カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジ が合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社 名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い 申し上げます。

ルネサスエレクトロニクス ホームページ (http://www.renesas.com)

2010年4月1日 ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社(http://www.renesas.com)

【問い合わせ先】http://japan.renesas.com/inquiry

ご注意書き

- 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
- 2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的 財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の 特許権、著作権その他の知的財産権を何ら許諾するものではありません。
- 3. 当社製品を改造、改変、複製等しないでください。
- 4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
- 5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところに より必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の 目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外 の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
- 6. 本資料に記載されている情報は、正確を期すため慎重に作成したものですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
- 7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、 各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確 認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当 社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図 されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、意図 されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、 「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または 第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、デ ータ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
 - 標準水準: コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、 産業用ロボット
 - 高品質水準:輸送機器(自動車、電車、船舶等)、交通用信号機器、防災・防犯装置、各種安全装置、生命 維持を目的として設計されていない医療機器(厚生労働省定義の管理医療機器に相当)
 - 特定水準: 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器(生命維持装置、人体に埋め込み使用するもの、治療行為(患部切り出し等)を行うもの、その他 直接人命に影響を与えるもの)(厚生労働省定義の高度管理医療機器に相当)またはシステム 等
- 8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
- 9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
- 10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用 に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、 かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し て、当社は、一切その責任を負いません。
- 11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお 断りいたします。
- 12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご 照会ください。
- 注1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレク トロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいい ます。



M16C R8C FoUSB/UARTデバッガ

ユーザーズマニュアル

ルネサスマイクロコンピュータ開発環境システム R8Cファミリ R8C/2xシリーズ R8C/2K、R8C/2L 接続時の注意事項



Rev.1.00 2008.12

Active X、Microsoft、MS-DOS、Visual Basic、Visual C++、WindowsおよびWindows NTは、米国Microsoft Corporation の米国およびその他の国における商標または登録商標です。

IBMおよびATは、米国International Business Machines Corporationの登録商標です。

Intel, Pentiumは、米国Intel Corporationの登録商標です。

AdobeおよびAcrobatは、Adobe Systems Incorporated(アドビシステムズ社)の登録商標です。

その他すべてのブランド名および製品名は個々の所有者の登録商標もしくは商標です。

安全設計に関するお願い

弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

- ●本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
- ●本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは責任を負いません。
- ●本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、予告なしに、本資料に記載した製品又は仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前に株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (http://www.renesas.com) などを通じて公開される情報に常にご注意ください。
- ●本資料に記載した情報は、正確を期すため、慎重に制作したものですが万一本資料の記述誤りに起因する損害がお客様に生じた場合には、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズはその責任を負いません。
- ●本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、 プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可 否を判断してください。株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、適用可否に対 する責任は負いません。
- ●本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを 目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原 子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、株式会社ルネサス テク ノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店へご照会ください。
- ●本資料の転載、複製については、文書による株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズの事前の承諾が必要です。
- ●本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたら株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店までご照会ください。

目次

1.	ユーザシステムとの接続例	4
2.	M16C R8C FoUSB/UARTデバッガの準備	5
3.	R8C UARTデバッガ使用時のメモリマップ	19
4.	モニタプログラム占有領域	
5.	R8C UARTデバッガ使用時の注意事項	
5.1.	R8C UARTデバッガ終了後、通信速度を変更して再度R8C UARTデバッガを	起動する
	場合	20
5.2.	ユーザプログラムのIDコードについて	
5.3.	ユーザプログラムのダウンロード可能領域について	21
5.4.	周波数特性について	24
5.5.	SFR操作に関する制限事項	25
5.6.	ストップモード、ウエイトモードに関する制限事項	
5.7.	ウォッチドッグタイマについて	25
5.8.	ユーザプログラムのリアルタイム性について	
5.9.	例外的なステップ実行について	27
5.10.	周辺機能に関する制限事項	
5.11.	フラグレジスタに関する制限事項	
5.12.	ブレーク中の周辺I/Oの動作について	



図 1 ユーザシステムとの接続例



図 2 RS-232C ケーブルを使用した回路例(R8C/2K,2L グループ)

2. M16C R8C FoUSB/UART デバッガの準備

M16C R8C FoUSB/UART デバッガ(以下、R8C UART デバッガという)では、PC とターゲ ットを RS-232C で接続し、デバッグができます。 モニタプログラムは、「M16C R8C FoUSB/UART デバッガ」インストール時にバンドルされ ますので、ユーザで準備する必要はありません。 また R8C ファミリの場合、R8C UART デバッガ起動時に自動的にモニタプログラムを書き 込みます。あらかじめ M16C FlashStarter 等のフラッシュライタでモニタプログラムを書き 込んでおく必要はありません。

a) High-performance Embedded Workshop を起動します。「スタート」-「プログラム」 - 「Renesas」-「High-performance Embedded Workshop」-「High-performance Embedded Workshop」をクリックすると、「ようこそ!」ダイアログボックスが表示され ます。

