peripheral I/O during a break

Although an interrupt cannot be accepted during a break, peripheral I/O continues operating. For example, when stopping a user program by a break during operating a timer, the timer continues counting, but the timer interrupt cannot be accepted.

## **REVISION HISTORY** M16C R8C FoUSB/UART Debugger

Rev. Date	Date	Page	Description	
	raye	Summary		
1.00	Dec. 12, 2008	-	First edition issued	

# M16C R8C FoUSB/ART Debugger User Manual Notes on Connecting R8C/2E, R8/2F

Publication Date	Rev.1.00 Dec. 12, 2008
Published by:	Sales Strategic Planning Div.
r ublished by.	Renesas Technology Corp.
Edited by:	Renesas Solutions Corp.

© 2008. Renesas Technology Corp. and Renesas Solutions Corp., All rights reserved. Printed in Japan.

## M16C R8C FoUSB/UART Debugger User's Manual



Renesas Electronics Corporation 1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan