RENESAS

RH850 ファミリ	リアルタイム OS RI850V4 V2.01.00
リリースノート	

R20UT3550JJ0100 Rev.1.00 2015.09.30

この度は、RH850 ファミリ リアルタイム OS RI850V4 をご使用いただきまして、誠にありがとうございます。 本資料では、本製品をお使いいただく上での制限事項および注意事項を記載しております。ご使用の前に、必ずお読みく ださいますようお願い申し上げます。

1.		製品構成	4
2.		ユーザーズマニュアルについて	5
3.		対象デバイスについて	6
4.		動作環境	7
	4.1.	ハードウェア環境	7
	4.2.	ソフトウェア環境	7
	4.3.	対応ツール	7
5.		インストール時の注意事項	8
	5.1.	インストール時の注意事項	8
		5.1.1. 管理者権限に関する注意事項	8
		5.1.2. 実行環境に関する注意事項	8
		5.1.3. ネットワーク・ドライブに関する注意事項	8
		5.1.4. インストール先フォルダ名に関する注意事項	8
		5.1.5. 機能の変更や修復に関する注意事項	8
		5.1.6. インストール・フォルダの変更に関する注意事項	9
		5.1.7. インストールするバージョンに関する注意事項	9
		5.1.8. インストーラの起動に関する注意事項	9
		5.1.9. プラグインの有効化	9
	5.2.	アンインストール時の注意事項	10
		5.2.1. 管理者権限に関する注意事項	10
		5.2.2. アンインストールのフォルダに関する注意事項	10
		5.2.3. インストーラ以外での追加/修正に関する注意事項	10
		5.2.4. アンインストール時の選択キーワード	10
6.		前リリース版との相違点	11

	6.1.	カーネルの相違点	11
	6.2.	コンフィギュレータの相違点	13
	6.3.	リアルタイム OS ビルド設定プラグインの相違点	14
	6.4.	リアルタイム OS リソース情報表示プラグインの相違点	15
	6.5.	リアルタイム OS タスク・アナライザ・プラグインの相違点	15
7.		RI850V4(V1)との相違点	16
	7.1.	RH850 対応	16
	7.2.	カーネルの変更点	16
	7.3.	コンフィギュレータの変更点	18
	7.4.	リアルタイム OS タスク・アナライザ対応	19
8.		注意事項	22
	8.1.	CS+ for CC のプロジェクト作成	22
		8.1.1. 本製品添付のサンプル・プロジェクトを使用する	22
		8.1.2. 新しいプロジェクトを流用する	
		8.1.3. RI850V4 のプロジェクトを流用する	
		8.1.4. マルチコア用の RI850V4 プロジェクトを作成する	
	8.2.	プラグインの有効化	29
	8.3.	コンフィギュレーション・ファイルのプロパティ	30
	8.4.	シミュレータ使用時の注意事項	32
		8.4.1. シミュレータ設定ファイルについて	32
		8.4.2. シミュレータを使用した場合の OS タイマについて	32
	8.5.	リアルタイム OS リソース情報表示パネルに関する注意事項	32
		8.5.1. 参照はリアルタイム OS 初期化後に行う	32
		8.5.2. デバッグ情報を生成したプログラムを使用する	32
	8.6.	リアルタイム OS タスク・アナライザに関する注意事項	33
		8.6.1. トレース・モード変更	33
		8.6.2. デバッグ・ツールの設定	
		8.6.3. トレースのタイムスタンプについて	
	8.7.	カーネル・ソース・コードのビルド方法	37
	8.8.	可変リセットベクタ機能を利用する場合の注意事項	37
		8.8.1. 可変リセットベクタ機能を利用し,かつ RBASE 相対の割り込み管理を行う方: 38	法について
	8.9.	E1 エミュレータを使用して「ハードウェア・トレース・モードでトレース・チャートを	取得する」
	を選	択した場合の注意事項	39
	8.10.	I/O ヘッダ・ファイル生成ツールを使用する場合の注意事項	40
	8.11.	CPU クロックの周波数について	40
9.		制限事項	41

9.1.	CS+ for	r CC 使用時の制限事項	41
	9.1.1.	リアルタイム OS ビルド設定プラグイン	41
	9.1.2.	リアルタイム OS 解析制御プラグイン	42
	9.1.3.	リアルタイム OS リソース情報表示プラグイン	42
	9.1.4.	リアルタイム OS タスク・アナライザ・プラグイン	42



1. 製品構成

RI850V4 は型名により、契約形態と提供物が異なります。

型 名 契約形態		提供物
RTRRH8500TR01ERR	評価契約,インストール可能な PC は 1 台	А
RTRRH8500TR01ERRLU	評価契約,インストール可能な PC は無制限	А
RTRRH8500TR01RRRUL	量産契約,量産数は 3000 台まで	А
RTRRH8500TR01RRRUU	量産契約、量産数は無制限	А
RTRRH8500TR01SRRUU	量産契約,量産数は無制限,ソース・コード付き	В

提供物は以下となります。

※ 本文書では、V850 用 RI850V4 を「RI850V4 (V1)」, RH 用 RI850V4 を「RI850V4RH」と記載します。

※ 「RI850V4」とのみ書かれてある箇所は、RI850V4(V1)、RI850V4RH 共通の内容を意味します

提供物		ツール名	バージョン
	リアルタイム OS RI850V4 カーネル・オブジェクト		V2.01.00
		コマンドライン・コンフィギュレータ CF850V4	V2.01.01.01
		CS+ for CC 用プラグイン	
		リアルタイム OS ビルド設定プラグイン(共通部)	V3.02.01.01
		リアルタイム OS ビルド設定プラグイン(RI850V4RH 依存部)	V2.01.01.01
В	А	リアルタイム OS 解析制御プラグイン(共通部)	V3.00.00.03
		リアルタイム OS 解析制御プラグイン(ulTRON4 依存部)	V3.02.00.01
		リアルタイム OS 解析制御プラグイン(RI850V4RH 依存部)	V2.01.00.03
		リアルタイム OS リソース情報表示プラグイン(共通部)	V3.01.00.01
		リアルタイム OS リソース情報表示プラグイン(ulTRON4 依存部)	V3.01.01.01
		リアルタイム OS タスク・アナライザ・プラグイン(共通部)	V3.00.02.01
		リアルタイム OS タスク・アナライザ・プラグイン(パネル部)	V3.00.00.03
		リアルタイム OS タスク・アナライザ・プラグイン(RI850V4RH 依存部)	V2.01.00.03
		リアルタイム OS RI850V4 カーネル ソース・コード	V2.01.00



