

RX ファミリ

R01AN1723JU0121 Rev.1.21 2017.6.30

e² studio に組み込む方法

Firmware Integration Technology

要旨

本アプリケーションノートでは、Firmware Integration Technology (以下 FIT と称す) モジュールを入手し、 e^2 studio のプロジェクトに追加するまでの手順について説明します。

対象デバイス

サポートしているデバイスは、FIT モジュールによって異なります。

FIT モジュール内の Readme.txt ファイルにサポートするデバイスリストがありますので、こちらでご確認 ください。



目次

1. 概要	
1.1 用語説明	3
1.2 FIT の構成	
1.3 FIT モジュールアプリケーションノートのファイル構造	5
	-
2. FIT コンフィギュレータを使用して FIT モジュールを組み込む方法	7
2.1 e ² studio プロジェクトの作成	
211 新規プロジェクトの作成	8
212 FIT モジュールのダウンロード	12
22 FIT モジュールの組み込み	15
221 FIT コンフィギュレータの記動	15
222 FIT コンフィギュレータの初期設定	16
222 モジュールの選択と詳細なフィルタの使用	17
2.2.0 ビノユ ルの送水と中間なノイルグの及方	18
2.2.4 パープコンの医療	
2.2.0 TH C ノユ ルの送バ	10
2.2.0 端于設定	
2.2.7 コードの主人	
$2.5 J \square \mathcal{Y} \bot \mathcal{Y} \vdash \mathcal{Y}$	
2.4 トフノルシューナイノク	Z1
	05
3. 参考トキュメント	



1. 概要

このアプリケーションノートでは、FIT モジュールを e² studio のプロジェクトに組み込む方法について説明 します。

 e^2 studio の使い方は、 e^2 studio のユーザーズマニュアルを参照してください。なお本 APN で使用している e2studio はバージョン v5.3.0.023 を使用しています。

1.1 用語説明

用語	意味
FIT	Firmware Integration Technology(ファームウェア総合テクノロジ)の略称です。
e ² studio	e ² studio は、グローバルに普及しているオープンソースの統合開発環境"Eclipse"
	と、C/C++言語開発を可能とする CDT プラグインをベースとし、ルネサス製 RX ファ
	ミリに対応した統合開発環境ツールです。
BSP	Board Support Package (ボードサポートパッケージ) の略称です。
	マイコンの初期設定、クロック設定、ボード設定などを行うモジュールです。
	FIT モジュールを使用するプロジェクトの基礎となります。
	BSP 用 FIT モジュールは"r_bsp"と称します。
CMTW	Compare Match Timer W(コンペアマッチタイマ W)の略称です。
	RX の CMTW 用 FIT モジュールは"r_cmtw_rx"と称します
プラットフォーム	ターゲットボードと同じ意味です。



1.2 FIT の構成

FIT は、BSP、周辺機能モジュール、ミドルウェアモジュールおよび、インタフェースモジュールで構成されています。

- BSP:マイコンの初期設定、クロック設定、ボード設定などを行うモジュール
- FIT 周辺機能モジュール: RX マイコンの周辺機能を制御するドライバ
- FIT ミドルウェアモジュール: TCP/IP やファイルシステムなどのミドルウェア
- FIT インタフェースモジュール: Socket API などの共通インタフェース

これらを用いることにより、ソフトウェア開発が容易になります。

図 1.1にFIT の構成を示します。



図1.1 FIT の構成



1.3 FIT モジュールアプリケーションノートのファイル構造

FIT モジュールアプリケーションノートは、単一のZIP ファイルとしてルネサス Web サイトからダウンロードできます。

FIT モジュールアプリケーションノートは、ファイルやフォルダの共通のサブセットが含まれています。 図 1.2に架空の FIT モジュール周辺機能 "abc" を含むFIT モジュールアプリケーションノートの内容を示し ます。



図1.2 FIT モジュールアプリケーションノートの内容



図 1.3 架空の FIT モジュール "r_abc_rx"の内容

図 1.3は FIT モジュールの一般的なファイル構造を示します。

すべての FIT モジュールは"r_"の接頭辞で始まります。これは、ルネサスモジュールであることを意味 します。"r_"の後は、すべて小文字でモジュール機能名称とアンダースコア RX ファミリを表す"_rx"が 続きます。

"doc"フォルダは FIT モジュールのドキュメントを格納しています。

"src"フォルダは FIT モジュールに必要なすべてのソースファイルおよびヘッダファイルを格納しています。 "src"フォルダ配下には必要に応じて、さらにサブフォルダが存在することもあります。

"ref"フォルダは "_config"の接尾辞を持つコンフィギュレーションヘッダファイルを格納しています。これは、FIT モジュールの設定ファイルです。ユーザのニーズに合わせてコードを形成する複数のマクロが含まれています。

"r_"から始まる FIT モジュール名のフォルダ直下には2つのファイルを格納しています。

1 つは接尾辞 "_if" を含むヘッダファイルです。このファイルは FIT モジュールを使用するためのインタフェース(API 関数のプロトタイプ宣言、型定義、マクロなど)を含んでいます。

2つ目は readme.txt ファイルです。このファイルは FIT モジュールのバージョン、機能などの情報を含んでいます。



2. FIT コンフィギュレータを使用して FIT モジュールを組み込む方法

2.1 e²studio プロジェクトの作成

e²studio の新規プロジェクトを作成し、FIT コンフィギュレータを使用し BSP、CMTW の FIT モジュールを 組み込む方法を説明します。

この章では、プラットフォームに Renesas Starter Kit+for RX64M(以下、RSK+RX64M)を使用した場合を例に 説明しています。その他プラットフォームをご使用の場合は適宜読み替えてください。

図 2.1に新規プロジェクト設定方法の流れを示します。



図 2.1 新規プロジェクト設定方法の流れ



2.1.1 新規プロジェクトの作成

- e²studio を起動し、以下のいずれかの方法で新規プロジェクト作成ウィザードをオープンしてください。
 例では C プロジェクトとします。
 - •メニューバーの「ファイル(F)」→「新規作成(N)」→「C Project」を選択してください(①)
 - ツールバーの ご ボタンをクリックし(①)、ウィザード選択画面から「C Project」を選択し(②)、「次 へ(N)」ボタンをクリック(③)してください。
 - ツールバーの [1] 横の▼ボタンをクリックし(①)、「C Project」を選択してください(②)

