

RL78 ファミリ

R01AN1232JJ0100 Rev.1.00 2012.06.28

CubeSuite+ スタートアップ・ガイド編

要旨

この資料は、CubeSuite+を用いた RL78 ファミリ サンプルコードの活用方法、および RL78 ファミリの開発 ツールの基本的な操作を、ユーザに理解していただくことを目的としています。この資料を読みながら、 実際にツールを操作することにより、サンプルコードの活用方法や開発ツールの基本的な操作に対する理 解を、より深めていただけます。

この資料では, RL78/G13のサンプルコードを用いて説明します。

対象デバイス

RL78/G13

目次

1.	はじめに	2
2.	開発環境について	2
3.	オブジェクト・モジュール・ファイルの生成	4
4.	E1 エミュレータを使用したオンチップ・デバッグ	. 11
5.	付録 コード生成	. 16



1. はじめに

対象者

この資料は、サンプルコードを初めて活用されるお客様、および RL78 ファミリの開発ツールを初めて使用されるお客様を対象としています。なお、開発ツールを使用する際に、Windows®の操作方法に関する 基礎知識を必要とします。

2. 開発環境について

RL78 ファミリを利用する場合の開発手順と開発ツールを説明します。

2.1 開発手順と開発ツールの関係

製品の開発手順と開発ツールについて、次の図に示します。



図 2-1 開発手順



2.2 使用する開発ツール

サンプルコードの動作確認をするためには、次の開発ツールを使用します。

2.2.1 CubeSuite+(統合開発環境)

CubeSuite+は、Windows上で使用する統合開発環境です。エディタ、コンパイラ、デバッガなどの開発ツールと連携し、効率的な開発ができます。おもに、次のものが用意されています。

a. デバイス依存情報ファイル デバイス固有の情報が入ったファイルです。

b. CA78K0R (C コンパイラ)

RL78 ファミリ用の組み込み制御プログラムを C 言語で記述するために開発された汎用性,移植性の高い C コンパイラです。CA78KOR を Windows 上で使用するためには, CubeSuite+が必要です。

- c. デバッグ・ツール
 - RL78 E1(Serial) E1 エミュレータを使用したオンチップ・デバッグを行います。
 - RL78 IECUBE

フルスペック・エミュレータ IECUBE を使用したデバッグを行います。

- CubeSuite+ RL78 シミュレータ
 RL78 ファミリ用の命令シミュレータです。
 ホスト PC 上で、ソース・コードと TAU CH0~CH3 のインターバル・タイマ機能の動作をシミュレートできます。
- d. 設計ツール
 - 端子配置

端子配置図から GUI 操作で兼用端子の設定をする機能です。設計書への端子配置図の貼り付けや,端子配置の EXCEL 出力が可能です。

● コード生成

マイコン周辺機能(タイマ,シリアル I/F, A/D など)を制御するデバイスドライバプログラムを, GUI 設定により自動生成できる機能です。

- 2.2.2 代表的な CPU 評価ボード
 - CPU ボード (例 QB-R5F100LE-TB (RL78/G13_64 ピン/ROM64KB) など)
 - Renesas Starter Kit (例 Renesas Starter Kit for RL78/G13 など)
 - RL78G13 Stick スターターキット (RL78G13-STICK)

3. オブジェクト・モジュール・ファイルの生成

サンプルコードからオブジェクト・モジュール・ファイル (HEX, LMF ファイル) を生成するために, CubeSuite+を用いてサンプルコードをビルドします。

この章では、サンプルコードのプロジェクト作成からビルドまでの動作例を説明します。

CubeSuite+のプロジェクト作成方法の詳細については、「CubeSuite+統合開発環境 ユーザーズマニュアル 起動編 (R20UT0975)」を参照してください。

- 注意 RL78 ファミリのサンプルコードでは、プロジェクトで最初に指定したデバイス品名を変更すること はできません。そのため、プロジェクトのデバイス品名を変更する場合は新規にプロジェクトを作成 してください。
- 備考1 ダウンロードしたサンプルコードのプロジェクトをそのまま使用する場合は,"ダウンロードした ファイル名+拡張子[*.mtpj]"のファイルをダブルクリックして CubeSuite+のプロジェクトを開いて ください。その場合は,"3.2 オプション・バイトの設定"からの手順で実施してください。
 - 2 LMF (ロード・モジュール) ファイルとは、シンボル等のデバッグ情報を含むオブジェクト・ファ イルです。
 - 3 HEX (ヘキサ)ファイルとは、LMFファイルから生成したデバッグ情報を含まない HEX 形式の オブジェクト・ファイルです。