2. ユーザーズマニュアルについて

本製品に対応したユーザーズマニュアルを以下に示します。本文書と合わせてお読みください。

マニュアル名	資料番号
RI シリーズ リアルタイム・オペレーティング・システム	R20UT0751.LI0105
ユーザーズマニュアル 起動編	120010101000100
RI850V4 V2 リアルタイム・オペレーティング・システム	
ユーザーズマニュアル コーディング編	R20012889JJ0101
RI850V4 V2 リアルタイム・オペレーティング・システム	
ユーザーズマニュアル デバッグ編	R20012090JJ0101
RI850V4 V2 リアルタイム・オペレーティング・システム	D2011T2801110101
ユーザーズマニュアル 解析編	R20012091JJ0101
RI シリーズ リアルタイム・オペレーティング・システム	
ユーザーズマニュアル メッセージ編	K20010756JJ0105

なお, ユーザーズマニュアルは PDF ファイルで提供媒体にパッケージされています。またルネサスエレクトロニクス のホームページから入手することができます。



3. 対象デバイスについて

本製品は、以下のデバイスに対応しています。

● RH850 (G3K コア, G3M コア) ファミリ



4. 動作環境

本製品を使用するには、次の環境が必要になります。

4.1. ハードウェア環境

次のハードウェア環境に対応しています。

- プロセッサ:1GHz 以上(ハイパー・スレッディング,マルチコア CPU に対応)
- メモリ容量:推奨 2GB 以上。最低 1GB 以上(64 ビット版 Windows® 7 および Windows® 8 では 2GB 以上)
- ディスプレイ: 1024×768 以上の解像度, 65536 色以上

4.2. ソフトウェア環境

次のソフトウェア環境に対応しています。

- Windows® 7, Windows® 8 および Windows® 8.1, Windows® 10 (32bit 版, 64bit 版)
- .NET Framework 4 + 言語パック
- Microsoft Visual C++ 2010 SP1 ランタイム・ライブラリ
- Internet Explorer 8.0 以上

いずれの場合も、最新の Service Pack がインストールされていることを推奨します。

4.3. 対応ツール

本製品は次の開発ツールに対応しています。

ツール名	提供元	バージョン
統合開発環境 CS+ for CC	ルネサス エレクトロニクス	V3.02.00 以降
C コンパイラ CC-RH	ルネサス エレクトロニクス	V1.03.00 以降
統合開発環境 MULTI	米国 Green Hills Software, Inc.	V6.1.6 Rel9.4.0 以降



5. インストール時の注意事項

本章では、インストール、アンインストール時の注意事項について説明します。

5.1. インストール時の注意事項

5.1.1. 管理者権限に関する注意事項

インストール(コピー)するには、Windows®の管理者権限が必要です。

5.1.2. 実行環境に関する注意事項

Windows®には, .NET Framework と Visual C++ のランタイム・ライブラリがインストールされている必要があります(CS+ for CC を実行するために必要です)。

5.1.3. ネットワーク・ドライブに関する注意事項

ネットワーク・ドライブからのインストールはできません。また,ネットワーク・ドライブへのインストールもで きません。

5.1.4. インストール先フォルダ名に関する注意事項

インストール先フォルダ名に指定可能な文字は、Windows®に準じます。 /*:<>?|"¥;, の 11 文字は使用できません。また、空白文字ではじまるものと空白文字で終わるものは指定できません。

指定する際に、絶対パスで指定し、相対パスでは指定しないでください。

また、インストール先フォルダの区切り子には ¥ を使用してください。/ は使用しないでください。

5.1.5. 機能の変更や修復に関する注意事項

インストール済みのツールに対して、機能の変更や修復を行う場合は、そのツールのインストール・パッケージを 用意し、インストール用プログラムを実行すると起動する、プログラムの保守画面で「変更」または「修復」を実行 してください。

「プログラムと機能」の[変更]ボタンから行うとエラーになります。



5.1.6. インストール・フォルダの変更に関する注意事項

インストール後にできる次のフォルダ(含むフォルダ以下のファイル)には、ツールが動作するために必要なファ イル類がありますので削除しないでください。

- Windows®が 32bit 版で、システムドライブが C:の場合 C:¥Program Files¥Common Files¥Renesas Electronics CubeSuite+¥
- Windows®が 64bit 版で,システムドライブが C:の場合 C:¥Program Files (x86)¥Common Files¥Renesas Electronics CubeSuite+¥

5.1.7. インストールするバージョンに関する注意事項

新しいバージョンがインストールされている場合には、古いバージョンがインストールされない可能性があります。

5.1.8. インストーラの起動に関する注意事項

日本語版以外の Windows®で、インストーラを起動するパスに多バイト文字が含まれているとエラーとなりイン ストールを実行することができません。

5.1.9. プラグインの有効化

本製品のインストール直後など、本製品のプラグインが無効になっている場合があります。「8.2 プラグインの有 効化」にしたがって本製品のプラグインを有効にしてください。



5.2. アンインストール時の注意事項

5.2.1. 管理者権限に関する注意事項

アンインストール(フォルダ/ファイル削除)するには、Windows®の管理者権限が必要です。

5.2.2. アンインストールのフォルダに関する注意事項

ツールのアンインストールの実行順序によっては、フォルダが完全に削除されない場合があります。この場合、ア ンインストールした後に残ったフォルダは、エクスプローラ等で削除してください。

5.2.3. インストーラ以外での追加/修正に関する注意事項

ツール,および,マニュアル類をインストールしたフォルダに,本製品のインストーラ以外の手段によって,追加 または修正されたファイルは,アンインストール時に削除できません。

5.2.4. アンインストール時の選択キーワード

本製品をアンインストールする場合は、2つの方法があります。

- 統合アンインストーラを使用する(CS+ for CC 自体をアンインストールする)
- 個別にアンインストールする(本製品のみをアンインストールする)

個別にアンインストールを行なう場合、コントロールパネルの

● 「プログラムと機能」

から、以下を削除してください。

- CS+ Realtime OS Common Plugins
- CS+ Realtime OS RI850V4RH Plugins
- CS+ Realtime OS RI850V4RH Object Release(量産契約, ソース・コード付き「以外」の場合)
- CS+ Realtime OS RI850V4RH Source Release(量産契約,ソース・コード付きの場合)



6. 前リリース版との相違点

6.1. カーネルの相違点

(1) カーネルのバージョン情報

カーネルのバージョン情報は以下のようになります。

項目	変更前	変更後
TKERNEL_PRVER	0x0201	0x0210

- (2) RH850 G3M コアに対応
 RI850V4 V2.00.01 では RH850 G3K コアにのみ対応していましたが, RI850V4 V2.01.00 では新たに RH850 G3M
 コアに対応しました。
- (3) ハードウェア浮動小数点演算機能に対応