	린(N)	Alt+シフト+N >	Makefile Project with Existing Code	1	🌣 • 🜔 • 🎴 • 🙋 🗁 🔗 • 🔲	🖬 🖢 + 🏹 + 🌤	(
ファイ	イルを開く(.)		C Project			クイック・アクセス]
閉じ	-ā(C)	Ctrl+W	C++ Project				
すべ	て閉じる(L)	Ctrl+シフト+W	う プロジェクト(R)				≥ /23
保管	ξ(S)	Ctrl+S	🔟 C/C++ プロジェクトに変換 (C/C++ ネーチャーを追	1D)		唐	■" 長示するアウトラインはあ
別名	5(-) 6保存(A)	6	🎽 ソース・フォルダ			ŧ	th.
すべ	て保管(E)	Ctrl+シフト+S	フォルダー				
前回	回保管した状態に戻す(T)	[ビ ソース・ファイル				
移動	ħ(V)						
名前	ńを変更(M)	F2					
更新	f(F)	F5	≝ ソフス ⇒ ∧ カカ				
行区	図切り文字の変換(D)	>	329				
印刷	l](P)	Ctrl+P	😚 その他(O)	Ctrl+N			
n_/	クフパーフの行いまうハム						
Port		í l					
				[™] ⇒ (W) へルプ(H)	< ☆ • 0 • 4 • @ ⇔ % •	• • • • • •	-
	tudio 編集(E)	ます	× 17	ೆಂ)(W) ∧ルプ(H) ೨ ▼ 🖻 ▼ 🕑 ▼	r] ☆ ▼ () ▼ % ▼] @ @ <i>A</i> ▼	: □ 〒 : ね ▼ २ ▼ 2イック・アクセス	- □ > + + + + + +]
70ÿI	rudio 編集(E)	ġ		ゔ(W) へルブ(H) 3 • ピ • G •	r] ☆ ▼ () ▼ % ▼]@ @ <i>%</i> ▼	। 	
	rudio 編集(E) を 新た ウイザードを選択 新しいてプロジェクトを主成し マパト・エクスプロ ウイザード(W):	ます		ジウ(W) へルプ(H) 3 ▼ C ▼ G ▼	r] ☆ ▼ () ▼ % ▼]@ @ <i>%</i> ▼	া । । । । । । । । । । । । । । । । । । ।	
7051	rudio 編集(E) ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・			ジ(W) ヘルプ(H) 3 • C • G •	*] * • O • • • 19 © % •	: ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ ፡፡ <u>74ック・アクセス</u> E	- □ > • • • • • • •] : ::::::::::::::::::::::::::::::::::
	rudio 環集(E) ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	^{يغ ب}		⁵ ク(W) ヘルプ(H) 3 ・ C ・ G ・	r] 🎋 • O • 9 • 19 🕒 🖉 •	। । । । । । । । । । । 2/170-779tz । । ।	- □ >
	rudio 環集(E) ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	.ます ② クトに変換 (C/C++ ネーチ	- X V * (う(W) ヘルブ(H) 3 • ピ • ピ •	r] 🌣 • O • 🥵 • [@ @ <i>P</i> •	: 미 : : : : : : : : : : : : : : : : :	- □ > - □ > - □
て イレー プロジェ	rudio 環集(E) 2000 第100	.ます ② クトに変換 (C/C++ ネーチ	- · · · · · · · · · · · · · · · · · · ·	⁵ ?(W) ヘルプ(H) 3 • ピ • G •	ri ☆ • O • ۹ •i⊕ ⊏ <i>%</i> •	0 ता 2 • २ • 2192-7242 ि ट	- □ >
7091	rudio 環集(E) ● 新 フィザードを選択 新しい Cプロジェクドを主成し アバクトエクスプ、 クパゲード(W): アバクスカ マーク CFI CProject EI CProject EI CProject EI CProject EI CProject EI CProject EI CProject EI CProject	ます ② クトに変換 (C/C++ ネーチ ect with Existing Code	- · · · · · · · · · · · · · · · · · · ·	⁵ ウ(W) ヘルプ(H) S • ピ • G •	r[☆▼0×9⊾×]@⊝%▼	: 0 11 : ユ ・ シ ・ 21ック・アクセス ロ E	- □ >
70ÿI	rudio 環集(E)	よす ② クトに変換 (C/C++ ネーチ set with Existing Code periset	- · · · · · · · · · · · · · · · · · · ·	⁵ 7(W) ヘルプ(H) 3 • ピ • G •	ri☆▼0×♀ vi❷⊜∦▼	: 0 11 : ユ ・ キ ・ 21ック・アクセス ー C	- □ > ・
с (70Ут	rudio 環集(E) 2 ■ 新 シイザードを選択 新しいてフジェクトを主成し がガード(W): アルタ入力 マート・エクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ マート・ロクスプ	よす ② クトに変換 (C/C++ ネーチ ect with Existing Code ject Project	- · · · · · · · · · · · · · · · · · · ·	や(W) へルブ(H) 3 ▼ ピ ▼ G ▼	ri☆ • O • ᡐ • i @ © % •	:0 11 : ユ ・ キ ・ 21ック・アクセス ー E	- □ > ・
с () / 70Ут	rudio 環集(E) 2 ■ 新社 クイザードを選択 新しいてフジェクトを主成し オガード(W): アルタ入力 マート・エクスプ マート・エクスプ マート・エクスプ マート・エクスプ マート・マジェー マート・マジョー マート・マショー マート・マショー マート・マジョー マート・マショー マート・マジョー マート・マショー マート・マショー マート・マショー マート・マショー マート・マショー マート・マン マート・マショー マート・マン マート・マン マート・マン マート・マン マート・マー マート・マー マート・マー 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 マート マート マート マート マート マート	よす ② クトに変換 (C/C++ ネーチ ect with Existing Code ject Project	- · · · · · · · · · · · · · · · · · · ·	⁵ 7(W) ∧IJJ(H) 3 • ₫ • ₢ •	ri∰ v 0 • 9 • i @ ⊡ % •	: 0 11 : ユ ・ ギ ・ 21ック・アクセス ー E	- □ >
c () Juyr	rudio 環集(E)	ます ② クトに変換 (C/C++ ネーチ ect with Existing Code ject Project ファイル	- · · · · · · · · · · · · · · · · · · ·	⁵ ⁵ (W) ∧IJ <i>ブ</i> (H) ⁹ ▼ ² ▼ ² ▼ ² ▼	ri ☆ • O • Q • i @ © % •	:0 11:4 × わ × <u>21ック・アクセス</u> ー E	- □ ×
с () 70Ут	rudio 編集(E)	ます ② クトに変換 (C/C++ ネーチ ect with Existing Code ject Project ファイル	- · · · · · · · · · · · · · · · · · · ·	⁵ ⁵ (W) ∧IJJ(H) ⁹ ▼ ² ▼ ² ▼ ² ▼	ri ☆ • O • Q • i @ ⊑ % •	:0 11:2 * う * <u>クイック・アクセス</u> ー E	- □ >
c () J J J U J U J	rudio 編集(E) ● 新 クイザードを選択 新しいて力ジェクトを主成し クパザードの注 ポレいて力ジェクトを主成し マパラスカ マパラス マパラスカ マパラス マパラスカ マパラス マパラスカ マパラス マパラス マパラス マパラス マパラス マパラス マパラス マパラス マパラス マパラ マパラ マパラ マパラ マパラ マパラ マパラ マパラ	ます ② クトに変換 (C/C++ ネーチ ect with Existing Code ject Project ファイル	- · · · · · · · · · · · · · · · · · · ·	⁵ ⁵ (W) ∧IJJ(H) ⁹ ▼ ² ▼ ² ▼ ² ▼	ri ☆ • O • Q • i @ ⊑ % •	:0 11:2 × わ × <u>21ック・アクセス</u> ー E	- □ >
c () Juÿr	rudio 環集(E) ● ウイザードを選択 新しいて力ジェクトを主成し クイザード(W): フィリクスカ クイザード(W): フィリクスカ ・ C ・ C ・ C ・ C ・ C ・ Project E Synergy C++1 G クラスス ローン ・ アンノートからご コーン ・ デンフレートからご コーン ・ デンフレートからご コーン ・ デンフレートからご コーン ・ デンフレートからご	ます ② クトに変換 (C/C++ ネーチ set with Existing Code ject 7ァイJル)	· · · · · · · · · · · · · · · · · · ·	⁵ 9(W) へルプ(H) 3 ▼ ピ ▼ G ▼	r[☆ • O • Q •] @ © <i>%</i> •	<u>2199-794</u> 2 □ [- □ > ・

RX ファミリ

C - e ² studio	- 0) X
ファイル(F) Vース(S) リファクタリング(T) ナビゲート(N) 検索(A) プロシ	ジェクト(P) Renesas Views 実行(R) ウィンドウ(W) ヘルプ(H)	
🔁 📲 🛯 🙆 🚽 🗞 🕶 🔏 🕼 🖉 💷 🖉 💆 🖂 🗷 🗏	8 \$	⇒ -
Makefile Project with Existing Code	クイック・アクセス 😰 📔	🗄 C/C++
C Project	· · · · · · · · · · · · · · · · · ·	
C++ Project		
📑 プロジェクト(R)	主	 ⊼⊽⇔⊾=∡`
C/C++ プロジェクトに変換 (C/C++ ネーチャーを追加)	マハリッ (ありま	せん。
3 ソース・フォルダ		
≧ フォルター		
▶ ヘッダー・ファイル		
▶ テンプレートからファイル		
G 25X		
<u>б</u> 977		
📑 その他(O) Ctrl+N		

- (2) 新規プロジェクトウィザード(C プロジェクト)の「プロジェクト名(P)」にプロジェクト名を入力して ください。(④) 例では "rskrx64m_fitcfg_demo" としています。
- (3) 「ツールチェイン」より "Renesas RXC Toolchain"を選択し(⑤)、「次へ(N)」ボタンをクリック(⑥) してください。

	■ Cプロジェクト Cプロジェクト 選択したタイプの Cプロジェクトを作成します	- • ×	
	プロジェクト名(P): rskrx64m_fitcfg_demo		
	 ごフォルト・ロケーションの使用(D) ロケーション(L): C:¥Workspace¥rskrx64m_fitcfg_d ✓ Create Directory for Project 	emo 参照(R)	
	プロジェクトの種類:	ツールチェイン:	
	 Executable (Renesas) Sample Project Static Library (Renesas) Sample Project Debug-Only Project Debug-Only Project 会 static ライブラリー 会 共用ライブラリー 会 実行可能 Makefile プロジェクト 	KPIT GNUARM-NONE-EABI Toolchain KPIT GNURL78-ELF Toolchain KPIT GNURX-ELF Toolchain Renesas CCRL Toolchain Renesas GCC for RL78 Renesas GCC for RX Renesas RXC Toolchain	
-	⑦ ブラットフォームでサポートされる場合のみ、プロジェク ⑦ < 戻る(B)	○ト・タイプと toolchain を表示 ⑥ 次へ(N) > 終了(F) キャンセル	-



- (4) 新規プロジェクトウィザード(ターゲット固有の設定の選択)の「ツールチェイン・バージョン」からバー ジョンを選択してください。(⑦) 例では "v2.06.00" としています。
- (5) 「デバッグ・ハードウェア」からデバッガを選択してください。(⑧) 例では "E1(RX)" としています。
- (6) 「データ・エンディアン」からエンディアンを選択してください。(⑨) 例では"Little-endian data" としています。
- (7) 「ターゲットの選択」から MCU を選択してください。(⑩) 例では "R5F564MLCxFC" としています。
- (8) 「次へ(N)」ボタンをクリックしてください(⑪)