3.1 新規プロジェクトの作成とサンプルコードの読み出し

3.1.1 サンプルコードの準備

ダウンロードしたファイルを解凍し、任意の場所(フォルダ)に保存してください。

3.1.2 CubeSuite+の起動と新規プロジェクトの作成準備

CubeSuite+を起動し, ツールバーの ³ ⁽³⁾ ^{スタート(2)} をクリックすると, スタートパネルがオープンします。ス タートパネルがオープンしたら, 図 3-1 に示す "新しいプロジェクトを作成する"の GO ボタンをクリッ クし, 新規にプロジェクトを作成します。



図 3-1 CubeSuite+ のスタートパネル



3.1.3 プロジェクトの指定

図 3-2 に示す"プロジェクト作成ダイアログ"画面が表示されたら、次の設定を行います。

プロジェクトで使用するデバイス品名を選択します。注

プロジェクト名とプロジェクト・ファイルの作成場所を指定します。

(ただし,指定した作成場所の下にプロジェクト名のフォルダを作成しない場合は,[プロジェクト名のフォルダを作成する(<u>A)</u>]のチェックを外してください。)

[既存のプロジェクトのファイル構成を流用する(<u>S</u>)]をチェックし,[流用元のプロジェクト(<u>P</u>)]に流 用元のサンプルコードのプロジェクト・ファイル名を指定します。ここで指定することで,流用もと のソース・ファイルが自動的に読み込まれます。

また,指定した流用元プロジェクトのファイル構成をコピーして使用する場合は,[プロジェクト・フォルダ以下の構成ファイルをコピーして流用する(<u>O</u>)]にチェックをしてください。

[作成(<u>C</u>)]ボタンをクリックして、プロジェクトを作成します。

【注】使用する品種名が選択欄にない場合、デバイス依存情報ファイルをインストールしてください。

プロジェクト作成)	
マイクロコントローラ(工):	RL78		~		
使用するマイクロコントローラ	(<u>M</u>):		使用デバイスを選択		
(マイクロコントローラを検	索できます)	アップテート(山)…	Com 7 ** 1 ** Caddy		
 ■ R5F101 JE(52pin) ■ R5F100LE(64pin) ■ R5F101LE(64pin) ■ R5F101LE(64pin) ■ R178/G13 (ROM:96KB) ■ R178/G13 (ROM:128KB) 					
 ・● ● RL78/G13 (ROM:192KB) ・● ● ■ RL78/G13 (ROM:256KB) ・● ● ■ RL78/G13 (ROM:384KB) ・● ● ■ RL78/G13 (ROM:384KB) ・● ・● ● ■ RL78/G13 (ROM:384KB) ・● ・● ・●					
プロジェクトの種類(<u>k)</u> : アプリケーション(CA78K0R)					
プロジェクト名(<u>N</u>):	名(N): sample_project				
作成場所(L):	C:¥Program Files¥	Renesas Electronics	▼ 参照(<u>R</u>)		
	🗹 プロジェクト名の	フォルダを作成する(<u>A</u>)			
C*Program Files*Renesas Electronics*sample_project*sample_project.mtpj ダウンロードした プロジェクトのファイル構成を流用する(S)					
流用元のブロジェクト(P): (流用元のブロジェクト・ファイルを入力して(ださい) ▼ 参照(W)					
プロジェクト・フォルダ以下の構成ファイルをコピーして流用する(①) 作成をクリック					
8	(°E)	成(0) キャンセル			

図 3-2 プロジェクト作成ダイアログ



3.2 オプション・バイトの設定

オプション・バイトは,ユーザ・オプション・バイトとオンチップ・デバッグ・オプション・バイトで構成されています。電源投入時またはリセット起動時に,マイコンは自動的にオプション・バイトの設定値を 参照して,動作を開始します。

ここでは、RL78/G13の設定を例に挙げて、CubeSuite+の[CA78K0R (ビルド・ツール)]のプロパティから、 オプション・バイトを設定する手順を説明します。

3.2.1 オンチップ・デバッグ・オプション・バイトの設定

オンチップ・デバッグ・オプション・バイトを設定することにより、マイコンの持つオンチップ・デバッ グ機能を有効にします。

[CA78K0R のプロパティ]画面が表示されたら、画面下の[リンク・オプション]タグを選択します。

[デバイス]カテゴリを開き,[オンチップ・デバッグを設定する] プロパティで [はい(-go)] を選択すると, [オンチップ・デバッグ・オプション・バイト制御値] プロパティ, [デバッグ・モニタ領域開始アドレス] プロパティ, [デバッグ・モニタ領域サイズ[バイト]] プロパティが表示されます。

[オンチップ・デバッグ・オプション・バイト制御値] プロパティ:

オンチップ・デバッグ・オプション・バイトの制御値を 0x なしの 16 進数で指定します。

(RL78/G13の場合,オンチップ・デバッグ動作禁止(0x04),オンチップ・デバッグ動作許可(0x84/0x85) を設定します)