RI850V4 V2.01.00 では新たにハードウェア浮動小数点演算機能に対応しました。 このため、システム・コンフィギュレーション・ファイルにおいて、以下の定義が追加となります。

DEF_FPSR (*fpsr*) // Initial FPSR value

(4) 米国 Green Hills Software 社製開発環境 MULTI に対応
 RI850V4 V2.01.00 では新たに、米国 Green Hills Software 社製開発環境 MULTI に対応しました。

(5) OS/コンフィギュレータ(CF850V4)の適合検知機能を追加 RI850V4 V2.01.00 では複数のカーネル・ライブラリが存在するため、ロード・モジュールにリンクしたカーネ ル・ライブラリと CF850V4 が出力する情報ファイルが不整合であった場合は、リンクエラーが発生する機能を 新たに追加しました。 下記のリンクエラーが発生した場合は、システム・コンフィギュレーション・ファイルの設定が正しいか確認を

下記のリングエラーが発生した場合は、システム・コンフィキュレーション・ファイルの設定が正しいが確認を 行ってください。



● コンパイラ CC-RH を使用している場合

エラーメッセージ(CC-RH)	意味
E0562310:Undefined external symbol	CF850V4 が出力する情報ファイル(xxx)は FPU 非
"kernel_confirm_target_fpunoloading" referenced in	対応版のカーネル・ライブラリを対象としています
XXX.	が,FPU 対応版のカーネル・ライブラリがリンクさ
	れているためリンクエラーとなっています。
E0562310:Undefined external symbol	CF850V4 が出力する情報ファイル(xxx)は FPU 対
"kernel_confirm_target_fpuloading" referenced in xxx.	応版のカーネル・ライブラリを対象としています
	が,FPU 非対応版のカーネル・ライブラリがリンク
	されているためリンクエラーとなっています。

● コンパイラ CCV850 を使用している場合

エラーメッセージ(CCV850)		意味
[elxr] (error) unresolved symbols: 1		CF850V4 が出力する情報ファイル (xxx)は FPU 非対
kernel_confirm_target_fpunoloading	from xxx	応版のカーネル・ライブラリを対象としています
		が, FPU 対応版のカーネル・ライブラリがリンクさ
		れているためリンクエラーとなっています。
[elxr] (error) unresolved symbols: 1		CF850V4 が出力する情報ファイル (xxx)は FPU 対応
kernel_confirm_target_fpuloading	from xxx	版のカーネル・ライブラリを対象としていますが、
		FPU 非対応版のカーネル・ライブラリがリンクされ
		ているためリンクエラーとなっています。

(6) 構造体パッキング機能を使用可能

RI850V4 V2.00.01 では、構造体パッキング機能(CC-RH の-Xpack オプション, CCV850 の-pack オプション) を使用することができませんでしたが、RI850V4 V2.01.00 では構造体パッキング機能を使用できるように変更 しました。

(7) RI850V4 を組み込む対象のプロセッサエレメント(PE)を指定可能

RI850V4 V2.00.01 では, RI850V4 を組み込む対象となる PE を指定することができないため, RI850V4 V2 を組 み込む対象の PE はメイン CPU に限られていました。RI850V4 V2.01.00 では, コンフィギュレータ CF850V4 の起動オプション –peid を指定することにより, RI850V4 を組み込む対象となる PE を指定することが可能で す。このため, RI850V4 をパフォーマンス CPU に組み込むことや, 同時に複数の PE に組み込むことが可能で す。

(8) カーネル・ライブラリの追加と格納先の変更

RI850V4 V2.01.00 では, FPU 機能に対応したカーネル・ライブラリ(libri_fpu)を追加しました。また, カー ネル・ライブラリを格納するフォルダパスが以下のように変更になりました。

・変更前(RI850V4 V2.00.01): <インストール・フォルダ> ¥library¥rh850g3k_ccrh¥r32
 ・変更後(RI850V4 V2.00.01): <インストール・フォルダ> ¥library¥rh850_ccrh¥r32
 CS+を使用する場合は、本変更点の影響はありません。



6.2. コンフィギュレータの相違点

RI850V4 V2.01,00 では、起動オプションおよびシステム・コンフィギュレーション・ファイルの記述について

変更点があります。

起動オプションに関する主な変更点は以下になります。

(1) 【追加】起動オプション -peid=<*id*> を追加
 RI850V4 V2.01.00 では、新たに起動オプション -peid=<*id*>を追加しました。本起動オプションにより、RI850V4
 を組み込む対象となる PE 番号を指定することが可能です。これにより、CF850V4 はパフォーマンス CPU を対象とした情報ファイルの生成が可能となります。
 【指定例】

-peid=3

- (2) 【変更】起動オプション -cpu △<*name*> に G3M コアのデバイス・ファイルを指定可能
 RI850V4 V2.01.00 では RH850 G3M コアに対応したため、-cpu △ <*name*> 起動オプションに、G3M コアの
 デバイス・ファイルを指定することが可能となりました。
- (3) 【変更】起動オプション -t △<TOOL name> に "GHS"を指定可能
 RI850V4 V2.01.00 では米国 Green Hills Software 社製開発環境 MULTI に対応したため、起動オプション -t △
 <TOOL name>に"GHS"を指定することが可能となりました。

システム・コンフィギュレーション・ファイルの記述に関する主な変更点は以下になります。

(1) 【追加】初期 FPSR レジスタ情報 DEF_FPSR
 タスク、割り込みハンドラ、周期ハンドラ、初期化ルーチン、アイドルルーチンの初期起動時に、FPSR に設定
 する初期値を定義します。
 【指定例】

DEF_FPSR (0x00020000);

- (2) 【変更】CPU 種別 chip_type に "G3M"を指定可能
 RI850V4 V2.01.00 では RH850 G3M コアに対応したため, chip_type に G3M を指定することが可能になりました。
- (3) 【変更】最大割り込み優先度 MAX_INTPRI に "INTPRI15" まで指定可能 RI850V4 V2.01.00 では RH850 G3M コアに対応しました。起動オプション-cpu △<name>に G3M のデバイス・ ファイルを指定した場合、もしくは CPU 種別 chip_type に "G3M"を指定した場合は、最大割り込み優先度 MAX_INTPRI に INTPRI0 から INTPRI15 までを指定することが可能になりました。

6.3. リアルタイム OS ビルド設定プラグインの相違点

リアルタイム OS ビルド設定プラグインの主な変更点は以下の通りです。

(1) 【追加】システム・コンフィギュレーション・ファイルのプロパティに「コンフィギュレータ起動設定」を追加

RI850V4 V2.01.00 では, コンフィギュレータの起動オプション -peid=<*id*> を追加しました。本起動オプショ ンを GUI 上から設定するインタフェースとして, システム・コンフィギュレーション・ファイルのプロパティ に「コンフィギュレータ起動設定」を新たに追加しました。