RXファミリ

- (9) 新規プロジェクトウィザード(コード生成支援ツールの選択)の「コード生成または Firmware Integration Technology(FIT)」をチェックします。(⑫)
- (10) 「FIT モジュールを使用する」をチェックします。(13)
- (11) FIT モジュールのタウンロードが必要な場合は「FIT モジュールのダウンロード」をクリックしてください。(④) 詳細は「2.1.2 FIT モジュールのダウンロード」を参照してください。例では FIT モジュールをダウンロードしません。
- (12) 以降の新規プロジェクトウィザードは、追加 CPU オプションの選択、グローバルオプションの選択、 標準ヘッダー・ファイルの選択となります。設定が必要な場合は「次へ(N)」ボタンをクリックし(⑤)、 不必要であれば「終了(F)」ボタンをクリック(⑥)してください。





2.1.2 FIT モジュールのダウンロード

FIT モジュールは e²studio の機能を使用して WEB からダウンロードします。なお FIT モジュールのダウン ロードは、新規プロジェクト作成時以外でもすることができます。詳細は「2.4トラブルシューティング」の (5)を参照してください。

(1) 新規プロジェクトウィザード(コード生成支援ツールの選択)の「FIT モジュールのダウンロード」をク リックしてください。(①)





- (2) e²studio が WEB から FIT モジュール情報を取得します。
- (3) 取得した結果が FIT モジュールのダウンロードウィザードに表示されます。デフォルトは RX Driver Package*のみを表示する絞り込みがチェックされていますので、RX Driver Package が表示されます。
 (2) もし各周辺機能やミドルウェアを表示させる場合は、絞り込みのチェックを外してください。(3)

*RX Driver Package とは、FIT モジュールすべてを1つにパッケージングしたソフトウェアプラット フォーム(フレームワーク)です。本パッケージを使用することを推奨しています。

■ FITモジュールのダウンロード				- 🗆	\times
ダイトル アスファミリ RX Driver Package Ver.1.11	1	ドキュメントNo. R01AN3467JJ0111	ドキュメントN Rev.1.11	発行日 2016/10/13	
- 絞り込み:					
✓ RX Driver Package のみ表示する					
FITモジュール・フォルター・バス:					* 82
C/ nenesas/ ec_studio/ Friviodules	オバブ湖山	選択をすべて経験			× 22
	977(进5)(違いを引くて用体			
				ダウンロード 🕇	ャンセル
					~
タイトル		ドキュメントNo.	ドキュメントN	発行日	Â
RX Jアミリ SRCモシュール (央語版) Fi RX Jアミリ SRCモシュール (央語版) Fi RX Family Valtage Datastics Circuit	rmware Integrati	R01AN12090EJ0111	Rev. 1. 11	2014/12/12	
RX Tarning Voltage Detection Circuit RX Tarsil ロングロード刑キューパッファ ((IONGO) モジュ	R01AN1889U0160	Rev 1.60	2015/07/09	
RX ファミリ バイト型キューバッファ (BYTE)	EO) モジュール Fir	R01AN1683JJ0160	Rev. 1.60	2016/01/29	
 RXファミリ SSIモジュール (英語版) Firit 	mware Integrati	R01AN2150EJ0120	Rev.1.20	2015/04/28	
III RX ファミリ EEPROM アクセスクロック同	_ 期式制御モジュー	R01AN2325JJ0233	Rev.2.33	2015/12/29	
RX Family SCI FIFO Module Using F	irmware Integra	R01AN2222EU0110	Rev. 1.10	2015/03/19	
RXファミリ Serial Flash memory アクセ	スクロック同期式	R01AN2662JJ0233	Rev.2.33	2015/12/29	
RX Family Flash Module Using Firm	ware Integratio	R01AN2184EU0200	Rev.2.00	2016/08/17	
RX ファミリ EPTPC モジュール Firmare Ir	ntegration Tech	R01AN1943JJ0112	Rev. 1.12	2016/11/11	
RXファミリ組み込み用TCP/IP M3S-T4	-Tinyを用いたFTP	R20AN0078JJ0104	Rev. 1.04	2016/10/01	
KN 7ァ5リ 組み込み用TCP/IP M Styling Package のみまニナス		R20AN0081U0104	Rev. 1.04	2016/10/01	•
					60 RZ
C/Kenesas/e2_studio/FIIModules					≫ 州飛
	すべて選択	選択をすべて解除			
				ダウンロード 🛔	ャンセル



- (4) ダウンロードしたい FIT モジュールのチェックボックスをチェックします。(④) 例では RX Driver Package をチェックします。
- (5) 「ダウンロード」ボタンをクリックします。(⑤) 一度も My Renesas による認証をしていない場合、 My Renesas ダイアログがオープンします。ルネサス WEB サイトで登録しているメールアドレスとパ スワードを入力してください。

RXファミリ RX Driver Package Ver.1.11 R01AN3467JJ0111 Rev.1.11 2016/10/13 A Support Rev.1.11 2016/10/13 Support Rev.1.11 2016/10/13 Rev.1.11 Rev.1.11 2016/10/13 Rev.1.11 Rev.1.11 2016/10/13 Rev.1.11 Rev.1.11 2016/10/13 Rev.1.11 Rev.1.11 Rev.1.11 Rev.1.11 Rev.1.11 Rev.1.11 Rev.1.11 2016/10/13 Rev.1.11 Rev.1.11	(4)	ドキュメントNo.	ドキュメントN	発行日
較り込み: 図 RX Driver Package のみ表示する FITモジュール・フォルダー・パス:	RXファミリ RX Driver Package Ver.1.11	R01AN3467JJ0111	Rev.1.11	2016/10/13
絞り込み: ☑ RX Driver Package のみ表示する FITモジュール・フォルダー・パス:				
	호リ込み: ☑ RX Driver Package のみ表示する ITモジュール・フォルダー・パス:			

(6) 以降は、「2.1.1 新規プロジェクトの作成」の(12)を参照してください。



2.2 FIT モジュールの組み込み

2.2.1 FIT コンフィギュレータの起動

メニューバーの「Renesas Views」→「e2 ソリューション・ツールキット」→「FIT Configurator」を選択(①) し、FIT コンフィギュレータを起動します。(②)