[デバッグ・モニタ領域開始アドレス] プロパティ:

デバッグ・モニタ領域の開始アドレスを 0x なしの 16 進数で指定します。

(デフォルト:内部 ROM のエンド・アドレス-1024 +1)。

[デバッグ・モニタ領域サイズ[バイト]] プロパティ:

デバッグ・モニタ領域のサイズを10進数で指定します。指定可能な値の範囲は、0~1024です(デフォルト:512)。



3.2.2 ユーザ・オプション・バイトの設定

ユーザ・オプション・バイトを設定することにより、ウォッチドッグ・タイマ(WDT)の設定、電圧検出回路(LVD)の設定、システム予約領域の設定を行います。

ユーザ・オプション・バイトの設定も、[リンク・オプション]タブの[デバイス]カテゴリで行います。 [ユーザ・オプション・バイトを設定する]プロパティで[はい(-gb)]を選択すると、[ユーザ・オプション・ バイト値]プロパティが表示されます。

[ユーザ・オプション・バイト値] プロパティ:

ユーザ・オプション・バイト値を0xなしの16 進数で指定します。(RL78/G13の場合, WDT停止(0xEF), LVD リセットモード 2.81V (0x7F),動作クロック f_{IH}=32MHz(HS モード)(0xE8)を設定します)

- 備考 オプション・バイトの設定方法についての詳細は、「CubeSuite+統合開発環境 ユーザーズマニュア ル RL78,78K0R ビルド編 (R20UT0730)」を参照してください。
- 注意 オプション・バイトは、リンク・オプションで設定する方法以外に、ソース・コード上でプログラミ ングする方法があります。なお、同時にオプション・バイトを設定した場合は、リンク・オプション で設定した値が有効になります。



3.3 オブジェクト・モジュール・ファイルの生成(ビルド)

サンプルコードからオブジェクト・モジュール・ファイル (HEX, LMF ファイル) を生成するために, CubeSuite+を用いてサンプルコードをビルドします。

ビルドには、次の種類があります。

種類	説明
ビルド	ビルド対象ファイルのうち、更新されたファイルのみビルドを実行します。
リビルド	ビルド対象のすべてのファイルのビルドを実行します。
ラピッド・ビルド	ビルド設定の変更と平行してビルドを実行します。
パッチ・ビルド	プロジェクトが持つビルド・モードを一括してビルドを実行します。

備考 ビルド方法の詳細は、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78,78K0Rビルド編 (R20UT0730)」を参照してください。

ここでは、"ビルド"を例に挙げて、HEXファイルの作成手順を説明します。

3.3.1 HEX ファイルの指定

プロジェクトを開いたら、次の手順で HEX ファイル名を設定します。

左側の"プロジェクト・ツリー"にある[CA78K0R(ビルド・ツール)](図 3-3 中に赤い下線で示す)を ダブルクリックして,[CA78K0Rのプロパティ]画面を表示させます。

[CA78K0R のプロパティ]画面が表示されたら,画面下の[オブジェクト・コンバート・オプション]タ グを選択します。

[ヘキサ・ファイル]カテゴリの[ヘキサ・ファイル名]プロパティに「任意のファイル名+拡張子(.hex)」 を入力します。(デフォルト設定は、プロジェクト・ファイル名と同じファイル名になっています)



図 3-3 オブジェクト・コンバータ・オプション画面



3.3.2 ビルドの実行

図 3-4 中の枠で示す 「「ボタンをクリックすることで、ビルドを実行します。ビルドを実行すると、ソー ス・ファイルから HEX ファイル(*.hex)が生成され、プロジェクト・ファイルのフォルダ内に格納されま す。ビルドが終了すると、CubeSuite+画面の出力画面に[ビルド終了(エラー:0個,警告:0個)]のメッセー ジが表示されます。