図 6-1 システム・コンフィギュレーション・ファイルのプロパティ

プロパティ		×
og sysiste のプロパティ		P -+
⊿ システム情報テーブル・ファイル		
ファイルを生成する	(はLV(ctgファイル変更時に更新する)(-i)	
出力フォルダ	%BuildModeName%	
ファイル名	sits	
〃 システム情報ヘッダ・ファイル		
ファイルを生成する	(はい(cfeファイル変更時に更新する)(-d)	
出力フォルダ	%BuildModeName%	
ファイル名	kemel_idh	
▲ エントリ・ファイル		
ファイルを生成する	(はい(cfgファイル変更時に更新する)(-e)	
出力フォルダ	%BuildModeName%	
771ル名	entry.s	
出力方式	直接べかり方式	
例外ハンドラ・ベクタ・アドレスを指定する	CONT	
▲ コンフィギュレータ起動設定		
ユーザ指定オブション	-peid=1	
コーザ指定オプション 任意のオブションを指定する際に、オブション文平列を直接記述します。		
システム・エノフィキュレーション・ファイル関連情報(ファイル情報)		-



6.4. リアルタイム OS リソース情報表示プラグインの相違点

リアルタイム OS リソース情報表示プラグインの主な変更点は以下の通りです。

(1) 【追加】セクションのスキップ機能に対応

RI850V4 V2.01.00 対応のリソース情報表示プラグインでは、デバッガの「ステップ・イン実行で指定セクション をスキップする」機能を利用して、リアルタイム OS のコードをデバッグ対象としないように設定します。リソー ス情報プラグインはデバッガの [プロパティ] パネルの「デバッグ・ツール設定」タブ-「実行制御」-「指定セクシ ョン」に、下記セクション名を登録します。

RTOS リソース情報プラグインが登録するセクション名称一覧 .kernel system

(2) 【追加】リアルタイム OS 資源変更機能を追加

RI850V4 V2.01.00 対応のリソース情報表示プラグインでは、ユーザ・アプリケーションのソース・コードを変更 することなく、RI850V4 が管理する資源の変更を行うための機能として、リアルタイム OS 資源変更機能を追加し ました。

6.5. リアルタイム OS タスク・アナライザ・プラグインの相違点

リアルタイム OS タスク・アナライザ・プラグインの主な変更点は以下の通りです。

(1) 【変更】E1/E20エミュレータを使用している場合は「ハードウェア・トレース・モードで, トレース・チャ ートを取得」が利用できない制限を解除

前版ではデバッガがハードウェア・トレース・モードに対応していないため、E1/E20 エミュレータを使用している場合は「ハードウェア・トレース・モードで、トレース・チャートを取得」を利用することができませんでしたが、RI850V4 V2.01.00 ではデバッガがハードウェア・トレース・モードに対応したため、利用可能となりました。

ただし、取得できるトレース・データの容量が少ないため、使用には注意が必要です。詳細については E1/E20 エミュレータのユーザーズマニュアルを参照してください。

- (2) 【変更】リアルタイム OS タスク・アナライザ・プラグイン内で例外が発生する不具合を改修 トレース機能が存在しないデバイスに LPD 接続しようとしたときに、リアルタイム OS タスク・アナライザ・プ ラグイン内で例外(F115FFFF)が発生する不具合を本版で改修しました。
- (3) 【変更】流用プロジェクト作成時に、タスク・アナライザ・プラグインの[プロパティ]パネルの情報が復帰されない 不具合を改修

流用プロジェクトの作成時に、タスク・アナライザ・プラグインの[プロパティ]パネルの情報が復帰されず、デフォルトの値となってしまう不具合を本版で改修しました。

7. RI850V4 (V1) との相違点

7.1. RH850 対応

RI850V4 が RH850 に対応しました(V2.00.00 以降)。これに伴い, CS+ for CC 上で CC-RH と RI850V4(RI850V4RH) の組み合わせによるプロジェクトを作成できるようになりました。

7.2. カーネルの変更点

(1) カーネル処理中の割り込み排他方法の変更

RI850V4 (V1)では、カーネル処理中は PSW.ID の値を1にして全マスカブル割り込みの受付禁止状態とする 方法で、割り込み排他制御を実現していました。本製品では、割り込み優先度マスクレジスタ (PMR)を用い て、一定の優先度以下の割り込みを受付禁止とする割り込み排他制御方法に変更しています。PMR に設定する 値を最大カーネル管理内割り込み優先度 (maxintpri) と呼びます。maxintpri より大きい割り込みはカーネル処 理中も受け付けられますので、高速な割り込み応答が可能です。

maxintpriより大きい優先度の割り込み(カーネル管理外割り込み)では、サービス・コールの呼び出しはでき ません。またカーネル管理外割り込みに対応するハンドラについてはカーネルが介在しないため、分岐処理を 独自で記述する必要があります。

maxintpri 以下の優先度の割り込み(カーネル管理内割り込み)も含めた全ての割り込みの優先度については、 カーネルは設定しませんので、直接割り込みコントローラの ICxxx レジスタを操作してください。コンフィギ ュレーション・ファイルにて登録する割り込みハンドラについては、すべてカーネル管理内割り込みとして適 切な優先度を設定してください。基本クロック用タイマ割り込みについても同様です。

maxintpri はシステム・コンフィギュレーション・ファイルで指定することができます。

(2) loc_cpu∕iloc_cpu

マスカブル割り込みの受付状態の制御は、PMR を操作する方法に変更しています。従って、loc_cpu/iloc_cpu を発行した場合、カーネル管理内割り込みの受付を禁止します。

これに伴い、カーネルは IMRm および ICxxx レジスタの操作を行わなくなります。



(3) 例外ハンドラ・アドレス変更機能への対応

RH850 では例外ハンドラ・アドレスを設定に応じて変更が可能です。リセットと例外は、PSW.EBV ビットと、 RBASE レジスタ、EBASE レジスタによって例外ハンドラ・アドレスの基準位置が変更できる「直接ベクタ方 式」によって例外ハンドラ・アドレスを決定します。さらに割り込みは、チャネルごとに「直接ベクタ方式」 と「テーブル参照方式」の選択が可能です(これらの機能の詳細については、ハードウェアのマニュアルを参 照してください)。

本製品は、直接ベクタ方式/テーブル参照方式に対応した割り込みエントリ・ファイルを出力します。どちら の方式のエントリ・ファイルを出力するかどうかは、コンフィギュレーション・ファイルのプロパティで決定 します(決定方法の詳細については「8.3 コンフィギュレーション・ファイルのプロパティ」を参照してくだ さい)。直接ベクタ方式を選択した場合は、エントリ・ファイルには例外ハンドラ・ベクタを出力します。テ ーブル参照方式を選択した場合は、エントリ・ファイルには割り込みハンドラ・アドレス・テーブルを出力し ます。