アイル(F) 編集(E) ソース(S) リファク	クタリング(T) ナピゲート(N)	検索(A) プロジェクト(P) Rer	nesas Views 実行(R) ウイン	/ドウ(W) ヘルプ(H)		0 - 0 - 1		
3 • 🛛 🕞 🛆 🗞 • 🔨 • 1	% 🗟 🕅 🕨 II 🗉	N R O R R R	C/C++(C)		•• ①	😕 🗁 🖋 📲		
			Tracing	> 💯 FIT Configurate	or		0190.002	E E E C/C++
<u>5</u> 70ジェクト・エクス70-∋- ∞			コード生成	> L 最適化アシスタン	/ ト			3 "3 "
> 😂 rskrx64m_fitcfg_demo	i⊟ d olle.		スマート・コンフィクレータ デバッグ	 > > スマート・フラウサ・ > コマート・マニュア 	-			e 表示するアウトライン
			パートナーのS	>				はめりません。
		_	ルネサス OS	>				
					1			
<u> </u>					1			
🖄 rskrx64m_fitcfg	g_demo					1		
C - 2 + 1	g_demo							
C - e2 studio F/L(E) 編集(E) ソース(S) リファ?	g_demo クタリング(T) ナビゲート(<u>N</u>)	検索(<u>A</u>) プロジェクト(<u>P</u>) Rer	nesas <u>V</u> iews 実行(<u>R</u>) ウイン	/ドウ(W) ヘルプ(H)				×
 C - e2 studio P P I I	g_demo クタリング(T) ナビゲート(<u>N)</u> 気 品 : へ レ ロ 回	検索(A) プロジェクト(P) Ret お ユ つ ユ 弓 変 (数	nesas <u>V</u> iews 実行(B) ウイン 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	バウ(M) ヘルブ(H) 3 マ ご マ G マ ! : :) - 9 - 10	: • ~ • • •] 加 - 神 - Y	×
²²⁵ rskxx64m_fitcfy C - e2 studio かんし 編集(E) ソース(S) リファ/ S ▼ 副 ● ● ● ● ▼ ※ ▼	g_demo クタリング(T) ナピブート(N) (1) 前: へ レ ロ 回	検索(A) プロジェクト(P) Ref ド ユ コ ユ 弓 武 (名)	hesas ýiews 実行(B) ウイン) : 「雪」 ダ : む : ピ ▼ (שלער שניאי די די די די נ) • Q • (<u>@</u> (: • ~ • : • •	1 シ ・ 和 ・ * クイック・アクセス	×
C - e2 studio C - e2 studio マイルビ 編集(E) ソース(S) リファク マーマ 常 電 二 ○ ○ マ る マ コクロジェクト・エクスプローラー ※	<u>g_demo</u> のタリング(T) ナビオート(N) (1) 日 日 日 日 日 日 日 日 日 日 日 日	検索(<u>A</u>) プロジェクト(<u>P</u>) Rer ド ユ ロ ユ 弓 変 (名	hesas <u>V</u> iews 実行(B) ウイン ※ 『 ¹¹ 』 <i>例</i> : 《 】 (バン(1) (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)) • Q • [@ (:] 加 · 和 · · · 24ック・アクセス 日 日	- □ ×
 ご rskx64m_fitcfy C - e2 studio r/ル(D 編集(D ソース(D リファ)) マ 副 副 副 副 マ ● マ ● マ プロジェクト・エクスプローラー ※ ご rskx64m_fitcfg_demo 	<u>g_demo</u> 775リング(T) ナビゲート(L)	検索(<u>A</u>) プロジェクト(<u>P</u>) Rer ピュロュモラ 家 (名	hesas <u>V</u> iews 実行(B) ウイン 2 : 『 』 : 愛 : 愛 : ② : ② ▼((山下 (山下 (山下 (山)) • Q • (@ (: • * • ! • 5] থ • ম • • //৩/০৮/৮০ ি চ	- □ × - □ × - · · · · - · · · ·
 ご rskxx64m_fitcfg C - e2 studio r/ル()	<u>g_demo</u> 729リング(T) ナビゲート(<u>N</u>) () () () () () () () () () () () () ()	検索(A) プロジェクト(P) Rer パコ. マ. エモ、安二征	hesas <u>Vi</u> ews 実行(B) ウイン 2 : 「望」 ダ : 乾一 : ♂ ▼ (イドウ(M) ヘルプ(H) 3 マ ご マ G マ :) + Q + (@ (: • * • • • •	ו איז	- □ × ■ □ × → × ■ □ □ □ ○ × 素示するアウトライン はありません。
 ご rskx64m_fitcfg C - e2 studio ペ(ルビ) 振集(E) ソース(S) リファ/ マ (E) (E) (E) (E) (E) (E) (E) (E) (E) (E)	g_demo 7/91)2/9(T) †27 K(<u>N</u>) S	検索(A) プロジェクト(P) Rer が ユ ロ ユ 弓 愛 (G	hesas ⊻iews 実行(B) ウイン 3 [¹¹³] 2 [2] 2 [2] (2] ▼ (/ドウω」 ヘルプ(出) 3 ~ ご ~ G ~ !) • Q • (@ (: • A • 0 5] 2 × 2 × 3 //1//77世ス □ □	- □ × - □ = □ × - = × - = × - × - = ×
 ご rskx64m_fitcfg C - e2 studio ゲ(ル() 振集() ソース() リファ/ マ () () () () () () () () () () () () ()	g_demo 799リング(T) † 拉丁-ト(<u>N</u>) ⑤ 励 № IP II ■ □ ○ ⑤ ⑤ 》 ▼	検索(Δ) プロジェクト(2) Re N ユ つ ユ モ 家 (6	hezas <u>V</u> iews 夏行(B) ウイン 2 - ¹¹ 2 - 2 - 2 - 2 - 2 - 2 - 2 - 2 - 2 - 2 -	1 ² 70 M ∧11/7 H 3 • C • C •) • Q • (B (: • * • • • •	য়া ২ থা ২ ২ ০৭৩০-৫০৫ ₽ ■	- □ × - □ × - ○ · · · · · - ○ · · · · · · - ○ · · · · · · - ○ · · · · · · · · - ○ · · · · · · · · · · · · · · · · · ·
 C - e2 studio イルE 編集E ソース(S リファク) マ 副 ● ● ● ▼ ▼ ▼ ▼ プロジェクト・エクスプローラー ※ rskx:64m_fitcfg_demo 	<u>g_demo</u> 7751)ング(T) + 727 - K-(A)	検索(A) プロジェクト(P) Rer ド ユ ロ ユ マ 安 袋 袋	hesas <u>V</u> iews 実行(B) ウイン ※ : ¹¹ : <i>1</i> : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 :	שלאר שילא די די ד	2 2] : 철 • 현 • • 2199-7942 모 모 모	- □ × ● ● ● ■ C/C++ ■ ● ■ C/C++ ■ ● ■ ■ C/C++ ■ ● ■ ■ C/C++ ■ ● ■ ■ C/C++
 ご rskx64m_fitcfg C - e2 studio イルビ 振集(E) ソース(S) リファジ マ (G) マ (G) マ (G) マ (G) マ (G) コ ブロジェクト・エクスプローラー S ご rskx64m_fitcfg_demo 	9_demo 7/9リング(T) ナビゲート(M) 低品 (N 下 T) 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日	検索(A) プロジェクト(P) Rer パコーローエーラ 変 (G	hesas ⊻iews 実行(B) ウイン 2 『 『 』 2 『 C ▼ (۲۶۱۳ (mc, mc, mc, mc, mc, mc, mc, mc, mc, mc,	· • • • ● 0 7 ②		//////// /////////////////////////////	- □ × 一 □ × ● × • • × ● □ □ C/C++ □ □ □ × ● × • • × ● × • • × ● ▼ → × ■ ■ → × ■ → ×
 ご rskx64m_fitcfg C - e2 studio ペルビ 編集() ソース() リファリ マ () () () () () () () () () () () () ()	g_demo 799192/9(1) †ਏ7K(<u>N</u>) இ இ № IP II = E % ₽ ♥	検索(A) プロジェクト(P) Rer N コ. マ. エ マ 家 : 6 PIT Configurator 23 FITモジュールを追加するプロS	hesas ⊻iews 実行(B) ウイン 3 1 ^m 3 1 g × (c) 5 2 7 1: rskrx64m_fitcfg_der	льд тр тр тр тр тр тр тр тр тр тр	7 2		○ 1-16 ○ 1-16 ○ 1-16 ○ 1-16 ○ 第子設定 1.75(a)	- □ × ● ▼ ● ▼ ● ▼ ● ▼ ● ■ □ C/C++ ■ 3 □ □ □ ● ▼ 表示す370ト5/17 はありません。
 ご rskx64m_fitcfg C - e2 studio アイル() 振集() ソース() リファ: マ () () () () () () () () () () () () ()	g_demo 799リング(T) †拉7K(M) ⑤ 励 ℃ ■ □ ■ ■ ○ ○ ○ □ ■ ■ ○ ◎ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○	検索(A) プロジェクト(P) Rer N コ. つ マ 家 : 6 FIT Configurator 23 FITモジュールを追加するプロS クーグットボード RSKRX64M	hetas <u>V</u> iews 実行(B) ウイン 2 1 1 2 1 2 1 2 マ (2 1 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2	バウω ヘルプ曲 3 ▼ C ▼ C ▼ : no 設定	7 2		 ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○<!--</td--><td>- □ X - □ X</td>	- □ X - □ X
 C - e2 studio C - e2 studio グルD 編集(D ソース(S) リファ: ブロジェクト・エクスプローラー ※ プロジェクト・エクスプローラー ※ C rskn:64m_fitcfg_dema 	g_demo 27911/1/(T) †£27K(M) ③	検索(A) プロジェクト(P) Rer N コ. つ . c 弓 定 値 FIT Configurator 28 FITモジュールを追加するプロS クーゲットボード RSKRXGAW 使用可能なモジュール	hetas <u>V</u> iews 実行(E) ウイン 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	(ドウω) ヘルプ(出) ③ ▼ (≧ ▼ (Ĝ ▼) ○ ▼ (≧ ▼ (Ĝ ▼) no 1設定) • Q. • [≥ () 2 (2) 3 #R(L¢EV)2-)		 2/19/)・アクセス 2/19/)・アクセス □ □ □ □	- □ × ■ □ × ■ □ CCC++ □ 3 ³ 3 □ □ 素示するアウトライン はありません。 の生成 ▽ □ □
 ご rskx64m_fitcfg C - e2 studio マイル(D 編集(D ソース(S リファ) マ 副 一 当 マ ~ ~ ~ ~ プロジェクト・エクスプローラー ※ プロジェクト・エクスプローラー ※ 	g_demo 2/9/10/7(T) †£7F(<u>N</u>) ③	検索(A) プロジェクト(D) Rer N コ. コ. ユ. マ. マ マ (C) FIT Configurator 23 FITモジュールを追加するプロ3 ターゲットボード RSKRX64W 使用可能なモジュール モジュール モジュール モジュール スト	hesas ⊻iews 実行(E) ウイン 注「「」 ジ ビ」 ご マ (バロクト: rskrx64m_fitcfg_der 1 2/1/4の詳細 ジ 概要) Beard Support Packa	/ドウ(W) ヘルブ(H) 3 * ご * G * ! no es.	 マ Q. マ (2) 選択したモジュー) モジュール 		 2 · 2 · 2 · · · · · · · · · · · · · · ·	- □ × ● ● ● ● ■ C/C++ 1 3 *3 = □ まがするアクトライン はありません。 ④ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
 ご rskx64m_fitcfg C - e2 studio Mル(D 編集(D ソース(D リファ) マ (D マス) マ (g_demo 79910/9(T) †£97-Fr(<u>N</u>) 1 ■ 1 ■ 1 ■ 1 ■ 1 ■ 1 ■	検索(A) プロジェクト(D) Rer N ユ つ ユ マ 32 (G) FIT Configurator 23 FITモジュールを追加するプロ3 ターゲットボード RSKRX64W 使用可能なモジュール モジュール 「 モジュール」、「 て、Dsp 3.44 r_Dyteq 1.66	hesas <u>Views</u> 実行(B) ウイン 注意 「立力ト: rskrx64m_fitcfg_der 1	/ドウ(W) ヘルプ(H) 3 ・ ピ ・ G ・ : 10 元 ges uffer library.	2 ② 選択したモジュー) モジュール		 21 - 10 - 10 - 10 - 10 - 10 - 10 - 10 -	- □ × - □ × - · · · · · - · · · · · - · · · · · · - · · · · · · · - · · · · · · · · - · · · · · · · · · · · · · · · · · · ·