ようこそ!		? 🔀
-オプション:-		ОК
2	○ 新規プロジェクトワークスペースの作成(⊆)	キャンセル
	● 最近使用したプロジェクトワークスペースを開く(の)	アドミニストレーション(<u>A</u>)
	C:¥WorkSpace¥aaa¥aaa.hws	
	○ 別のプロジェクトワークスペースを参照する(B)	

 「新規プロジェクトワークスペースの作成」 ワークスペースを新規作成する場合に選択します。
 「最近使用したプロジェクトワークスペースを開く] 既存のワークスペースを使用する場合に選択します。 開いたワークスペースの履歴が表示されます。
 「別のプロジェクトワークスペースを参照する] 既存のワークスペースを使用する場合に選択します。 開いた履歴が残っていない場合に使用します。

既存のワークスペースを選択し、[OK]ボタンを押すとt)の画面になります。

b) [新規プロジェクトワークスペースの作成]ラジオボタンを選択し、[OK]ボタンを押して ください。

ようこそ!	? 🔀
-オプション:	ОК
∑ 新規プロジェクトワークスペースの作成 ○	キャンセル
○ 最近使用したプロジェクトワークスペースを開く():	アドミニストレーション(<u>A</u>)
C:¥WorkSpace¥aaa¥aaa.hws	
○ 別のプロジェクトワークスペースを参照する(B)	

c) Project Generator が開始されます。ツールチェインがインストールされている場合、以 下の画面が開きます。

新規プロジェクトワークスペース		?×
プロジェクトタイプ プロジェクトタイプ Application Cource startup Application Empty Application Fundation Debugger only - M16C Compa Debugger only - M16C Compa Debugger only - M16C E8 SYS Debugger only - M16C FoUSE Debugger only - M16C FoUSE Debugger only - M16C PC470 Debugger only - M16C PC750 Debugger only - M16C PC750 Debugger only - M16C Simula Debugger only - M16C Simula	ワークスペース名(W): sample プロジェクト名(P): sample ディレクトリ(D): C¥WorkSpace¥sample CPU種別(C): M16C マールチェイン(T): Renesas M16C Standard ▼	<u>. </u>
	OKキャン	セル

・[ワークスペース名]

新規作成するワークスペース名を入力してください。ここでは例として "sample" と入力します。

- 「プロジェクト名]
 プロジェクト名を入力してください。ワークスペース名と同じでよろしければ、入力する必要はありません。
- [CPU 種別]
 該当する CPU 種別を選択してください。R8C ファミリでは "M16C" を選択してください。

・[ツールチェイン]

ツールチェインをご使用になる場合、該当するツールチェイン名を選択してください。 使用しない場合、[None]を選択してください。

・[プロジェトタイプ]リストボックス
 使用したいプロジェクトタイプを選択してください。ここでは例として、Cスタートアップに対応したプロジェクトタイプを選択します。

New Project-1/5-Select Target Cl	PU.Toolchain version	? 🗙
	Toolchain version : 5.43.00	•
	Which CPU do you want to use for this project? CPU Series:	
	M16C/20 M16C/10 M16C/Tipy	
	CPU Group:	
20000	28 29 2A 2B	
and a second	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 Car	ncel

d) 次に、ツールチェインの設定を行います。

使用するツールチェインバージョン、CPU シリーズ、CPU グループを選択し、[Next]ボタンを 押してください。ここでは例として、R8C/27 グループを選択します。

New Project-2/5-Sett	ing the Contents of Files to be Generated 🛛 🛛 🛛 🔀
	What kind of initialization routine would you like to create? ROM size: 32K Ise Standard I/O Library (UART1) Use Heap Memory Heap Size: 0x80 Generate main0 Function C source file Use OnChip Debugging Emulator None Firmware Address: 0x Size: 0x Work RAM Address: 0x Size: 0x
I	< Back Next > Finish Cancel

e) 次に、使用するマイコンの ROM サイズ等の設定を行います。

使用するマイコンの ROM サイズ、ヒープサイズ等を選択し、[Next]ボタンを押してください。 オンチップデバッギングエミュレータには"None"を選択してください。

New Project-3/5-Setting the Stack Area	? 🛛
	nat are the stack settings? User Stack Pointer : ✓ Use User Stack Stack Size: □x80
	interrupt Stack Pointer : Stack Size: 0x80
< Back Next >	Finish Cancel

f) 次に、スタックの設定を行います。

スタックサイズ等を設定し、[Next]ボタンを押してください。

New Project-4/5-Setting the Tar	get System for Debugging	? 🗙
	Targets : M16C R8C Compact Emulator M16C R8C FoUSB/UART M16C R8C PC7501 Emulator M16C R8C Simulator R8C E7 SYSTEM R8C E8 SYSTEM External Debugger : M16C Family KD30 V4.00 Release 1A M16C Family KD30 V4.00 Release 1A M16C Family PD30MF V1.00 Target type : R8C/Tiny	
< Back	Next > Finish C	Jancel

g) ツールチェインの設定が終了したら、以下の画面が表示されます。

ここで、M16C R8C FoUSB/UART にチェックし、[Next]ボタンを押してください。 必要であれば、他の製品にもチェックをしてください。

h) 次に、コンフィグレーションファイル名を設定します。
 コンフィグレーションとは、デバッグ情報の出力の有無、および最適化の有無といった
 ビルドのオプションの設定内容に名前を付けて保持しているものです。