エントリ・ファイルはいずれかの方式でのみの出力となるため、テーブル参照方式を選択する場合は、アプリ ケーションで使用するすべての割り込みチャネルに対しテーブル参照方式とすることを推奨します。割り込み チャネルごとの割り込みベクタ方式は、カーネルは設定しませんので、直接割り込みコントローラの ICxxx レ ジスタを操作してください。

(4) 割り込み管理機能サービス・コール以下のサービス・コールは未サポートとなります。

dis_int/ena_int/chg_ims/ichg_ims/get_ims/iget_ims

- (5) CPU 例外ハンドラCPU 例外ハンドラの静的生成は未サポートとなります。
- (6) タスク例外処理機能 タスク例外処理ルーチン、およびタスク例外処理機能は未サポートとなりました。以下のタスク例外処理機能 サービス・コールは使用できません。

ras_tex/iras_tex/dis_tex/ena_tex/sns_tex/ref_tex/iref_tex

(7) レジスタモード
 従来は 22/26/32 レジスタモードに対応していましたが、32 レジスタモードのみとなります。



7.3. コンフィギュレータの変更点

システム・コンフィギュレーション・ファイルの記述について主な変更点は以下になります。

(1) 【追加基本情報】最大カーネル管理内割り込み優先度 maxintpri

最大カーネル管理内割り込み優先度を指定します。本定義は静的 API 情報より前に記述してください。

記述形式:

MAX_INTPRI (maxintpri);

maxintpri として指定可能な値は以下のようになります。本定義の省略時は INTPRIO が指定されたものとして 解釈します。

- RH850 G3K コアの場合:「INTPRI0(最高優先度), INTPRI1, INTPRI2, INTPRI3, INTPRI4, INTPRI5, INTPRI6, INTPRI7(最低優先度)」
- RH850 G3M コアの場合:「INTPRI0(最高優先度), INTPRI1, INTPRI2, INTPRI3, INTPRI4, INTPRI5, INTPRI6, INTPRI7, INTPRI8, INTPRI9, INTPRI10, INTPRI11, INTPRI12, INTPRI13, INTPRI14, INTPRI15(最低優先度)」

定義例: MAX_INTPRI (INTPRI3);

- (2) 【変更】RI シリーズ情報
 RI_SERIES で指定可能な rtos_ver は「V2xy」に限られます。ここで指定する値はリアルタイム OS のバージョン番号の上位 3 数字を指定します。例えばバージョン番号が「V2.01.02」の場合、上位 3 数字「2.01」を使用して「V201」と指定します。
- (3) 【変更】CPU 種別 chip_type
 CPU_TYPE で chip_type として指定可能な値は「G3K」または「G3M」に限られます。
- (4) 【変更】基本クロック用タイマ割り込みの割り込み要因コード tim_intno CLK_INTNO で tim_intno として指定可能な値は、デバイス・ファイルで規定されている割り込み要因名、また は割り込み要因コード(0x1000~0x11ff)に限られます(ただし、存在しない割り込みの割り込み要因コード を指定した場合はエラーとなります)。
- (5) 【変更】最大割り込みハンドラ数 maxint, 最大割り込み要因コード maxintno
 MAX_INT で maxint として指定可能な値は「0x0~0x200」に, maxintno として指定可能な値は「0x1000~
 0x11ff」に限られます。ただし、この上限値は RH850 のアーキテクチャ上の最大値であり、使用するデバイスのデバイス・ファイルで規定されている割り込み要因の総数により上限値が変動します。
- (6) 【変更】割り込みハンドラ情報
 DEF_INH で inhno として指定可能な値は「0x1000~maxintno」範囲に限られます(ただし,存在しない割り込みの割り込み要因コードを指定した場合はエラーとなります)。

(7) 【削除】システム情報・静的 API 情報 以下は未サポートとなります。

REGMODE / DEF_TEX / DEF_EXC

7.4. リアルタイム OS タスク・アナライザ対応

RI850V4 (V1) では, リアルタイム OS アプリケーションの解析機能として「AZ850V4」を提供していましたが, RI850V4RH からは「リアルタイム OS タスク・アナライザ」を提供します(AZ850V4 は提供しません)。リアルタイ ム OS タスク・アナライザは, CS+ for CC 用プラグインとして提供します。詳しい使用方法は「RI シリーズ リアル タイム・オペレーティング・システム ユーザーズマニュアル 解析編」を参照してください。



図 7-1 AZ850V4 のイメージ図

図 7-2 リアルタイム OS タスク・アナライザのイメージ図



また、リアルタイム OS タスク・アナライザの対応に伴い、プロジェクト・ツリーに「リアルタイム OS 関連ファイル」というカテゴリを追加し、その中にトレース情報ファイル(trcinf.c)を登録します。このファイルは読み込み専用

ファイルで,ユーザが変更する必要のないファイルですが,アプリケーションとともにビルドして組み込む必要がある ファイルです。



図 7-3 リアルタイム OS 関連ファイル

またトレース情報ファイルでは、インクルードしているファイルの関係で、CC-RH ビルド・ツールのプロパティにある「ビルド方法」カテゴリの「インクルード・ファイルが存在しないソースの扱い」を「再コンパイル/アセンブルしない」としてください。

Â	CC-RH のプロパティ		2	-+
	標準ライブラリを使用する	はい		-
	標準ライブラリ関数を使用する	はい(-LIBrary=libc)		
	数学ライブラリ(倍精度)関数を使用する	はい(-LIBrary=libm)		
	数学ライブラリ(単精度)関数を使用する	はい(-LIBrary=libmf)		
	非局所分岐関数を使用する	いいえ		
⊿	よく使うオプション(へキサ出力)			
	ヘキサ・ファイルを出力する	はい		
	ヘキサ・ファイル・フォーマット	モトローラ・Sタイプ・ファイル(-FOrm=Stype)		
	出力フォルダ	%BuildModeName%		
	出力ファイル名	%ProjectName%.mot		
\triangleright	分割出力ファイル	分割出カファイル[0]		
\triangleright	レジスタ・モード			
\triangleright	エラー出力			
\triangleright	警告メッセージ			
⊿	ビルド方法			=
	→ 括ビルドを行う	1417		-
	インクルード・ファイルが存在しないソースの扱い	再コンパイル/アセンブルしない		
⊳	ハージョン選択			
⊳	記録			
⊳	その他			+
-	the last (a Metric based of the			
13	バルート・ファイルか仔在しないソースの扱い コンコーノルがインカル、ドレスハスコーノルが左キしたハゼ		ガルサスわじ	554 + 444
辺辺	-人・ノアイルルイノツルートしているノアイルル144年しんい場 」ます。	言、そのソース・ファイルを再コンハイルノアセン。	711/9 @APC:	がを選
100				
\ <u></u> #	ŧ通オブション / コンバイル・オ / アセンブル・オ	<u> </u>	/0ヘッダ・フ:	7 / 🔻