 ご rskx64m_fitcfg C - e2 studio P/I/LD 福集(E) ソース(S) リファ: マ (R) (R) (R) (R) (R) (R) プロジェクト・エクスプローラー S ご rskx64m_fitcfg_demc 	<u>g_demo</u> 79907/0, †£7-K <u>N</u> S S S S S S S S S S S S S S S S S S S	検索(A) プロジェクト(D) Ref N 3. 0 1.2 マ 32 日 N 3. 0 1.2 マ 32 日 FIT Configurator 23 FITモジュールを追加するプロ3 ターグットボード RSKRX64W 使用可能なモジュール モジュール パー r_bsp 3.44 r_byteq 1.66 r_comty_rx 1.22	hesas Views 実行(B) ウイン シーション・シーン バロクト: rskrx64m_fitcfg_der 1	ドウ(W) ヘルプ(H) ③ ▼ ⑦ ▼ ③ ▼ ③ ▼ 〕 ■ ■ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	マ Q マ (2) 選択したモジュール		 21.90・79 tex 21.90・79 tex □ □ □ □<td>- □ × ■ ■ □ CC++ ■ □ □ □ □ □ □ ■ □ □ □ □ □ ■ □ □ □ □ □ ■ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □</td>	- □ × ■ ■ □ CC++ ■ □ □ □ □ □ □ ■ □ □ □ □ □ ■ □ □ □ □ □ ■ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ □ □ ■ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 ご rskxx64m_fitcfg C - e2 studio r/IルE 編集(E) ソース(S) リファ: マ (E) (E) (E) (E) (E) (E) (E) (E) (E) (E)	<u>g_demo</u> 799974(T) †£7-K <u>N</u>) S	検索(A) プロジェクト(2) Rer パ ユ ロ ユ マ ズ (2) FIT Configurator 23 FITモジュールを追加するプロン ターグットボード RSKRX64M 使用可能なモジュール モジュール パー r_bsp 3.44 r_byteq 1.66 r_can_rx 2.11 r_cmt_rx 3.00 r_dr.rx 2.00	hesas Views 実行(B) ウイン シーション・シーン バロクト: rskrx64m_fitcfg_der 1	NFウ()) ヘルプ(山) マーマ マ マ マ マ マ マ マ マ マ マ マ マ マ マ マ マ マ マ	マ Q → (金 → (金 ∩ (マ Q)) ス (2) 3 (2) 3 (2) 3 (2) 3 (2)		 21.90 × 21 × 12 21.90 × 70 ± 3 □ □ □	- □ × ■ □ × ■ □ □
 ご rskxx64m_fitcfg C - e2 studio r/IルE 編集(E) ソース(S) リファ: マ (E) (E) (E) (E) (E) (E) (E) (E) (E) (E)	<u>g_demo</u> 799974(T) †£7-K <u>N</u> S	検索(A) プロジェクト(2) Rer パ ユ ロ ユ マ ズ ぼ (2) FIT Configurator 23 FITモジュールを追加するプロS ターゲットボード RSKRX64W 使用可能なモジュール モジュール パー r_cbsp 3.44 r_byteq 1.66 r_can,rx 2.10 r_cac,rx 2.10 r_dac,rx 2.97 r_dmaca,rx 1.02	hesas Views 実行(B) ウイン シートン マイン マイン マイン マイン マイン マイン マイン マイン マイン マイ	no Rěfit ges. uffer library. M and RX64M. r creating timer tick. nverter driver	マ Q → (金 → (金)) マ Q → (金) オ ストーン・(本) (1 → (-1)) オ オーン・(-1) オ オーン・(○ 2 × 2 × 2 × 3 × 5 × 5 × 5 × 5 × 5 × 5 × 5 × 5 × 5	- □ × ■ ■ □ × ■ ■ □ CC++ ■ □ □ □ = = = = = = = = = = = = = = = =
 ご rskxx64m_fitcfg C - e2 studio アイルビ 編集(E) ソース(S) リファ: マ 二 二 ※ マ へ 、 マ プロジェクト・エクスプローラー ※ ご rskxx64m_fitcfg_deme 	<u>g_demo</u> 7991070(1) †t27-1-K <u>M</u>) S	検索(Δ) プロジェクト(2) Ref 対 ユ ロ エ 弓 変 (4) FIT Configurator ※ FITモジュールを追加するプロジ ターグットボード RSKRX64W 使用可能なモジュール モジュール パー r_cbsp 3.44 r_byteq 1.60 r_cont.x 3.00 r_cont.x	hesas Views 実行(B) ウイン シーロン マーレン マーレン マーレン マーレン マーレン マーレン マーレン マーレ	no REFT ges. uffer library. M and RX64M. r creating timer tick. nverter driver	マ ② 第二日の1000000000000000000000000000000000000		○ 1-F ○ 3-F ○ 5-F <p< td=""><td>- □ × ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●</td></p<>	- □ × ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
 ご rskrx64m_fitcfg C - e2 studio アイルビ 編集(E) ソース(S) リファ: マ (C) (C) (C) (C) (C) (C) (C) (C) (C) (C)	<u>g_demo</u> 7991970(1) †t27-1-1 <u>N</u>) S	検索(A) プロジェクト(P) Ref N ユ ロ エ モ ア (C) 第 FIT Configurator 23 FITモジュールを追加するプロ3 ターグットボード RSKRX64M 使用可能なモジュール モジュール パー r_cbyteq 1.66 r_con_x 2.01 r_cot_rx 2.01 r_dmaca_rx 1.00 モジュールの追加>> また 1	hesas ⊻iews 実行(B) ウイン ますり: 「skrx64m_fitcfg_der 1 ~ 21/L*の注意 ジ 概要 0 Board Support Packa 0 Boyte-based circular b 0 CATW Driver 0 CATW Driver 0 Simple CMT driver fo Digital-to-Analog Co 4 DMAC driver 5 DTC Machula	no Person	2 ② 3 ◆ Q ◆ (金)		○ 1-F	- □ × ● ▼ ● ▼ ● ▼ ● ■ ■ □ CC++ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ■ □ CC++ ● ▼ ● ▼ ● ▼ ● ▼ ● ■ □ CC++ ● ▼ ● ▼ ● ▼ ● ■ □ CC++ ■ ■ ■ □ CC++ ■ ■ ■ □ CC++ ■ ■ ■ ■ ■ ■ □ CC++ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
 ご rskx64m_fitcfg C - e2 studio アイルビア 編集(E) ソース(S) リファ: ブロジェクト・エクスプローラー 23 プロジェクト・エクスプローラー 23 ご rskx64m_fitcfg_demc 	<u>g_demo</u> 7991970(1) †£7K <u>N</u>) S	検索(A) プロジェクト(P) Ref N コ. マーム モ ア (C) FIT Configurator 23 FIT Eジュールを追加するプロS ターグットボード RSKRX64M 使用可能なモジュール モジュール パー r_bsp 3.40 r_can_rx 2.10 r_can_rx 2.10 r_cdmtw_rx 1.22 r_cmtw_rx 1.22 r_cmtw_rx 1.20 r_dmaca_rx 1.00 	hesas ⊻iews 実行(B) ウイン またいのでは、 なたりた: rskrx64m_fitefg_der 1 ジ 概要 3 Board Support Packar 3 CAN API for the RX71 3 CAN API for the RX71 3 CAN API for the RX71 3 CAN API for the RX71 4 DMAC driver for 5 DTC Madula	no	・ Q ・ (き)(・)(き)(・)(き)(・)(き)(・)(き)(・)(き)(・)(き)(き)(き)(き)(き)(き)(き)(き)(き)(き)(き)(き)(き)		⑦ ・ 2 ・ 2 ・ 2 ・ 2 ・ 2 ・ 2 ・ 2 ・ 2 ・ 2 ・	- □ × ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ま示するアウトライン はありません。 ● ▼ ■ ■ ● ▼ ■ ● ■ ● ▼ ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ●
 ご rskx64m_fitcfg C - e2 studio アイル(E) 福集(E) ソース(S) リブア: ブロジェクト・エクスプローラー ※ プロジェクト・エクスプローラー ※ ご rskx64m_fitcfg_demc 	g_demo 799197/07, †£7KN ③ ↔ K = 10 = □ □ □ ♥ ↓ ● ▼	検索(Δ) プロジェクト(2) Rer N ユ つ エ モ ア (4) FIT Configurator 23 FITモジュールを追加するプロ3 ターグットボード BSKRX64M 使用可能なモジュール モジュール パー r_bsp 3.44 r_byteq 1.64 r_can_rx 2.10 r_cds_rx 2.91 r_dmaca_rx 1.04 r_dmac_rx 2.04 Eジュールの追加>> 詳細 1	hesas Views 実行(B) ウイン シートン マートン マートン マートン マートン マートン マートン マートン マ	rk'9000 AJLJ(H) 3 • C • C • I 10 • C • I 10 • C • I 10 • C • I 10 • C • C • I 10 • C • C • I 10 • C • C • C • I 10 • C • C • C • C • C • C • C • C • C •	2 ② 3 ▼ Q ▼ (2) 3 単沢したモジュー) モジュール ▼ < <<モジュールの		⑦ ショード 初一 マー ○ クイソク・アクセス □ □ □ □	- □ × ● ▼ ● ▼ ● ▼ ● ▼ ● ■ ● ▼ ● ▼ ● ▼ ま示す370トライン はありません。 ● ■ ■ ● ■ ■ ● ▼ ● ■ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼ ● ▼
 ご rskx64m_fitcfg C - e2 studio MULD 編集(D ソース(S) リブア: ブロジェクト・エクスプローラー ※ プロジェクト・エクスプローラー ※ ご rskx64m_fitcfg_demo 	g_demo 799197/07, †£7KN 18 19 10 10 10 10 19 10 1	検索(Δ) プロジェクト(2) Rer N ユ つ エ モ ア (4) FIT Configurator 23 FITモジュールを追加するプロ3 ターグットボード BSKRX64M 使用可能なモジュール モジュール パー r_bsp 1.64 r_can_rx 2.10 r_cds_rx 2.91 r_dmaca_rx 1.04 r_dmac_rx 2.91 r_dmaca_rx 1.04 r_dmaca_rx 1.	hesas Views 実行(B) ウイン シード語 ジード語 ジー (C) マー(オークト: rsknx64m_fitcfg_der オーマン 21ルタの注意 ジー、 概要) Board Support Packar) Board Support Packar) CAN API for the RX71) CAN API for the RX71	rk'9000 AJLJ(H) 3 • C • C • I 10 F ges. uffer libray. M and RX64M. r creating timer tick. nverter driver	2 ② 3 ▼ Q ▼ (2) 3 単沢したモジュー) モジュール ↓ < <<モジュールの		⑦ ショ・ 泡 ・ ベ クイソク・アクセス ○ □ □ □ ○ □ □ □ ○ □ □ □	- □ × ● ▼ ● ▼ ● ▼ ● ▼ ● ■ ● ▼ ● ▼ ● ▼ ● ▼ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■ ● ■