New Project-5/6-Setting the Debug	ger Options	? 🗙
	Target name : MIGC RSC FoUSB/UART Configuration name : Debug_MIGC_RSC_FoUSB_UART Detail options : Item Setting Modify	
< Back	Next > Finish Car	ncel

New Project-6/6-Changing the File Na	mes to be	Grea	ted 🤶 🔀
New Project-6/6-Changing the File Nat	The followin generated: File Na typedefine resetprg resetprg initsct fvector intprg sfr_r827 sfr_r827 heap sample cstartdef	e Creat e source Ext h c h c h c h c h c h c h	ted ? X e files will be <u>Description</u> define scalar types. initialize for C language. include some headder files. initialize each sections. define the macro for initiali: define the macro for initiali: define the fixed vector table define the fixed vector table define the sfr register. (for define the sfr register. (for define the size of heap. main program file. define the size of stack.
Contraction of the second s	<		
< Back	Next >		Finish Cancel

j) これまでの設定により High-performance Embedded Workshop が生成するファイルが 表示されます。[OK]ボタンを押すと、High-performance Embedded Workshop が起動し ます。

Summary	? 🔀		
Summary Project Summary: PROJECT GENERATOR - PROJECT NAME: san PROJECT DIRECTORY: C:¥ CPU SERIES: R80 CPU GROUP: 27 TOOLCHAIN NAME: Rer TOOLCHAIN VERSION: 5.43 GENERATION FILES: C:¥WorkSpace¥sample¥sample¥ty define scalar types. C:¥WorkSpace¥sample¥sample¥re initialize for C language. C:¥WorkSpace¥sample¥sample¥re	Pipe WorkSpace¥sample¥sample >/Tiny lesas M16C Standard Toolchain 100 pedefine.h esetprg.c		
 include some headder files. C:¥WorkSpace¥sample¥sample¥initsct.c initialize each sections. C:¥WorkSpace¥sample¥sample¥initsct.h define the macro for initialization of sections. 			
Click OK to generate the project or C	Cancel to abort.		
🔽 Generate Readmettxt as a summa	ry file in the project directory		
[OK Cancel		

k) セクションアドレスの確認をします。「ビルド」-「Renesas M16C Standard Toolchain」 を選択してください。

🟟 sample - High-performance Embed	ded Workshop				- 2 🛛
ファイル(E) 編集(E) 表示(V) ブロジェクト(P)	ビルド(四) デバッグ(四) 基本設定(U) ツール(田)	テストの ウィンドウ心	へルプロク		
	Renesas M16C Standard Toolchain	DefaultSession	🖃 🖄 😡 🛛 🖓 🕅	■性肾肾	
35	(\$ 1)/(///() Orl+F7				
	EDLF(E) F7 株 オペアをFilF(A)				1
B- S sample	補数ビルド回し				
🗟 🔄 C source file	クリーン アクティブブロジェクト(L)				
fvector.c heap.c	クリーン全プロジェクトモン				
≟ initsete	すべての後年間は多を更新しい				
⊥ resetpre.c	として、 して、 して、 して、 して、 して、 して、 して、				
🛁 🛋 sample.c	ビルドからお余外の南部家田				
cstartdeth	ビルドフェーズ(12)				
- III nrtscth - III resetpre.h	ビルドの構成(2)				
B strj627h	リンク順の指定(8)				
- (i) (speciaries	Makeファイルの生成(3)_				
	Concept International Con-				
	12				
H SP N IN IX IX IA IA 10 10	8				
					2
					1
Build (Debug) Find in Files) M	acro) Test) Version Control /				2
			ह स स	E Default1 desktop	INS

 I) 「Renesas M16C Standard Toolchain」画面が表示されます。「リンカ」タブを選択し、カ テゴリの「セクション配置順序」を選択してください。セクションアドレスの先頭が、表 1 に示すモニタプログラム占有領域と重なっていない事を確認してください。重なっている場合は、アドレス値を編集してください。

Renesas M16C Standard Toolchain	· · · · · · · · · · · · · · · · · · ·
コンフィグレーション: Debug ■ All Loaded Projects ■ © Source file ■ © Assembly source file	コンパイラ アセンモ リンカ ライブラリアン ロードモジュールコンパー・・ カテゴリ (Y): でクション配置順序 Address Section istack heap NE rom_NO data_SDI data_SOI data_SOI data_SOI data_NII data_NOI switch_table program program Dx0000FEDC vector リンカオブション(T): -L "%clib" -G -MS -O "\$(CONFIGDIR)¥\$(PROJECTNAME).x 30" -ORDER data_SE=0400 bss_SE,data_SO,bss_SO,data_NE,bss_NE,data_NO,bss_NO,stack,is vector
	OK キャンセル

m) ソースプログラムをダブルクリックするとエディタが起動し、編集ができます。

w sample - High-performance Embedded Workshop つっくまた) 原葉(E) 本子(A) づつづっわん(P) PER(B) デバッグ(D) 基本設定(D)	
	Detaursesson P(X, W) Mallosc T(X, W)
95	
Constantial and a second a	
g. 8. 9. 2.	
취 01 07 AL AT 입1 입7 / 1월 묘 ?	
	(A)
4	
≤ ≥ [\ Build & Debug & Find in Files & Macro & Test & Version Control /	
Ready	Defaulti desktop DKS NUM