図 7-4 CC-RHのプロパティ設定

さらに、システム依存情報としてヘッダ・ファイル「usrown.h」が必要となります。このファイルはサンプル・プロジェクトに含まれています(下記フォルダ参照)

C:¥Program Files¥Renesas Electronics¥CS+¥CC¥SampleProjects¥RH850¥RH850_F1L_RI850V4RH¥ appli¥include

このファイル内では,基本クロック用タイマで使用するカウントレジスタアドレス,および,1カウント当たりの時間をマクロ定義します。詳細については「RI850V4 V2 リアルタイム・オペレーティング・システム ユーザーズマニュアル コーディング編」を参照してください。



8. 注意事項

8.1. CS+ for CC のプロジェクト作成

本製品を使用したプロジェクトを作成するには、以下の3つの方法があります。

- 本製品添付のサンプル・プロジェクトを流用する
- 新しいプロジェクトを作成する
- RI850V4 (V1) のプロジェクトを流用する

8.1.1. 本製品添付のサンプル・プロジェクトを使用する

CS+ for CC のスタートパネルの [サンプル・プロジェクトを読み込む] エリアで [RH850] タブを選択し, 「RH850_F1L_RI850V4」という名称のプロジェクトを選択して [GO] ボタンをクリックしてください。その後, プ ロジェクトを作成するフォルダ位置を指定することでサンプル・プロジェクトが読み込まれます。

〇 CS+ for CC - [スタート]	The star and the base of a set of the set of	×
ファイル(F) 編集(E) 表示(V) プロジ	ジェクト(P) ビルド(B) デバッグ(D) ツール(T) ウインドウ(W) ヘルプ(H)	
🚳 スタート(S) 🍟 👗 🖻 🖻 🕨	ッマ語典型 - 100% - 原研 - 人 原 D, h I B D h	, se ce ce se
ブロジェクト・ツリー 무 🗙	Q 29-1	- x
2 @ 2 2	1. RX610_RI600V4 73U	
	CO 2. RH850_F1L_R1850V4RH 3. sample_r178 4. reset_clock_change_freq_0	
	e' studio / CubeSuite / High-performance Embedded Workshop / PM+のプロジェクトを開く e' studioや旧開発環境で作成したプロジェクトをCS+プロジェクトへ変換することができます。 サポートするバージョン: -e' studio e' studioが出力したropoファイルを読み込むことができます。 ※異なるコンパイラを使用したプロジェクトを使うます。 ※異なるコンパイラを使用したプロジェクト間の変換時にはインクルード・パスとマウロを変換します。 サンプル・プロジェクトを読み込む すべにどルトマゴー、サンプル・プロジェクトを参数用意しています。下のリストから選択後、GOボタンを押してください。そ たサンプル・プロジェクトをごにコピーするかを決めます。	- -
ブロジェクト・ファイル(*mtpj)を開くには、ここに ドロップしてください。	GO RH850 RH RH850 FIL Tutorial Analysis RH850 FIL Tutorial Basic Operation RH850 MultiCore_EIx_Tutorial_Basic_Operation	
		₽ X
F7 スタートパ F2 名前の変更 F3 次を接	▲ 「日次を置換」「日本を置換」「日本に、「日本にはなるデバー」「日本に、「日本に、「日本」」」「日本」」「日本」」「日本」」「日本」」「日本」」」「日本」」」「日本」」「日本」」」「日本」」「日本」」」「日本」」「日本」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」「日本」」」」「日本」」」「日本」」」」」「日本」」」「日本」」」「日本」」」「日本」」」」「日本」」」「日本」」」「日本」」」」「日本」」」「日本」」」「日本」」」「日本」」」」「日本」」」「日本」」」「日本」」」「日本」」」」「日本」」」「日本」」」「日本」」」」」」」「日本」」」「日本」」」」「日本」」」」「日本」」」「日本」」」」」「日本」」」」」」」「日本」」」「日本」」」「日本」」」」」「日本」」」」」」」」	イン「肥関数ヘジャ」
		<mark>▲</mark> 非接続

図 8-1 サンプル・プロジェクトを使用する場合

8.1.2. 新しいプロジェクトを流用する

CS+ for CC のスタートパネルの [新しいプロジェクトを作成する] エリアの [GO] ボタンを押し [プロジェクト 作成] ダイアログをオープンします。

図 8-2 プロジェクト作成ダイアログ(プロジェクト新規作成)

プロジェクト作成							
マイクロコントローラ(<u>T</u>):	RH850						
使用するマイクロコントローラ(<u>M</u>)	e						
(マイクロコントローラを検索)	できます) アップデート(山)…						
 ■ RH850/F1L ■ R7F701010×AFP(48pin) ■ R7F701013×AFP(64pin) ■ R7F701013×AFP(64pin) ■ R7F70103×AFP(80pin) ■ R7F70103×AFP(100pin) ■ R7F70103×AFP(144pin) ■ R7F70103×AFP(144pin) ■ R7F70103×AFP(14pin) ■ R7F70103×AFP(176pin) ■ R1850/F1L(WS1) 							
プロジェクトの種類(<u>K</u>):	アプリケーション(RI850V4,CC-RH)						
プロジェクト名(<u>N</u>):	sample_appli						
作成場所(<u>L</u>):	D¥Work						
	フロジェクト名のフォルダを作成する(A)						
D:¥Work¥sample_appli.mtpj							
📄 既存のプロジェクトのファイルね	構成を流用する(<u>S</u>)						
流用元のプロジェクト(P): (流用元のプロジェクト・ファイルを入力してください) 💌 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(<u>O</u>)						
	(作成(<u>C)</u> キャンセル ヘルブ(<u>H</u>)						

- [マイクロコントローラ] : 「RH850」を選択してください
- [プロジェクトの種類]:「アプリケーション(RI850V4, CC-RH)」を選択してください。

[作成] ボタンを押すと、プロジェクトが作成されます。

8.1.3. RI850V4 のプロジェクトを流用する

CubeSuite+や CS+ for CACX で作成した RI850V4(V1)を使用したアプリケーションを流用することができます。

CS+ for CC のスタートパネルの [新しいプロジェクトを作成する] エリアの [GO] ボタンを押し [プロジェクト 作成] ダイアログをオープンします。