			ここを? ビルド	フリックして, を開始します				
🔘 sample_project - CubeSuite+ -	[ナロパティ]		7/					_ 7 ×
ファイル(E) 編集(E) 表示(V) プロジェクト(P)	ビルド(<u>B</u>) デバッグ(<u>D</u>) ツール	いの ウインドウ(型) ヘルプ(4	Ð					
: 🖚 スタート(S) 🚚 📄 🍠 : 💥 📭 ([™] ∂ 6 ₩ ♣ ♣	. 💌 🤅	🖬 🚮 📐 🖏 🖏 ካ	🖲 🕟 🗠 🎙 🕫 Ç	≡ °≡ ¥a			
I 🖓 🖓 🖉 🖉 🖉 🧐 🤻		_	プロジェクトをビルドします。(F	0				
プロジェクト・ツリー 🛛 🗛 🗙	🕋 プロパティ 🐃 コード タ	E成 🧖コード生成プレビュ	-	<u> </u>				- x
9 0 8	🔨 CA78KOR のプロパティ							
 ■ Sample project (プロジェクト)* ■ RSF100LE (マイクロン・トローラ) ■ フトギ成 (戦計ツール) ■ クロック発生回路 ■ オート ■ 割り込み ● ジリアル ■ イクロンパータ ● クロックパレ・タイマ ● クロックルレ・タイマ ● クロックルレ・タイマ ● クロックルレ・タイマ ● クロックルレ・タイマ ● ロックルレ・タイマ ● ロペコン・ローラ ■ CATBURD (ビルド・ツール) ● アイル ● ログラム解析 保術ツール) ● ゴッケイル 	日 ヘキサ・ファイル ヘキサ・ファイルと出力す。 ヘキサ・ファイルと力フォ) ヘキサ・ファイルシカーマ。 ヘキサ・ファイルクライルラフィーマ。 ヘキサ・ファイルを了へ ロート マイルファイルを了へ ロート マイルマイルマート マイルマイルマート マイル マイルマート マイルマート マイルマート マイル マイルマート マイル マイルマート マイル マイル マイルマート マイル マイルマート マイル マイルマート マイル マイル マート マイルマート マーマ マイルマート マイル マート マイルマート マート マイルマート マーマ マイルマート マート マート マート マート マート マート マート マート マート	5 1,5 3 3 3 3 5 5 5		はい 米BuildModeName SAMPLE_PRO インデンは反義へキャ しいえ はいへい 電子 下 電子	5% 3RAM.hex 1·7∂-マγ⊦(-kie)			
 ● 10 CUF*9=D/主成フォル ● スタートアップ ● フード主成 ● r_systeminitc ● r_systeminitc ● r_systeminitc ● r_systeminitc 	ヘキサ・ファイル名 ヘキサ・ファイルのファイル名 ・コブジョンに相当します。 主に次のブレースホルダにす 共通オブション / コンパ 出力	を指定します。拡張子は、自 1応しています。… イル・オプション / アセンブ	由に指定可能です。 ル・オブション / リンク・オブショ	出力画面 表示され	面に[ビルド れます	終了]のメッ	ッセージが	<u>• / زون</u>
<pre></pre>	10403023:先しご端子割り] ビルドの用 ビルド終了(エラ 終了しました) [EOF]	当てを設定してくたさい。 取合(2012年3月22日 14:28 取合(2012年3月22日 14:28 	(48) ========」 -」 -」 実験:0プロジェクト)(2012年	3月22日 14:26:52)	با			▲
		- 1011-00 / 00010	te faith to te to te f					~
	<u>↓すべてのメッセージ **</u>	コード主成 / *ラビッド・ビル	レド <u>/ *ビルド・ツール</u> /	50				•
F1 F2 F3	F4	FS	Fi Fi	FB	F9	FHB	FTI	H2
Manager 1 - Constraints and the second se								▲ 非接続

図 3-4 ビルドの実行画面

備考 HEX ファイル (*.hex) は、ルネサス製の El エミュレータとフラッシュ書き込みソフトウエア Renesas Flash Programmer (RFP)を使用することで、マイコンのフラッシュメモリヘプログラムを書き込むこと ができます。フラッシュ書き込みの詳細は、「Renesas Flash Programmer フラッシュ書き込みソフトウ エア ユーザーズマニュアル (R20UT0599)」を参照してください。



4. E1 エミュレータを使用したオンチップ・デバッグ

CubeSuite+ が提供するデバッグ機能を使用することにより, RL78 ファミリ用に開発されたプログラムを, 効率良くデバッグすることができます。

ここでは、ルネサス製 El エミュレータ を使用したオンチップ・デバッグの設定例を説明します。CubeSuite+ が提供するデバッグ機能の詳細については、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78 デ バッグ編 (R20UT0978)」を参照してください。

4.1 デバッグを始める前の準備

図 4-1 に示すように、ホスト・マシン、El エミュレータ、および必要に応じてターゲット・ボードを接続 します。接続方法についての詳細は、「E1/E20 エミュレータ ユーザーズマニュアル 別冊 RL78 接続時の注 意事項 (R20UT1994)」を参照してください。



【注】 ターゲット・システムとの通信方式として、シリアル通信のみサポートしています(JTAG 通信は使用不可)。

図 4-1 ホスト・マシンとデバッグ・ツールとの接続例【E1】



4.2 オンチップ・デバッグの設定手順

4.2.1 デバッグ・ツールの選択/切り替え

プロジェクト内で使用するデバッグ・ツールを選択します。使用するデバッグ・ツールの選択/切り替えは、プロジェクト・ツリーパネル上の [RL78 デバッグ・ツール名(デバッグ・ツール)] ノードを右クリック することで表示されるコンテキスト・メニューにより行います(図 4-2 参照)。