2.2.2 FIT コンフィギュレータの初期設定

FIT コンフィギュレータの動作をカスタマイズすることができます。

- (1) ビューメニューのドロップダウンメニュー▽から「コンフィギュレータの初期設定」を選択し、コンフィギュレータの初期設定ダイアログを表示します。(①)
- (2) 生成オプションのモジュール生成設定を設定します。(2) 例はデフォルトの"モジュールを上書きする"としています。設定値の説明は以下の通りです。
 - ・ "モジュールを上書きする"

すべての FIT モジュールが再生成されます。

• "モジュールが存在する場合は何もしない"

「コード生成」ボタンをクリックしたときにユーザプロジェクト内のすべての既存 FIT モジュール は再生成されません。

- (3) 「モジュールの上書き/削除時、"trash"フォルダにコピーする」のチェックボックスを設定します。
 (③) 例ではチェックボックスにチェックします。(デフォルト)
- (4) 依存関係オプションの依存モジュール設定を設定します。(④) 例はデフォルトの"依存モジュールを 追加する(非再帰)"とします。設定値の説明は以下の通りです。
 - ・"依存モジュールを追加する(非再帰)"

選択した FIT モジュールに直接関係する依存モジュールは自動的に追加されます。

• "依存モジュールを追加しない"

選択した FIT モジュールに直接関係する依存モジュールがあっても自動的に追加されません。

- (5) 依存関係オプションの依存関係のチェックを選択します。(5) 例はデフォルトの"依存モジュールの バージョンを厳密にチェックする"とします。設定値の説明は以下の通りです。
 - "依存モジュールのバージョンを厳密にチェックする"

FIT モジュールと依存モジュールのバージョンが、評価された組み合わせであることを厳密に チェックします。

• "依存モジュールのバージョンをチェックしない"

依存関係のエラーや警告は表示されません。

• "より新しいバージョンの依存モジュールはチェックから除外する"

要求されるバージョンより新しい依存モジュールが選択されている場合、依存関係の警告は表示 されません。選択した依存モジュールが、要求されているバージョンより古い場合、警告が表示 されます。

(6) 「OK」ボタンをクリックします。(⑥)

*FIT Configurator	22		
ITモジュールを追加する	るプロジェクト:	rskrx64m_1	
a_Hwk#_K por	VEANA	~ ~	生成オプション ペットワークからモジュールをクリンロートする
)-))[/][-]-		× <u>7</u>	モジュールを上書きする
使用可能なモジュール	·		図モジュールの上書き/削除時、'trash'フォルダにコピーする 3
モジュール	バージョン	概要	
r_bsp	3.40	Board	
r_byteq	1.60	Byte-	な存在シュール設定「依存モジュールを追加する(評再席) ✓ 4
r_can_rx	2.10	CAN.	依存関係のチェック 依存モジュールのバージョンを厳密にチェックする く 5
r_cmtw_rx	1.20	CMT	
r_cmt_rx	3.00	Simpl	FITE52-U-77/U9-1/C2
r_dac_rx	2.91	Digita	C:¥Renesas¥e2_studio¥FITModules 参照 初期設定に戻す
r_amaca_rx	1.04	DIMA	
r_uic_rx	2.00	Clock	
T 21 U D D to .		CHER.K	OK キャンセル
モンユールの追加>>		L	
詳細			

2.2.3 モジュールの選択と詳細なフィルタの使用

使用可能なモジュール一覧で表示する FIT モジュールを制御できます。「フィルタの詳細設定」を選択(①) するとフィルタの詳細設定ダイアログが表示されます。フィルタのオプション(ファミリ、シリーズ、グルー プ等)は現在のプロジェクト用に設定されたデバイスになっています。例ではデバイスが "R5F564MLCxFC" を選択しているため、"RX"ファミリ、"RX600" シリーズ、"RX64M" グループに自動的に選択されます。

FITモジュールを追加す	¤ るプロジェクト:	rskrx64m_fitcfg_demo	1			 ・「」 コードの生) ・ ・ ・	₫ ▽ □ □
ターゲットボード RSK	RX64M	✓ フィルタの詳細設定				オプション	值
使用可能なモジュール	,			~ 選択したモジュール	, 		
モジュール	バージョン	■ フィルタの詳細設定		·1			
r bsp	3.40			+			
r byteg	1.60	- 使用可能なFITE:	ジュールの一覧	表示			
r can rx	2.10	 これらのオプションをも 	刃期設定から変更し;	た場合、デバイスが対応していな	にNFITモジュールも表	長示されます。	
r cmtw rx	1.20						
r cmt rx	3.00						
r_dac_rx	2.91	ファミリ RX	~ シリーズ	RX600 🗸	グループ	RX64M 🗸	
r_dmaca_rx	1.04	Tool chain Papara	· · · 御祭	指定したい。シン	マゴリケーション	援定したい	
r_dtc_rx	2.05	Nor chain Neriesa	3 * 100 %2	JAXEO/6V	1797-737	THAL COOL	
r eeprom spi	2.33						初期設定に
モジュールの追加>>							
	_						



2.2.4 バージョンの選択

デフォルトでは、インストールされた各 FIT モジュールの最新バージョンが、使用可能なモジュール一覧 に表示されます。しかし以前のバージョンを使用したい場合は、各モジュールの「バージョン」欄をクリッ クし、表示されたドロップダウンリストから選択します。