プロジェクト作成							
マイクロコントローラ(工):	RH850						
使用するマイクロコントローラ(M);						
	できます) アップデート(山)						
R7F701035xAFP(R4850/F1L(WS1) R4850/P1M R4850/P1M R4850/R1L	176pin) 品種名:R7F701035xAFP 内蔵ROMサイズIVバイト: Code Flash:2048 Data Flash:204 Data Flash:20 内蔵RAMサイズUバート: Local RAM(Primary):131072 Local RAM(Secondary):32768 Retention RAM:32768 Trace RAM:329728						
プロジェクトの種業類(<u>K</u>):	アプリケーション(RI850V4,CC-RH)						
プロジェクト名(<u>N</u>):	sample_appli						
作成場所(<u>L</u>):	D:¥Work						
	プロジェクト名のフォルダを作成する(A)						
D:¥Work¥sample_appli.mtpj							
🔽 既存のプロジェクトのファイル	構成を流用する(ら)						
流用元のプロジェクト(<u>P</u>):	流用元のブロジェクト(P): D:¥Work¥V850ES_JG3_R1850V4¥V850ES_JG3_ IPI 参照(W)						
🔽 ブロジェクト・フォルダ以下の	構成ファイルをコピーして流用する(0)						
「作時気(<u>C</u>) キャンセル ヘルプ(<u>H</u>)							

図 8-3 プロジェクト作成ダイアログ (プロジェクト流用)

- [マイクロコントローラ] : 「RH850」を選択してください
- [プロジェクトの種類]:「アプリケーション(RI850V4, CC-RH)」を選択してください。
- [既存のプロジェクトのファイル構成を流用する]を選択し、流用元プロジェクトを選択します(流用元プロジェクトのプロジェクトファイルを選択します)。
- 作成するプロジェクトフォルダ以下に、ファイルをコピーして流用する場合は[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]を選択します。

[作成] ボタンを押すと、プロジェクトが作成されます。

8.1.4. マルチコア用の RI850V4 プロジェクトを作成する

CS+で RH850 マルチコア用 RI850V4 プロジェクトを構築する手順について説明します。CS+でマルチコア用プロジェクトを作成する場合、コア毎にサブ・プロジェクトを構築します。

1. マルチコア用プロジェクトの新規作成

CS+ for CC のスタートパネルの [新しいマルチコア用プロジェクトを作成する] エリアの [GO] ボタンを押し [プ ロジェクト作成] ダイアログをオープンします。

プロジェクト作成		x						
マイクロコントローラ(<u>T</u>):	RH850							
使用するマイクロコントローラ(<u>M</u>):							
	索できます) アップデート(<u>U</u>)…							
R7F701513(BG/ R7F701512(BG/ R7F701512(BG/ R7F701508(BG/ R7F701507(BG/ R7F701507(BG/ R7F701508(CG R7F701508(CG R7F701502(CG	A272pin) A272pin) A272pin) 日積名のMサイズ[[バパイト]: Code Flash6144 Data Flash64 内蔵RAMサイズ[[バイト]: Local RAM(CPU1):196608 Local RAM(CPU2):196608 Local RAM(CPU2):196608 Local RAM(CPU3):196608 Local	E						
プロジェクトの種類(<u>K</u>):	マルチコア用ブート・ローダ(CC-RH)							
プロジェクト名(<u>N</u>):	sample							
作成場所(上):	D:work	▼ 参照(<u>R</u>)						
	✓ プロジェクト名のフォルダを作成する(A)							
D:work¥sample¥sample.mtp	j							
 □ 各コア専用のmain関数を含むスタートアップ・ファイルの雑形を同時に作成する(E) ◎ コア数分のアプリケーション・プロジェクトとして作成(S) ◎ 1つのプロジェクトとして作成(D) 								
	作成(<u>C</u>) キャンセル	~)1/J(H)						

図 8-4 プロジェクト作成ダイアログ(マルチコア用プロジェクトの新規作成)

- [マイクロコントローラ] : 「RH850」を選択してください
- [プロジェクトの種類]:「マルチコア用ブートローダ(CC-RH)」を選択してください。
- [各コア専用の main 関数を含むスタートアップ・ファイルの雛形を同時に作成する]のチェックを外します。

[作成] ボタンを押すと、プロジェクトが作成されます。

2. サブ・プロジェクトの追加

プロジェクトの「プロジェクト・ツリー」パネルを開きます。「プロジェクト・ツリー」パネル内にあるプロジェ クトフォルダにカーソルを合わせて右クリックし、「追加」→「新しいサブ・プロジェクトを追加」を選択し、サ ブ・プロジェクト作成ダイアログをオープンします。



サブプロジェクト作成	5	A. 3.	3.3		x
マイクロコントローラ(工):	RH850				-
使用するマイクロコントローラ(M):				
篇(マイクロコントローラを検索	できます)	「アップデート(<u>U</u>)…」			
R7F701503(LOFP R7F701502(LOFP R7F701501(LOFP R850/F1H-GW RH850/F1H-GW RH850/F1H RH850/F1L RH850/E1L RH850/E1L RH850/E1M-S	176pin) 176pin) 176pin)	品種名/R7F01503 内蔵ROMサイズ[K)パイ Code Flash:6144 Data Flash:64 内蔵RAM(イズ[V]イト Local RAM(GPU]):19 Local RAM(GPU]):19 Local RAM(55536 Retention RAM:1310 注動酒種報:Number of (ト]:]:]6608]6608 72 Cores: 2	E	
プロジェクトの種類(<u>K</u>):	アプリケーション(RJ	850V4,CC-RH)		ŀ	-
プロジェクト名(<u>N</u>):	sub_appli				
作成場所(<u>L</u>):	D:¥work¥sample_	appli		▼ 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>)			
D:¥work¥sample_appli¥sub_ap	opli¥sub_appli.mtsp				
📄 既存のプロジェクトのファイル	構成を流用する(<u>S</u>)				
流用元のプロジェクト(<u>P</u>):	(流用元のプロジ:	クト・ファイルを入力してく	だざい)	- 参照(₩)	
□ 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>0</u>)			
	(Ť	成(<u>C)</u> キャン	セル	へルプ(<u>H</u>)	

図 8-5 プロジェクト作成ダイアログ (サブ・プロジェクトの追加)

- [マイクロコントローラ] : 「RH850」を選択してください
- [プロジェクトの種類]:「アプリケーション(RI850V4, CC-RH)」を選択してください。

サブ・プロジェクトは、既存の RI850V4 プロジェクトを流用して作ることもできます。流用してプロジェクトを 作成する方法は「8.1.3 RI850V4 のプロジェクトを流用する」の手順を参考にしてください。