n) プログラム作成後、「ビルド」--「ビルド」または「すべてをビルド」 でビルドできます。

🖗 sample - High-performanc	ce Embedded	l Workshop = [sample.c]	
	プロジェクトの	ビルド(8) デバック(1) 基本設定(1) ツール(1)	521-S 940490W AU709 - 6 ×
🗋 🗅 🚅 🗟 🖉 🎯 👗 🐚	🖻 \varTheta 🛛	Renesas M16C Standard Toolchain	w/Session 💌 🦄 🗣 🖓 💌 💌 🛱 🖌
<u>s</u>		③ エノパイル(E) Ctrl+F7	
<u> </u>	(2.6.8. C	HARD FT	
- S sample	1/11/-5 0	補数ビルド他)	
B-C Source file	2 3	クリーン アクティブプロジェクト(1)	
heap.c	4	クリーン全プロジェクト(E)	*/
initset.e	6 7	9~C00819001+2.0010/	*/
2) resetpre.c	8 9	PULISTICIES CENTRE	roject Generator (Yer.4.8). */ */
Dependencies Dependencies	11 12	F&K7*~	· · · · · · · · · · · · · · · · · · ·
initseth	13 14	Falender(n)	
— ■ resetproh — ■ str_r627h	15 16	0.2000000000	
_ i typedefine/		Malana (1.0 th th(2))	
		maxe, 71 /W/1±0/102/-	1
	-		15
			212
	Sample	c	
01 01 AL AT 21 21 4	2 10 10 1	?	
			2
Build (Debug) Find i	in Files Macro	λ Test λ Version Control	
すべてをビルドします			🖬 🖾 🖾 Default1 desktop Read-write 1/16 1 INS NUM

o) ビルドの結果が表示されます。

ファイル(E) 編集(E) 表示(V)	ie Embedded Workshop - Isampie.c] ブロジェクトビア ビルド図 デバッグロ 基本設定(リーツール(ローテスト(S) ワインドウ(M) ヘルブ(M)	
	응 🕀 🖉 팬 프 🍈 Debug 💌 DefaulSession 💌 거 😜 🐂 💌 백 융 🍒	
3		
<u> </u>	[行義長] 5 [19-2	
Sample	1 //	
G source file G fvector.c	3 /* FilE temple.c */ 4 /* DATE :Red.Exb.20.2009 */	
→ ▲ heap.c	5 //* DESCRIPTION imain program file. */ 6 /* CPU GROUP :27 */	
1 intpre.c	7 /* This file is generated by Renesas Project Cenerator (Yer.4.8). */	
i sample.c	9 /* *//	
Dependencies Estartdef.h	11 12 void main(void)	
initseth initsetprg.h	14 15 }	
sfr_r827.h	16	
g. D. a. B. i	S samples	
OL OT AL AT SU		
Load Module Converte	er (Imc30) for R8C/Tiny,#16C/60 Series Version 4-01-01.000	
Copyright(C) 2005. R and Renesas Solution	Renesas Technology Corp. ms Corp., All Rights Reserved.	
Phase M16C Load Modu	ule Converter finished	
Build Finished		
0 Frrors, 0 Warpings		
0 Errors. O Warnings		
0 Errors, 0 Warnings		

p) 次に、ターゲットと接続します。R8C UART デバッガを使用する設定があらかじめ登録 されているセッションファイルに切り替えることにより、簡単に接続できます。

🖗 sample - High-performance Embedded Workshop - [sample.c]	1 d 🗙
ション・シャット 御書の 単生の ないろう ちょう かん かい かいろう かくたい マント アン・マント ション・ション・ション・ション・ション・ション・ション・ション・ション・ション・	- 8 ×
Urdrude-start Seasc-MIGL_RBC_FBUSE_IU	
	1
S W sample	E
Course Tile 2 * FILE issuple.c */	-
→ in treators 4 /* DATE :: ##d, Feb 20, 2008 #/ → in heaps 5 /* DESCRIPTION ::sain programs file. #/	
-3) interset 6 /* CPU GROUP :27	
resetprec /* This file is generated by Renesas Project Generator (Ver.4.8). #/ // //	
a) catardefh 12 void sain(void) a) i	
eventurgh 121	
ut/start 1 16	
	1
	21.6
W. W. W. C. Sample	
<u>취</u> 01 07 AL AT 입법 21 22 19 🖬 ?	
Load Wodule Converter (10x30) for R8C/Tiny.W16C/60 Series Version 4.01-01-000	-
and Reneas Solutions Corp. All Rights Reserved.	
Phase NIBC Load Module Converter finished	
Build Finished 0. Froms. 0. Ternings	
	-
	*
C Bunded A Debus A Find in Films A Marco A Test A Version Control /	>
	1611 164