ターゲットボード RSKRX64M v <u>フィルタの詳細設定</u>							值
使用可能なモジュール					L .		
モジュール	バージ	概要	^	モジュール	バージ		
r bsp	3.40	Board Support Packages.					
r_byteg	1.60	Byte-based circular buffer library.					
r_can_rx	2.10	CAN API for the RX71M and RX64M.					
r_cmtw_rx	1.20	Y VTW Driver					
r_cmt_rx	1.10	mple CMT driver for creating timer tick.					
r_dac_rx	1.20	gital-to-Analog Converter driver					
r_dmaca_rx	1.04	DMAC driver					
e dta na	2.05	DTC Madula	×				
モジュールの追加>	>			<<モジュールの削	削除		
24 (m							
8千市山							
依存モジュール: r_	bsp バージョン	2.80, 2.90, 3.00, 3.01, 3.10, 3.20, 3.30, 3.40			^		

2.2.5 FIT モジュールの選択

- (1) 使用可能なモジュール一覧から使用する FIT モジュールを選択します。(①) 例では BSP と CMTW の FIT モジュールを選択します。
- (2) 「モジュールの追加>>」ボタンをクリックします。(②) 選択された FIT モジュールが、選択された FIT モジュール一覧に表示されます。(③)
- (3) 選択したモジュール一覧から FIT モジュールを削除する場合は、該当モジュールを選択し、「<<モ ジュールの削除」ボタンをクリックしてください。(④)

ターゲットボード R	SKRX64M	→ <u>フィルタの詳細設定…</u>					17737	18
使用可能なモジュー	-JV		1		選択したモジュール	(3)		
モジュール	バージ	概要	Ŭ	^	モジュール	1-9		
r_bsp	3.40	Board Support Packages.			r_bsp	3.40		
r_byteq	1.60	Byte-based circular buffer libra	ry.		r_cmtw_rx	1.20		
r_can_rx	2.10	CAN API for the RX71M and RX	64M.					
r cmtw rx	1.20	CMTW Driver						
r_cmt_rx	3.00	Simple CMT driver for creating	timer tick.					
r_dac_rx	2.91	Digital-to-Analog Converter dri	iver					
r_dmaca_rx	1.04	DMAC driver						
e dhe es	2.05	DTC Madula		*				
モジュールの追加	» (2)				<<モジュールの肖	除 (4)		
計判								



2.2.6 端子設定

端子設定が必要な FIT モジュールを使用する場合、使用する端子機能と割り当て先を選択することができます。

- (1) 選択したモジュール一覧から端子設定をする FIT モジュールを選択します。(①) 例では "r_cmtw_rx" を選択します。
- (2) 端子設定一覧から該当するチャネルのチェックボックスにチェックします。(2) 例では CMTW0 の チェックボックスにチェックします。
- (3) 使用する端子のポートを選択します。Unused を選択するとドロップダウンリストが表示され、使用するポートを選択します。(③) 例では TOC0 端子を "PC7"、TOC1 端子を "PE7" としています。

ターケットホード RSKF 使用可能なモジュール	₹X64M	────────────────────────────────────		選択したモジュール				2
モジュール	バージ	概要	^	モジュール		▼ TIC0端子	Unused	
r bsp	3.40	Board Support Packages		- then			Unused DC7	
r byteg	1.60	Byte-based circular buffer library.		r cmtw rx	1.20	▼ 1000端子	DE7	3
r can rx	2.10	CAN API for the RX71M and RX64M.						
r cmtw rx	1.20	CMTW Driver				▼ (L) CINTUR ▼ TIC2端子	Unused	
r cmt rx	3.00	Simple CMT driver for creating timer tick.				▼ TIC3端子	Unused	
r dac rx	2.91	Digital-to-Analog Converter driver				▼ TOC2端子	Unused	
r_dmaca_rx	1.04	DMAC driver				₣ TOC3端子	Unused	
r_dtc_rx	2.05	DTC Module						
r eeprom spi	2.33	Clock Synchronous Control Module for FER.	\checkmark					
モジュールの追加>>				<<モジュールの削	除			
詳細 依存モジュール: r_b: This software prov ticks. The sofware	sp バージョン vides a unit a also allov	/ 2.80, 2.90, 3.00, 3.01, 3.10, 3.20, 3.30, 3.40 fied, abstracted interface for setting up the CMTV /s output compare and input capture operations	V peri on Cl	ipheral generating MTW.	timer			



2.2.7 コードの生成

- (1) FIT コンフィギュレータのツールバーの「コードの生成」ボタンをクリックします。(①)
- (2) FIT コード生成ダイアログが表示されます。表示内容に問題がなければ「OK」ボタンをクリックしま す。(2)

🚆 FIT Configurator	23		🕒 コードの生成 🌣 🖓
FITモジュールを追加	するプロジェクト	י: rskrx64m_fitcfg_	 ④ FIT3-ド生成 - 内容 × 端子設定
ターゲットボード RS	SKRX64M	~ <u>2111/30</u>	以下のモジュールがインストールされます。 r_bsp (v3.40) 値
使用可能なモジュ・	-JL		r_cmtw_rx (v1.20)
モジュール	バージ	概要	以下のインクルードパスがプルジェクトの設定に追加されます。
r_bsp	3.40	Board Support Pa	/\${ProjName}/r_bsp
r_byteq	1.60	Byte-based circul	/\${ProjName}/r_config /\${ProjName}/r_cmtw_rx
r_can_rx	2.10	CAN API for the F	/\${ProjName}/r_cmtw_rx/src
r_cmtw_rx	1.20	CMTW Driver	/\${ProjName}/r_pincfg
r_cmt_rx	3.00	Simple CMT drive	
r_dac_rx	2.91	Digital-to-Analog	
r_dmaca_rx	1.04	DMAC driver	v
r_dtc_rx	2.05	DTC Module	
r eeprom spi	2.33	Clock Synchrono	続けるにはOKを、戻るにはキャンセルをクリックしてください。
モジュールの追加	>>		
詳細			2
			OK キャンセル

(3) 選択した FIT モジュールのコードが生成されます。生成されたコードはプロジェクト・エクスプロー ラの該当プロジェクトに追加されます。(③) 例では"r_bsp"、"r_cmtw_rx"、"r_pincfg"、"r_config" フォルダが追加されます。"r_pincfg"フォルダは、端子設定機能を設定することで生成され、端子機 能を有効にする関数が記述されているファイルが格納されています。"r_config"フォルダは、FIT モ ジュールを使用すると必ず生成され、各 FIT モジュールのコンフィギュレーションファイルが格納さ れています。



2.3 プロジェクトのビルド

ユーザプログラムを作成し、プロジェクトをビルドしてください。



2.4 トラブルシューティング

- (1) Q: FIT コンフィギュレータでコード生成した FIT モジュールを削除する方法は?
 - A:FIT モジュールを削除する場合は、下記の手順で削除できます。例では"r_cmtw_rx"を削除しま す。
 - FIT コンフィギュレータの選択したモジュール一覧で削除したい FIT モジュールを選択します。(①)
 - ・ 「モジュールの削除」ボタンでクリック(②)すると、選択された FIT モジュールは選択した モジュール一覧から削除されます。(③)





- ・ FIT コンフィギュレータのツールバーの「コードの生成」ボタンをクリックします。(④)
- FIT アンインストール内容ダイアグラムが表示されます。表示内容に問題がなければ「OK」 ボタンをクリックします。(⑤)

FITモジュールを追加す	するプロジェクト	rskrx64m	✓ ▼ 端子設定 4
ターゲットボード RSI 使用可能なモジュー	KRX64M Jl	~ 2	以下のモジュールはアンイソストールされます: r_cmtw_nx (v1.20) 以下のイングルードパイがプロジェクトの設定から削除されます。
モジュール	バージ	概要	/\${ProjName}/r_cmtw_rx
r bsp	3.40	Board Sup	/\$(ProjName)/r_cmtw_rx/src
r_byteq	1.60	Byte-base	/s(Projivame)/r_princing
r_can_rx	2.10	CAN API f	
r_cmtw_rx	1.20	CMTW Dri	
r_cmt_rx	3.00	Simple CM	
r_dac_rx	2.91	Digital-to-	
r_dmaca_rx	1.04	DMAC driv	
r_dtc_rx	2.05	DTC Modu	C
r eeprom spi	2.33	Clock Syne	アンインストールを実行するにはOKを、スキップするにはキャンヤルをクリックレ
モジュールの追加>	>		てくだきい。
詳細			6
Dependencies: The r_bsp packa and MCU inform folder contains f	ຊີປ ge provides nation for di files that are	a foundation fferent board common to a	OK キャンセル a MCU group. These files provide functionality such as easy register

 選択した FIT モジュールのコードが削除されます。(⑥) 例のようにプロジェクト・エクス プローラーから "r_cmtw_rx" と "r_pincfg" が削除されます。("r_pincfg" が削除されるのは "r_cmtw_rx" と依存関係にあるためです。)





(2) Q: 選択されたモジュール一覧にアイコンが表示されます。どうすればよいでしょうか?