🔯 sample_project - GubeSuite+ -	[プロジェクト・ツリー]		_ 7 🔀
ファイル(E) 編集(E) 表示(V) ブロジェクト(P)	ビルド(1) デバッグ(1) ツール(1) ウインドウ(W へルブ(L)	
: 🙉 スタート(S) 📮 🖂 🗦 🗶 🐚	高しつで「品単単		
A 0.0		主政ノレビュー	• ×
2 🛛 🞽	AL 18 512 - 9 0070/171		
□ R sample project (プロジェクト)*		64	
— 🖲 R5F100LE (マイクロコントローラ)	内部20/00/01/20/20/20	04 2006	
- 🔄 コード生成 (設計ツール)	E bnab	1000	
🥑 クロック発生回路	メイン・クロック問題時[MHz]	400	
k	サブ・クロック店!波数[kH2]	32,768	
S \$1113.3	タイマ/トレース用クロック周波数の選択	CPU5回ック周辺波数	
	タイマ/トレース用りロック周波数の単位	MHz	
	タイマ/トレース用クロック店に波数		
- A/U_J/\-9	日 コンフィキュレーション		
	シミュレータ・コンフィギュレーション・ファイル	を使用する いいえ	
──● ウォッチドッグ・タイマ			
── リアルタイム・クロック			
―● クロック出力/ブザー出力			
- C DMAコントローラ	ここを右クリック		
■ 香圧検出回路			
へ CA78K0B (どは長・ツーは)			
	1		
◆ オロド= 1 4215 g 使用するデ/	、ッグ・ツール(D) ▶ RL78 IECUBEΦ		
	PL78 E1(Seria)(())		
+ T FUE-9-0+03 7710			
	内部ROMのサイ:	y and the second s	
□ □ □ - ド生成	RL70 EZ Emulator	② 使用するデバッグ・ツール[RL78 E1(Serial)]を	
-Sir_main.c	■ RL78 シミュレータ(S)	クリックレイ 切り巻えます シー・シー・シー・シー・シー・シー・シー・シー・シー・シー・シー・シー・シー・シ	
-9 r_systeminit.c	TREETIER AE		
- ego.c	出力		4 X
- a r_cec_user.c	プロジェクトの流用作成を完了しました	t。_〈コピー成功:11ファイル〉」	A
- r porto	190403029:先に端子割り当て参設定して	くださし、 」	
C r port user c	ビルド開始(sample project, D	2(
	ビルド終了(エラー:0個, 警告:	0(0)	
	======= 終了しました(成功:1ブロミ	フェクト, 失敗:0プロジェクト)(2012年3月22日 14:26:52) =======_。]	
r_cg_userdefine.n	TEOF1		
r_cg_cgc.h			
r_eg_porth			M
i — ≝i r_ik.dr	オペてのメッセージ / *コード生成 / *	ラビッド・ビルド / *ビルド・ツール /	-
日ブロジェクト・ツー・日名前の変更 F32	た検索 円次を置換 戸5実	行 FEEルド&デバッ・・・・ FIEリレークせずに・・・ FIEブレークの酸・・・ FIEステップ・オー・・・ FIEステップ・イン	日に開設へジャンプ
デバッグ・ツールにシリアル接続したE1を使用しま?	f.		<mark>⊷</mark> 非接続

図 4-2 デバッグ・ツールの選択/切り替え【E1】



4.2.2 オンチップ・デバッグの動作環境の設定

E1 エミュレータを使用する場合の動作環境の設定を図 4-3 のプロパティパネルで行います。ここでは、 "ターゲット・ボードと接続"の設定を行います。その他項目の設定については、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78 デバッグ編 (R20UT0978)」を参照してください。