q) Init 画面が表示されます。[Serial]ラジオボタンを選択し、[参照]ボタンを押してください。

1	nit (M16C R8C	FoUSB/UART)		×
	MCU デバッグ情		フリプト	
	MCU:		参照	
	C Parallel	Serial C LAN	N CLPT CUSB	
	Port:	СОМЗ	•	
	Baud Rate:	38400	•	
	Monitor Debug	ガライブラリを使用する.		
		ОК	キャンセル ヘルプ 「今後、このダイアログボックスを表示し」	ない

r) 「R8C-Tiny Series」を選択してください。

Select MCU File	
ファイルの場所型: 🔁 McuFiles 💽 🖛 🗈 💣 🎟 🕇	
M16C-2D Group M16C-26 Group R8C-Tiny Series M16C-6N Group M16C-26A Group M16C-6S Group M16C-28 Group M16C-6V Group M16C-29 Group M16C-10 Series M16C-62A Group M16C-10 Series M16C-62A Group M16C-24 Group M16C-62P Group	
ファイル名(N): *.mcu 閉(の)	
ファイルの種類(I): MCU Files (*.mcu) マ キャンセル	

s) MCU ファイルは、グループにより以下の MCU ファイルを選択してください。 R8C/2K グループ:R5F212K4UART.MCU R8C/2L グループ:R5F212L4UART.MCU

Select MCU File		? 🛛
ファイルの場所型: 🔂 R8C-Tiny S	Series 💽 🗲	• 🖬 📩
Image: R5F21154.MCU Image: R5F2 Image: R5F21154UART.MCU Image: R5F2 Image: R5F21164.MCU Image: R5F2 Image: R5F21164UART.MCU Image: R5F2 Image: R5F21174.MCU Image: R5F2 Image: R5F21174UART.MCU Image: R5F2 Image: R5F21174UART.MCU Image: R5F2	21184UART.MCU B R5F2127 21194UART.MCU B R5F2128 21237UART.MCU B R5F2129 21256UART.MCU 21258UART.MCU 21266UART.MCU	6UART.MCU 4UART.MCU 4UART.MCU
<		>
ファイル名(N): R5F21276UAR	T.MCU	開((_)
ファイルの種類(①: MCU Files (*.m	ncu)	 キャンセル

- ここでは例として、R8C/27 グループを選択します。
 - t) COM ポートとボーレートを選択し、「OK」を押すとモニタプログラムをダウンロードします。

Init (M16C R8C FoUSB/UART)	×
MCU デバッグ情報 実行モード 起動スクリプト	
MCU: R5F21276UART.MCU 参照	
C Parallel 💿 Serial C LAN C LPT C USB	
Port: COM3	
Baud Rate: 🛛 🕄 🕄 🐨	
Monitor Debug	
└─ モニタデバッガライブラリを使用する.	
OK ++>>セル ヘルプ	
🥅 今後、このダイアログボックスを表示しな	<i>(</i>)3

u) 「デバッグ」-「ダウンロード」-「ダウンロードファイル(X30 ファイル)」でユーザ プログラムをダウンロードします。

🖗 sample – Hich-performance Embedded Workshop – [sample.c] 📃 🖸 🔀				
◇ ファイル(E) 編集(E) 表示(V) プロジェクト(E) ビルド(B) 5	F/15/20) 基本総定位 ツール① テスト⑤ ウムドウ他 ヘルブゼ - 8	×		
🗅 🚅 🖉 🦉 X 🖻 🖄 (🖗 🖽 🖽	デバッグセッション(型) (IGC_ROC_FOUS 💌 入 😦 🐂 🔍 💌 🙀 🛣			
S 0 15 10 8 2 9 9 14	デバッグの設定位)。			
1	EFOPUDIteskQD			
	至L要行位) F5	-1		
🖻 🚱 Sample 👔 🖓 💭 👌	到 y toy H 使男行 (E) Shift+F5			
B-G C source file 行番号 パー S- ソース B	E1 27-第行			
1 (**** 1 been c 2 /*	目的カージル位置まで実行(2)	1		
initsetc 3 /* 0 1	■μcカーソル位置にPO値を設定(P) */			
	会体を指定して実行(19)- */ かりの意味を行うの。 */			
🚠 sample.c 7 /* T	Renound a project Generator (Ver.4.8). */			
	(1) 7577757 (2) (1) (1) (1) (1) (1) (1) (1) (1) (1) (1			
Dependencies 11 void	(P ステップアウト(0) Shift+F11			
initeeth 13 (ステップ(2)			
	27x7E-F00 ·			
typedefine#	第 7075ムの停止(H)			
	2000HL(2)			
	34U01447(B).			
	720-FQ) All Download Modules (A)			
	-	1		
	2			
🔄 🗐 🕘 💆 🖉 Samples				
A 01 01 A1 A1 21 21 0 10 10 10 10				
Connected		_		
I ≥ ∫ Build) Debug √ Find in Files) Macro) Test) W	Version Control	-		
	図 図 図 Default1 desktop Read-write 1/16 1 INS N	UM		

v) 「デバッグ」-「CPU のリセット」でユーザプログラムをリセットします。



w) ユーザプログラムの先頭にカーソルが移動し、デバッグを開始できます。



3. R8C UART デバッガ使用時のメモリマップ

図 3に16KB内蔵版のメモリマップを示します。



図 3 16KB 内蔵版メモリマップ

4. モニタプログラム占有領域

ROM / RAM	モニタプログラム占有領域	
8KB / 1KB	ベクタ等 FFE8h~FFEBh,FFECh~FFEFh,	
	FFF4h~FFF7h	
16KB / 1.5KB	RAM 8FFh~9FFh	
	フラッシュメモリ COOOh~C9FFh	
	ベクタ等 FFE8h~FFEBh,FFECh~FFEFh,	
	FFF4h~FFF7h	