A:選択されたモジュール間で矛盾が見つかった場合に、エラーや警告のアイコンが表示されます。 下記にアイコンの一覧表を示します。一覧表に記載されている解決方法を参照してください。

アイコン	エラー/警告の種類	説明と解決方法
۵	依存関係のエラー	依存関係のエラーは、示された項目に使われた 必要なモジュールが追加されていない場合に発 生します。
		<u>解決方法</u> :このエラーを解決するには、必要な FIT 依存モジュールを探して、現在の選択モ ジュールー覧に追加してください。
۵	依存関係の警告	依存関係の警告は、示された項目と選択された 依存関係のバージョン番号に不一致があった場 合に発生します。
		<u>解決方法</u> :この警告を解決するには、選択された依存関係のバージョン番号が示された項目の バージョン番号と一致するようにしてください。
		まったく同じバージョンが用意できない場合、 必要とされているバージョンより新しいバー ジョンを使用してください。
4	競合の警告	競合の警告は、FIT モジュールで使用する端子が 既に他の FIT モジュールやコード生成で設定し た機能により使用されている場合に発生しま す。
		<u>解決方法</u> :この警告を解決するには、競合して いる端子機能を他の端子に割り当ててくださ い。
後	依存関係と競合の警告	このアイコンは、選択された FIT モジュールで、 依存関係と競合の両方の問題が発生しているこ とを示します。この警告を解決するには、両方 の問題に対応する必要があります。

- (3) Q: FIT コンフィギュレータの使用可能なモジュールに FIT モジュールを追加したいのですが、どの ようにすればよいでしょうか?
 - A: FIT コンフィギュレータのドロップダウンメニュー▽の「ネットワークからモジュールをダウン ロードする」を選択してください。FIT モジュールのダウンロードダイアログが表示し、WEB か ら FIT モジュール情報を取得します。必要な FIT モジュールを選択しダウンロードしてください。 詳細は「2.1.2 FIT モジュールのダウンロード」の(3)~(5)を参照してください。
- (4) Q:端子設定機能を使用してコード生成しましたが、端子から波形が出力されません。
 - A:端子設定機能で生成された"r_pincfg"フォルダに"r_pinset.c"ファイルがあります。このファイ ル内に端子機能を有効にする関数(R_<周辺機能チャネルまたはFIT モジュール識別子>_PinSet_< 周辺機能チャネル>)が記述されています。この関数をご使用のユーザプログラムでコールしてく ださい。なおFIT モジュールで使用する端子機能を有効にするタイミングについては、各FIT モ ジュールのマニュアルを参照してください。

- (5) **Q**: e²studio プロジェクトを作成したときに、FIT モジュールのダウンロードをしませんでした。FIT モジュールを使用するためダウンロードをしたいのですが、どのようにすればよいでしょうか?
 - A: FIT モジュールをダウンロードするには、下記の手順でダウンロードすることができます。なお FIT モジュールが使用可能なプロジェクトは、プロジェクト作成時に FIT モジュールの使用を有 効にしたプロジェクトとなります。詳細は「2.1.1新規プロジェクトの作成」の(9)~(12)を参照し てください。
 - FIT コンフィギュレータを起動します。起動方法は「2.2.1FIT コンフィギュレータの起動」を 参照してください。
 - FIT コンフィギュレータのドロップダウンメニュー▽の「ネットワークからモジュールをダウンロードする」を選択してください。FIT モジュールのダウンロードダイアログが表示し、WEB から FIT モジュール情報を取得します。必要な FIT モジュールを選択しダウンロードしてください。詳細は「2.1.2 FIT モジュールのダウンロード」の(3)~(5)を参照してください。

		 <u>フィルタの詳細設定…</u> 				オプション	ネットワークからモジュールをダウンロードする
吏用可能なモジュール				選択したモジュール			
モジュール パ	(ージ	概要	^	モジュール	バージ		
r_bsp 3.	.40	Board Support Packages.					
r byteg 1.	.60	Byte-based circular buffer library.					
r_can_rx 2.	.10	CAN API for the RX71M and RX64M.					
r_cmtw_rx 1.	.20	CMTW Driver					
r_cmt_rx 3.	.00	Simple CMT driver for creating timer tick.					
r_dac_rx 2.	.91	Digital-to-Analog Converter driver					
r_dmaca_rx 1.	.04	DMAC driver					
r_dtc_rx 2.	.05	DTC Module					
r eenrom sni 2.	.33	Clock Synchronous Control Module for FEPROM Access	¥				
モジュールの追加>>				<<モジュールの削	除		
詳細							
					~		

3. 参考ドキュメント

テクニカルアップデート/テクニカルニュース (最新の情報をルネサスエレクトロニクス ホームページから入手してください。)

ユーザーズマニュアル:開発環境

CC-RX コンパイラ ユーザーズマニュアル (R20UT3248JJ)

(最新の情報をルネサスエレクトロニクス ホームページから入手してください。)

e²studio 統合開発環境 ユーザーズマニュアル 入門ガイド (R20UT2858JJ) (最新の情報をルネサスエレクトロニクス ホームページから入手してください。)

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/

お問合せ先 <u>http://japan.renesas.com/contact/</u>



╕╆┋┰┋╕ᡬᢋ	RX ファミリ e ² studio に組み込む方法
[汉吉] 吉仁 亚米	Firmware Integration Technology

	1	1					
Pov	<u> ※</u> 行口	改訂内容					
		ページ	ポイント				
1.00	2013.07.17	_	初版発行				
1.10	2013.11.15	5	e ² studio プラグインの更新に伴う修正				
1.11	2014.10.15		3.1 FIT モジュールのインストール				
		5	説明追加				
			3.2 プラグインを使用して FIT モジュールを組み込みます				
		6	説明追加				
			3.3 FIT プラグインメッセージ				
		8	説明追加				
1.20	2017.03.31	—	FIT コンフィギュレータ機能追加による全面改訂				
1.21	2017.06.30		2.1.2 FIT モジュールのダウンロード				
		12	説明追加				
		24	2.4 トラブルシューティングの追加				
		25	3. 参考ドキュメントの修正				

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。

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、本ドキュメントおよびテクニカルアップデートを参照してください。

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(予約領域)のアクセス禁止

【注意】リザーブアドレス(予約領域)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(予約領域)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、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違うと、内部ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品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・システムの設計におい 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て生じた損害(お客様 または第三者いずれに生じた損害も含みます。以下同じです。)に関し、当社は、一切その責任を負いません。
2.	当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の 知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。 当社は、本資料に基づき当社または第三者の特許権、業件権その他の知め財産権を何に許諾するものではなりません。
4.	当社製品を、全部または一部を問わず、改造、改変、複製、その他の不適切に使用しないでください。かかる改造、改変、複製等により生じた損害に関し、当社 は、一切その責任を負いません。
5.	当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。 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、エド破滅、ハーフナル破砕、座来用ロホラト等 高品質水準: 輸送機器(自動車、電車、船舶等)、交通制御(信号)、大規模通信機器、 金融端末基幹システム、各種安全制御装置等
	当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム(生命維持装置、人体に埋め込み使用するもの等)、もしくは多大な物的損害を発生させ るおそれのある機器・システム(宇宙、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等)に使用されることを意図 しておらず、これらの用途に使用することはできません。たとえ、意図しない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負い ません。
6.	当社製品をご使用の際は、最新の製品情報(データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使 用上の一般的な注意事項」等)をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指 定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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。
7.	当社は、当社製品の品質および信頼性の向上に努めていますが、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を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を 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、お客様の機器・システムとしての 出荷保証を行ってください。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、お客様の機器・システムとしての安全検証をお客様の責任で行ってく ださい。
8.	当社製品の環境適合性等の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。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。
9.	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・システムに使用することはできません。また、当社製品および技術 を、(1)核兵器、化学兵器、生物兵器等の大量破壊兵器およびこれらを運搬することができるミサイル(無人航空機を含みます。)の開発、設計、製造、使用もし くは貯蔵等の目的、(2)通常兵器の開発、設計、製造または使用の目的、または(3)その他の国際的な平和および安全の維持の妨げとなる目的で、自ら使用せず、か つ、第三者に使用、販売、譲渡、輸出、賃貸もしくは使用許諾しないでください。
10	当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。 お客様の転売 貸与等により 本書(本ご注音書きを含みます) 記載の誘条件に抵触して当社製品が使用され その使用から損害が生じた場合 当社は一切その
11.	この音味のねれ、質子等により、不習、不正正志習をと言いより。) 記載の語本件にきねじて当れるにかっていたが、ての使用からし言か上した場合、当社は、すての 責任を負わず、お客様にかかる使用に基づく当社への請求につき当社を免責いただきます。 .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禁じます。
12. 注1	. 本資料に記載された情報または当社製品に関し、ご不明点がある場合には、当社営業にお問い合わせください。 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 またまたは問題に保まする人体でいいます。
注2	但按または间接に味有する会社をいいます。 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。

(Rev.3.0-1 2016.11)

RENESAS

ルネサスエレクトロニクス株式会社

http://www.renesas.com

※営業お問合せ窓口の住所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。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24 (豊洲フォレシア)

■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:https://www.renesas.com/contact/

■営業お問合せ窓口