日 5 ボッ・ボード 通信方式 エミュレータから電 供給電圧	との接続 配源供給をする(最大	(200 mA)				1線式(TOOL はい 5.0V	0)		
🕼 sample_project - CubeSuite+ -	[プロパティ]								X
ファイル(E) 編集(E) 表示(V) プロジェクト(P)	ビルド(<u>B</u>) デバッグ(<u>D</u>) ツール	① ウインドウ(W) ^	ルプ(円)						
፤ 🚳 スタート(S) 🚚 💽 🗿 ፤ 💥 🐚 ו	B O O C A A A A A A A A A A A A A		- : 🖓 🖓 🔬 :	🚮 🕒 🧠 🗐	(►) [►] [SE (je če 👗			
4 ×	- 11 プロパティ 11 コード生	1成 愛コード生成フ	10E1-						- x
2 @ 🙎	▲ RL78 E1 (Serial) のプ[]パティ							
😑 <u> sample project (ታበジェクト)*</u>					64				
	内部RAMサイズビソト				4096				\
□***********************************	a the property state the	4.7.EK/14 FI			4				•
		u-1			内蔵カロッカを使	田する			
	サブ・クロック周辺変数[kHz]			内蔵クロックを使	用する			1
	モニタ・カロッカ				システル				
	日ターケット・ホードとの接着	R.			1線式(10010)				
	エミュレータから電源供給	をする(最大200 mA)			(tr)				
	供給電圧				5.0V				
リアルタイム・クロック	tetral and				0000000	000000000000000000000000000000000000000			
インターバル・タイマ	セキュリティル フラッシュ まき換えを許可	する				000000000000000000000000000000000000000			
	ワイド・ボルテージ・モード	を使用する			はい				
	起動時にフラッシュ ROM ³	を消去する			いいえ				
■ 電圧検出回路									
「「「「「「」」」」									
□-□									
	ターケット・ホードとの接続								
I main.c				/	,				
🔤 r_systeminit.c	★ 接続用設定 / デバッグ	・ツール設定 / ダウ	ンロード・ファイル設定	人フック処理設定	/				•
	出力								д X
🚰 r_cgc_user.c	ブロジェクトの流用作成	を完了しました。(コピー成功:11ファ	(JL) <mark>1</mark>					^
		12012年3月22日	14:26:48) =======	ل,=:					
San Laseric	ビルド開始(samp ビルド終ア(エラ	le_project, Defaul ~:0個、警告:0個)	له (tBuild						
r_cg_macrodriver.h	・※?しました	- (成功:1ラロジェク	ト, 失敗:0ブロジェ	クト)(2012年3月22	日 14:26:52) =====	ل,=====			
r_cg_userdefine.h	EOF]								
r_cg_cgc.n									
t k dr		- to the -the distance -		. /					~
	すべてのメッセージ *]−ド生成 / *ラピッ	ריאוניצי א אראי	<u>-n</u>					•
F1 F2 F3	F4	F5	FG	F7	FB	F9	FHI	FN	FR2
									▲非接続

図 4-3 動作環境設定【E1】 (プロパティパネル)

a. 通信方式

E1 エミュレータ がターゲット・システム上の RL78/G13 とシリアル通信を行う際の通信方式を指定します。1 線式(TOOL0) を選択してください。

b. エミュレータから電源供給をする(最大 200mA)

E1 エミュレータからターゲット・システムへ電源を供給するか否かを指定します。電源を供給する場合は [はい]を選択してください(デフォルトでは [いいえ] が指定されます)。なお、エミュレータから電源供給しない場合は、[いいえ]を選択して、外部から電源を供給してください。

c. 供給電源

このプロパティは、 [エミュレータから電源共有をする(最大 200mA)] プロパティにおいて、 [はい] を指定した場合のみ表示されます。ターゲット・システムへ供給する電圧を指定します。

例:5.0V ・・・ターゲット・システムへ 5.0V の電圧を供給します。

【注】E1 エミュレータと接続中にこのカテゴリ内のプロパティを変更することはできません。

4.2.3 CubeSuite+にデバッグ・ツールを接続する

デバッグ・ツールバーの 🎑 ボタンをクリックすることで, デバッグ・ツールと接続したのち, LMF ファ イルのダウンロードを実行します。



図 4-4 デバッグ・ツールへのダウンロード例



4.2.4 CubeSuite+からデバッグ・ツールを切断する

デバッグ・ツールバーの が ボタンをクリックすることにより,現在接続しているデバッグ・ツールとの通信を切断します。デバッグ・ツールと切断すると,メイン・ウインドウのステータスバーが,図4-5のように変化します。



図 4-5 デバッグ・ツールから切断したステータスバー



5. 付録 コード生成

5.1 コード生成の使用方法

CubeSuite+のコード生成は, RL78 ファミリが提供している周辺機能(クロック発生回路の機能, ポートの 機能など)を制御する際に必要な情報を CubeSuite+ のパネル上で設定することにより, 対応するソース・コー ド (デバイス・ドライバ・プログラム)を出力します。

ここでは、コード生成の操作手順を説明します。コード生成の詳細は、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78 設計編 (R20UT0976)」を参照してください。

5.1.1 コード生成パネルでの設定

使用する周辺機能(クロック発生回路の機能,ポートの機能など)のコード生成パネルをオープンして, 必要な情報を設定します。



図 5-1 コード生成パネルの設定例

備考 コード生成パネルで入力間違いや必須項目の未入力,設定する端子が他の機能で既に使われているな どの場合に,情報として誤っていることを示す **①** アイコンを該当箇所に表示します。 **①** アイコ ン上にマウス・カーソルを移動した際には,これらの警告に関する情報(解決するためのヒント)が ポップアップ表示されます。