表1 モニタプログラム占有領域

5. R8C UART デバッガ使用時の注意事項

5.1. R8C UART デバッガ終了後、通信速度を変更して再度 R8C UART デバッ ガを起動する場合

R8C UART デバッガ終了後、ターゲットマイコンはボーレート値を維持します。そのため、 通信速度を変更して再度 R8C UART デバッガを起動した場合、通信エラーが発生します(前 回の通信速度を使用する場合は R8C UART デバッガを起動できます)。

通信速度を変更する場合、ターゲットの電源をオフにしてから、再度電源を投入してくだ さい。

5.2. ユーザプログラムの ID コードについて

R8C UARTデバッガを使用する場合、ユーザプログラムのIDコードは必ず<u>オールFFh</u>にして ください。

アドレス	ID No.	ベクタテーブル
0FFDFh – 0FFDCh	ID1	未定義命令
0FFE3h – 0FFE0h	ID2	オーバフロー
0FFE7h – 0FFE4h		BRK 命令
0FFEBh – 0FFE8h	ID3	アドレス一致
0FFEFh – 0FFECh	ID4	シングルステップ
0FFF3h – 0FFF0h	ID5	ウォッチドッグタイマ、
		発振停止検出、電圧監視2
0FFF7h – 0FFF4h	ID6	予約
0FFFBh – 0FFF8h	ID7	予約
0FFFFh – 0FFFCh	(注 1)	リセット

表2 ID コードの格納番地

注1:0FFFFh 番地に設定する値は、ハードウエアマニュアルを参照してください。

5.3. ユーザプログラムのダウンロード可能領域について

R8C UARTデバッガを使用する場合、図 3に示すようにRAM、Flash Memoryの一部をモニ タプログラムが使用します。R8C UARTデバッガは、ユーザプログラムがモニタプログラ ムと重なった場合、モニタプログラムと重なった領域のみダウンロードを行いません。 このとき、R8C UART デバッガはエラー出力をしませんのでご注意ください。エラー出力 を行う場合は、以下の設定を行ってください。

a) 「firm.c」を作成し、以下の内容を記述してください。

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

b) 作成したファイルをプロジェクトに追加します。「プロジェクト」-「ファイルの追加」 を選択してください。ファイルの選択画面が表示されるので「firm.c」を選択してください。

🖗 UseOnGhipDebuggerEmu	latorNone - High-performance Em	bedded Workshop = [UseOnChipDebuggerEmulatorN	one.c]				
◇ ファイル(E) 編集(E) 表示(L)	フロジェクト(E) ビルド(E) デバッグ(D) さ	基本設定型 ツール田 テストロ ウィンドウビ ヘルブビ					_ 8 X
Control of the second sec	プロコンパセ ビルド型 デパブロジンパは防薬型 アクライブロジンパは防薬型 フロジンパは家型 フロジンパルドネック 4 パパガロド型 4 オロジンパルドネック 4 オロジンパルドネック 4 アンドイスの作家の 7 フロジンパルドネック 7 アイイルの営業子の 7 フィーネンパギャジリジン 1 1 1	基本就電型 アール① アスト型 ウムアウ型 ヘルプ型 ■ 特 高 高 単 単 の 田 田 丞 [Debog ChisDebuggerEmulatorNone.c Hay 26, 2006 Program file. sted by Renesss Project Generator (Yer.4.8).	DefaulSession	_ } ₽			- 0 ×
typedefine)							
	- - → UseOxChipD. → resetpre.c						1
C VIO Processing "C:VIO processing "C:VIO processing "C:VIO processing "C:VIO processing "C:VIO nov processing "C:VIO nov processing "C:VIO processing "C:VIO processing "C:VIO processing "C:VIO	Han 2 Control (100 bourger Enu) Konace Wilse On Chi 100 bourger Enu Konace Wilse On Chi 100 bourger Enu	at or None VIIse On Chip Debuzzer Emulator None VD at or None VIIse On Chip Debuzzer Emulator None VD ator None VIDse On Chip Debuzzer Emulator None VDse ator None VIDse On Chip Debuzzer Emulator None VIDse Ator None VIDse Non Chip Debuzzer Emulator None VIDse Ator None VIDse None VIDse Ator None VIDse None VIDse Ator None VIDse None VIDse Ator None VIDse Ator None VID	bug¥fvector.r30" bug¥fneap.r30" bug¥initst.r30" bug¥initst.r30" bug¥fusetorg.r30" bug¥fusetor.r30" bug¥fusetor.r30" bug¥ineap.r30" bug¥initset.r30"	zerEmulatorN	one.r30‴		
Duild (Debug) Find	l in Files 1 λ Find in Files 2 λ Macro λ Tes	t λ Version Control /					2
ファイルをプロジェクトに追加します			Default1 desk.top	Read-write	11/15	2 INS	NUM