5.1.2 ソース・コードの確認

[コード生成プレビュー]を選択してオープンしたコード生成プレビューパネルで,コード生成パネル上で 設定した情報に応じたソース・コードを確認できます。



5.1.3 ソース・コードの出力

コード生成パネルの[コード生成(G)] ボタンをクリックし、ソース・コードを出力します。



図 5-2 ソース・コードの出力例



5.2 コード生成が出力するソース・コード規則

5.2.1 ソース・コードの構成

以下に、コード生成が出力するファイルの一覧を示します。

衣 5.1 コート生成の山力ノアイル一覧	表:	5.1 ⊐	- ド生成の出力ファイル-	-覧
----------------------	----	-------	---------------	----

出力単位	ファイル名	出力内容
各周辺機能	r_cg_ <i>周辺機能名</i> .c	初期化関数,API 関数 ^注
	r_cg_ <i>周辺機能名</i> _user.c	割り込み関数,コールバック関数
	r_cg_ <i>周辺機能名</i> .h	レジスタへの代入値マクロを定義
プロジェクト	r_main.c	main 関数
	r_systeminit.c	各周辺機能の初期化関数コール
		R_CGC_Get_ResetSource のコール
	r_cg_macrodriver.h	全ソース・ファイルで共通使用する
		マクロを定義
	r_cg_userdefine.h	空ファイル(ユーザ定義用)
	r_lk.dr	リンク・ディレクティブ

 [【]注】 API 関数の詳細は、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78 設計編 (R20UT0976)」を参照してください。

5.2.2 ソース・コード規則

<リスト省 /**********	略> ************************************	*****	*****	
Exported glo	bal variables and functions (t	to be accessed by other files) ************************************	******	
/* Start user	code for global. Do not edi	it comment generated here	この間にグロー	·バル変数を定義できます
uint16_t	result_buffer;	/* AD converter result buff	er */	
/* End user of /************************************	code. Do not edit comment ************************************	generated here */ ***********************************	****	
******	*******************************	***************************************	***************	· · · · · · · · · · · · · · ·
void main(vo	oid)		の間にユーザ・ブロ	コグラムを記述できます
/* Start user uint8_t count	code. Do not edit comment ;;	t generated here */ /* Loop counter */		
result_buffer	= 0U;	/* Initialize result buffer */		
DI();		/* Disable interrupt */		
R_ADC_Set_	_OperationOn();	/* Enable comparator opera	tion */	
/* stabiliz for (count=0)	zation wait time(about 1us) - U; count<3U; count+1)	*/		
{ NOP();		API 関数 [r_adc.c 参照]		
} R_ADC_Star	rt();	/* Start AD converter */		
<リスト省 }	略>			
/* End user (}	code. Do not edit comment	generated here */		
備考 これら	コメント文の文字列は変更	更しないでください。		



ここでは、" サンプルコード A/D コンバータ編"のソース・コードを例に挙げて命名規則を説明します。 main.c

r_adc.c

<リスト省略> void R_ADC_Create(void) { ADCEN = 1U: /* supply AD c	— 2 進定数 lock */		
<リスト省略> /* Set ANI0 pin as analog input */ PM2 - 0x0111:	\square	レジスタ(ADM0 レジスタ) に代入する値は, 下線で示した Byte の総和(OR)が入ります	
ADM0 = _28_AD_CONVERSION_C ADM1 = _00_AD_TRIGGER_SOFT <リスト省略>	LOCK_5 _00 AD_TIM WARE _00_AD_CONV	IE_MODE_NORMAL_1 _00_AD_OPERMODE_S 'MODE_CONSELECT;	ELECT;
/*************************************	*******	***********	
* Function Name: R_ADC_Start			
* Description : This function starts the AD co	onverter.		
* Arguments : none * Beturn Value : none			
**************************************	****	*****	
void R ADC Start(void)			
{			
ADIF = $0\mathbf{U}$;	/* clear INTAD in	terrupt flag */	
ADMK = 0U;	/* enable INTAD	interrupt */	
ADCS = 1U;	/* enable AD conv	rersion */	
} /************	*****	******	
End of function R_ADC_Start			
***************************************	*****	**********************/	

備考 これらコメント文の文字列は変更しないでください。

a. グローバル変数の定義

グローバル変数の定義は、 "/* Start user code for global. Do not edit comment generated here */"以降に記述してください。なお、"/* End user code. Do not edit comment generated here */"までの範囲は、再度コード生成を行っても書き換わることはありません。

- 注意 この領域以外でグローバル変数定義した場合,再度コード生成を行った際に, CubeSuite+によっ て上書き消去されます。
- b. ユーザ・プログラムの記述