c) コンパイルオプションを追加します。「ビルド」-「Renesas M16C Standard Toolchain」 を選択してください。

🖗 Use On Chip Debugger Emulator Na	ne - High-performance Embedded Workshop -	[UseOnGhipDebuggerEmulatorNone	s.c)				_ 6 🛛
◇ ファイル(E) 編集(E) 表示(L) ブロジ	エクト (2) ビルド (8) デバック(2) 基本設定(2) ツール(2)	テストシ ウィンドウ心 ヘルプビ					- 8 ×
D 📽 🗟 🖉 🗃 🕺 🛍	Renesas M16C Standard Toolchain	🕸 🖽 📇 Debug	DefaultSession	🖃 🦄 🖗			
Cartoria Construction Constru		orNone.c	UP#ALSetion	***			M
Constant State Constant State	ueOxChipD. ⊴ resetore. 2 2 2 2 2 2 2 2 2 2 2 2	ni pûbbuzger Eaul at or None Yûebu i pûbbuzger Eaul at or None Yîbbu i pûbbuzger Eaul at or None Yîbbu	e¥fvactor-30" e¥hens-70" e¥inityct-30" e¥inityct-30" e¥fvactor.30" e¥fvactor.30" e¥fvactor.30" e¥beas-730 e¥heas-730 e¥inityct-30"	er Emulator No	ne . r 30 ~		
			efault1 desktop	Read-write	11/15	2 INS	NUM

 d) 「Renesas M16C Standard Toolchain」が表示されます。「オプション項目」を「識別子 定義」に変更し、「追加」ボタンを押下してください。

Renesas M16C Standard Toolchair	. ? 🛛
コンフィグレーション: Debug □-ি All Loaded Projects □-ি UseOnChipDebuggerEmulat □-□ C source file ⊡-□ Assembly source file	 ユンパイラ アセンブラ リンカ ライブラリアン ロードモジュールコンバー.▲ ▶ カテゴリ(2): ソース ▼ オブション項目(5): ■E-D]識別!子を定義する(2): Define Value 道加(A) 川除(E)
	コンパイラオブション(1): -c -finfo -dir *\$(CONFIGDIR)* -R8C のK キャンセル

e) 「識別子の定義」画面が表示されます。マクロ名に「__UART__」と記述し「OK」ボタンを押下してください。

識別子の定義	? 🛛
マクロ名(M):	
	OK キャンセル

f) リンカオプションの設定を行います。「Renesas M16C Standard Toolchain」画面で「リン カ」タブを選択し、カテゴリの「セクション配置順序」を選択してください。表1に示す モニタプログラム占有領域の先頭アドレスに「FirmArea_NE」を追加してください。

Renesas M16C Standard Toolchain	? 🛛
コンフィグレーション: Debug ■ ④ All Loaded Projects ■ ⑤ UseOnChipDebuggerEmulat	コンパイラ アセンザ・リンカ ライブラリアン ロードモジュールコンバー・▲ ▶ カテゴツ (Y): (2025) - 2025 - 202
	リンカオプション(I): -L "r8clib" -G -MS -O "\$(CONFIGDIR) ¥\$(PROJECTNAME).x30" -ORDER data_SE=0400,bss_SE,data_SO,bss_SO,data_NE,bss_NE,data_NO,b
	OKキャンセル

以上の設定により、ユーザプログラムがモニタプログラム占有領域と重なった場合はエ ラーが出力されるようになります。

5.4. 周波数特性について

モニタプログラムは、下記に示すメインクロック(Xin)の周波数の範囲内で動作します。 下記以外の周波数については動作しない場合がありますので、この範囲の周波数を持つ発 振子を使用してください。

1MHz (Min.) ~ 20MHz (Max.)

各周波数と通信可能速度を表 3 に示します。ただし、上記の周波数の範囲でも、メインク ロックを分周して 1MHz 未満で使用すると、動作しない場合がありますのでご注意くださ い。また、低速オンチップオシレータ、Xcin クロックはシステムクロックとして使用でき ません。

周波数	通信速度(bps)					
	1200	2400	4800	9600	19200	38400
20MHz	×	×	0	0	0	0
16MHz	×	×	0	0	0	0
14MHz	×	×	0	0	0	0
12MHz	×	×	0	0	0	0
10MHz	×	0	0	0	0	0
8MHz	×	0	0	0	0	0
6MHz	×	0	0	0	0	0
4MHz	0	0	0	0	0	×
2MHz	0	0	0	0	×	×
1MHz	0	0	0	×	×	×

表 3 通信可能速度

〇:通信可

×:通信不可

注:温度、電圧などの条件によっては、通信できない場合があります。その時は通信速度を 下げてご使用ください。

5.5. SFR 操作に関する制限事項

表 4 にレジスタ操作に関する制限事項を示します。なお、変更が禁止されているレジスタ を変更した場合、モニタプログラムは正常に動作しません。

レジスタ名	初期値	制限事項	変更
プロセッサモードレジスタ0	00hに初期化されます。	シングルチップモードのみ	Δ
プロセッサモードレジスタ1	00hに初期化されます。		0
システムクロック制御レジスタ0	08hに初期化されます。	CM05 ビットは "0" にしてください。	Δ
システムクロック制御レジスタ1	28hに初期化されます。	CM13,CM15 ビットは "1"、CM14 ビッ	Δ
		Hは 'D'' にしてください。	
高速オンチップオシレータ	03hに初期化されます。	FRA00 ビットは "1" にしてください。	Δ
制御レジスタ0			
高速オンチップオシレータ		変更しないでください。	×
制御レジスタ1			
高速オンチップオシレータ	03hに初期化されます。		0
制御レジスタ2			
発振停止検出レジスタ	00000x00b に初期化されま	00hにしてください。	Δ
	す。		
プロテクトレジスタ			0
フラグレジスタ		D フラグへの書き込みは無視されます。	Δ
		ユーザプログラムでDフラグを11"に設定	
		しないでください。	
ISP	057Fhに初期化されます。	モニタプログラムが使用しない領域を設	Δ
(割り込みスタックポインタ)		定してください。	

表4 SFR 操作に関する制限事項

○:変更可能 ×:変更禁止 △:変更可能(一部制限あり)

5.6. ストップモード、ウエイトモードに関する制限事項

ユーザプログラム上でストップモード、ウエイトモードを使用する場合、R8C UART デバ ッガはフリーランモードで起動するようにし、あらかじめ、RAM ウインドウ、C ウォッチ ウインドウ、ASM ウインドウを閉じてください。また、ストップモード、ウエイトモード を解除した後にブレークポイントを設定するなどして、ブレークポイントで止まるまで、 R8C UART デバッガの操作をしないようにしてください。

5.7. ウォッチドッグタイマについて

モニタプログラム動作中はウォッチドッグタイマのリフレッシュを行います。ウォッチド ッグタイマを使用したユーザプログラム実行中に、例えばメモリ内容の参照や変更などの 操作によってモニタプログラムが介在した場合、ウォッチドッグタイマがリフレッシュさ れますのでご注意ください。

5.8. ユーザプログラムのリアルタイム性について

Sampling Run (サンプリング) モード
 サンプリングモードでは、Go 実行時および Come 実行時にユーザプログラムの実行状態
 を定期的に監視します。そのため、ブレークなどによるユーザプログラムの停止を検出す
 ることができます。通常のデバッグを行うときに選択してください。

- ・Free Run(フリーラン)モード
- フリーランモードでは、Go 実行時および Come 実行時にユーザプログラムの実行状態を

監視しません。そのため、ユーザプログラムのリアルタイム性は確保されますが、ブレー クなどによるユーザプログラムの停止を検出できません。したがって、ユーザプログラム が停止しても、R8C UART デバッガは Go 実行および Come 実行動作を停止しません。 R8C UART デバッガを停止させるには、STOP ボタンを押してください。

注:フリーランモード時、RAM ウインドウ、C ウォッチウインドウ、ASM ウインドウを 閉じた状態で使用してください。

5.9. 例外的なステップ実行について

・ソフトウエア割り込み命令

ソフトウエア割り込みを発生させる命令(未定義命令、オーバフロー、BRK 命令、INT 命令)の命令内部処理を連続してステップ実行はできません。

例)INT 命令の場合



・INT 命令

INT 命令を用いたプログラムのデバッグは、INT 命令内部処理にソフトウェアブレークを 設定し、Go コマンドと共に使用してください。

例)



5.10. 周辺機能に関する制限事項

以下の端子はホストコンピュータとの通信で使用します。他の端子等と接続しないでくだ さい。

- ・R8C/2K,2L グループ
 - P0_0/AN0 (32pin)、P4_5/INT0 (17pin)

5.11.フラグレジスタに関する制限事項

ユーザプログラム上でフラグレジスタを操作する場合は、デバッグフラグ(Dフラグ)を変 更しないように、<u>FSET命令、FCLR命令</u>で行ってください。

5.12. ブレーク中の周辺 I/O の動作について

ブレーク中、割り込みは受け付けませんが、周辺 I/O は動作し続けます。例えば、タイマを 動作させた後にブレーク等でユーザプログラムを停止させたとき、タイマはカウントし続 けますが、タイマ割り込みは受け付けられません。

改訂記録

Rev.	発行日		改訂内容
		ページ	ポイント
1.00	2008.12.12	-	初版発行

M16C R8C FoUSB/UART デバッガ ユーザーズマニュアル R8C/2K、R8C/2L 接続時の注意事項

- 発行年月日 2008 年 12 月 12 日 Rev.1.00
- 発行 株式会社 ルネサス テクノロジ 営業企画統括部 〒100-0004 東京都千代田区大手町 2-6-2
- 編集 株式会社 ルネサス ソリューションズ

©2008 Renesas Technology Corp. and Renesas Solutions Corp., All Rights Reserved. Printed in Japan.

M16C R8C FoUSB/UART デバッガ ユーザーズマニュアル