ユーザ・プログラムは, "/* Start user code. Do not edit comment generated here */"以降に記述してくだ さい。なお, "/* End user code. Do not edit comment generated here */"までの範囲は, 再度コード生成を 行っても書き換わることはありません。

注意 この領域以外でユーザ・プログラムを記述した場合,再度コード生成を行った際に, CubeSuite+ によって上書き消去されます。

c. API 関数

API 関数は、ユーザ・プログラムでコールして使用してください。それにより、コード生成で出力した 各周辺機能のレジスタ設定や動作処理を行います。

API 関数の詳細は、「CubeSuite+統合開発環境 ユーザーズマニュアル RL78 設計編 (R20UT0976)」 を参照してください。



d. 2 進定数

ビット・レジスタに直接記述する場合,コード生成のソース・コードは2進定数で記述します。

表 5.1 2 進定数の型 一覧

添字なし2 進数	int, unsigned int, long int, unsigned long int
u, または U の添字付き	unsigned int, unsigned long int
I, またはL の添字付き	long int, unsigned long int
u, または U の添字, および I,	unsigned long int
またはL の添字付き	

e. レジスタ設定

コード生成で出力したレジスタは、"r_cg_ *周辺機能名*.h"に自動的に定義した変数の総和(OR)を代入して設定しています。



ホームページとサポート窓口

ルネサス エレクトロニクスホームページ http://japan.renesas.com/

お間合せ先

http://japan.renesas.com/contact/

すべての商標および登録商標は、それぞれの所有者に帰属します。



改訂記録

			改訂内容
Rev.	発行日	ページ	ポイント
1.00	2012.06.28	_	初版発行

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意 事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の 記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子 を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れ たり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」 で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。 外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の 状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのか かる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのア ドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてくださ い。

4. クロックについて

【注意】リセット時は,クロックが安定した後,リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。 リセット時、外部発振子(または外部発振回路)を用いたクロックで動作を開始するシステムでは、ク ロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子(また は外部発振回路)を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切 り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性 が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実 施してください。

		ご注意書き
	1.	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
l		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。
l	3.	当社製品を改造、改変、複製等しないでください。
l	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。
l	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導守し、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
l	•	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。
l	6.	本資料に記載されている情報は、止催を期すたの慎重に作成したものですか、誤りかないことを保証するものではありません。カー、本資料に記載されている情報
l	7	の誤りに起因する損害かね各様に生した場合においても、当在は、一切ての頁仕を見いません。 ************************************
l	1.	当社は、当社製品の面具水準を「標準水準」、「高面具水準」のよび「存足水準」に分類しております。また、各面具水準は、以下に示す用述に製品が使われるこ したき回しておりますので、平特制日の日気水準まで確認ください、お気だけ、平特の文書にして更新の姿態を得てこしたく、「結束水準」に八粒された円冷に
l		こを思因しておりよりので、当社製品の加具小学をご確認ください。お各様は、当社の文書による事則の単語を得ることなく、「特定小学」に力強された出述に ※注制日本店用することができません。また、む安洋は、※注の文書にとる事前の承諾を得ることなく、音励されていたい田冷に※注制日本店用することができ
l		当社装加を使用することができません。また、の各様は、当社の大音による争削の本語を持ることなく、急凶されていない用述に当社装加を使用することができません。当社の文書に上る事前の承諾を得ることなく、「特定水準」に公頼された田途またけ音励されていたい田途に当社制品を施用したことに上しな安様また
l		よどの。当社の大吉による事前の承認を行ることない、「付た小牛」に力規された市座よだは急凶されていない市座に当社装配を使用したことによりお各様よた は第二者にたじた指定等に期」 当社は 「一柄るの志広な色いません」なな、当社制日のデータ・シート データ・ゴック等の姿妙で特に日度水準のまこがない
l		は第二日に上した頃日中に因し、当社は、 切じの員はを良いよど心。ない、当社表面の アーク・アード、アーク・アファキの員件で特に面積水牛の衣水がない 提合け 一連進水進制品であることを表します
l		海口は、赤牛小牛衣仙でのもこことなじます。 煙進水準・ コンピュータ DA雄哭 通信機哭 計測機哭 AV機哭 安索 工作機械 パーソナル機哭 産業田口ボット
l		高品質水準·輸送機器(自動車、雷車、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
l		
l		37-2222-2222-2222-222-222-222-222-222-22
l		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
l	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
l		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。
l	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
l		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
l		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
l		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。
l	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
l		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
l		て、当社は、一切その責任を負いません。
l	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	l. 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	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。
1		

RENESAS

ルネサスエレクトロニクス株式会社

http://www.renesas.com

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2(日本ビル)

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。 切2-6-2 (日本ビル) (03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。 総合お問合せ窓口: http://japan.renesas.com/contact/

■営業お問合せ窓口