3. システム・コンフィギュレーション・ファイルの登録

プロジェクトの「プロジェクト・ツリー」パネルを開きます。「プロジェクト・ツリー」パネル内にあるサブプロ ジェクト・フォルダにカーソルを合わせて右クリックし、「追加」→「新しいファイルを追加」を選択し、「ファ イル追加」ダイアログをオープンします。



凶 0-0 ノアイル迫加アイアロン	义	8-6	ファイル追加ダイアロ・	グ
-------------------	---	-----	-------------	---

ファイル追加	X					
ファイルの種類(<u>T</u>): Cソース・ファイル (*.c) ヘッダ・ファイル (*.h; *.inc) アセンブリ・ソース・ファイル (*.asm; *.s; *.fsy) Pythonスクリプト・ファイル(*.py) システム・コンフィギュレーション・ファイル(*.cfg) テキスト・ファイル(*.txt)						
空のシステム・コン	フィギュレーション・ファイルです。					
ファイル名(<u>N</u>):	cfg					
作成場所(」):	D:¥work¥sample_appli¥sub_appli 参照_(<u>R</u>)					
	OK キャンセル ヘルプ(H)					

[ファイルの種類]:「システム・コンフィギュレーション・ファイル(*.cfg)」を選択してください
 [OK]ボタンを押すと、システム・コンフィギュレーション・ファイルが作成されます。

4. サブ・プロジェクトの対象となる PE 番号を設定

プロジェクトの「プロジェクト・ツリー」パネルを開きます。「プロジェクト・ツリー」パネル内に登録したサ ブ・プロジェクトのシステム・コンフィギュレーション・ファイルにカーソルを合わせて右クリックし、システ ム・コンフィギュレーション・ファイルの「プロパティ」ダイアログをオープンします。



プロパティ		×					
od sample.cfg のプロパティ		P -+					
	(はい(cfgファイル変更時に更新する)(-i) %BuildModeName% sits (はい(cfgファイル変更時に更新する)(-d) %BuildModeName% kernel_idh (はい(cfgファイル変更時に更新する)(-e) %BuildModeName% entrys 直接ペクタ方式 いいえ -peid=2						

図 8-7 システム・コンフィギュレーション・ファイル関連情報

● [ユーザ指定オプション] :-peid=<*id*>を指定してください。*id* は本プロジェクトの対象となる PE 番号を指 定してください。

複数の PE で RI850V4 を利用する場合は、2.~4.の手順を再度繰り返してください。

以下に、マルチコア用プロジェクトを生成する際の注意事項を示します。

- (ア) 1. の手順によってマルチコア用ブートローダと合わせてマルチコア用のブート処理 (boot.asm)が生成され ます。この自動的に生成される boot.asm には、割り込みベクタの定義が含まれています。一方で、CF850V4 が出力する情報ファイルにも割り込みベクタの定義が含まれています。CF850V4 に設定したオプションに よっては、両者の出力が重複する場合があります。具体的には、システム・コンフィギュレーション・ファ イルの「プロパティ」ダイアログを開き、「エントリ・ファイル」内にある設定項目「出力方式」として「直 接ベクタ方式」を選択した場合、「例外ハンドラ・ベクタ・アドレスを指定する」として「いいえ」を選択 すると RI850V4 用コンフィギュレータが出力する割り込みエントリ処理と boot.asm 処理内で定義している 割り込みベクタが重複します。この場合、boot.asm から重複している割り込みベクタの定義を削除してくだ さい。重複したまま使用した場合は、RI850V4 の割り込み管理機能が正常に動作しません。また、boot 処理 で行うべき処理については「RI850V4 V2 ユーザーズマニュアル コーディング編」を参照してください。
- (イ) 1. の手順によって「各コア専用の main 関数を含むスタートアップ・ファイルの雛形を同時に作成する」を 選択した場合に生成されるプロジェクトは、RI850V4 を使用しないプロジェクトです。従って、RI850V4 を 使用する場合はチェックを入れないようにしてください。チェックを入れてサブ・プロジェクトの生成を行 った場合は、一旦サブ・プロジェクトを削除し、再度 2. 以降の手順を実施して RI850V4 用のサブ・プロジ ェクトの登録を行ってください。

8.2. プラグインの有効化

本製品のインストール直後は、本製品のプラグインが CS+ for CC に読み込まれず、無効になっている場合がありま す。本製品のプラグインが無効になっていると、ビルドできないなどの問題が生じます。

CS+ for CC の [プラグインの管理] ダイアログの [追加機能] タブで、以下のプラグインを有効にしてください。

- リアルタイム OS ビルド設定プラグイン(共通部)
- リアルタイム OS 解析制御プラグイン(共通部)
- リアルタイム OS リソース情報表示プラグイン(共通部)
- リアルタイム OS タスク・アナライザ・プラグイン(共通部)

ックインの管理 ibeSuite+起動時に読み込むプラグインにチェックしてください。 の設定は次回起動時に有効となります。 CubeSuite+の動作に必須のプラグインはグレー表示となってお イクロコントローラ用プラグインのチェックは、外さないことを推奨し	5り、チェックを外すことはできません。また、基本機能タブにおいて、開発対象とな します。
基本機能 追加機能	
モジュール名	説明
 許 IronPythonコンソール・プラグイン 許 アップデート・マネージャ・プラグイン ゴ エディタ・パネル ゴ ニド生成プラグイン 計 コード生成プラグイン2 許 スタック見積もりツール デ ツール間インターフェース(TIP)プラグイン 許 デバッグ・コンソール・プラグイン デ プログラム解析プラグイン 	IronPythonのコマンドとCubeSuite+拡張機能が使用できるコンソールです。 CubeSuite+ アップデート・マネージャと連携するプラグインです。 エディタ・パネルのプラグインです。 デバイスドライバを自動生成するプラグインです。 デバイスドライバを自動生成するプラグインです。 スタック使用量をツリー形式で表示するツールです。 TIPを使用したツール間インターフェイス用プラグインです。 標準1/Oをサポートするデバッグ・コンソール・プラグインです。 プログラムの解析を行うプラグインです。
 図 いレックイムOSタスク・アナライザ・ブラグイン(共通部) 図 いレックイムOSビルド設定ブラグイン(共通部) 図 いレックイムOSリソース情報表示ブラグイン(共通部) 図 いレックイムOS解析制御プラグイン(共通部) 図 いレックイムOS解析制御プラグイン(共通部) 図 就子配置プラグイン 	リアルタイムOSが組み込まれたプログラムの解析を行うプラグインです。 リアルタイムOSのビルド情報を設定するプラグインです。 リアルタイムOSの資源情報を表示するプラグインです。 リアルタイムOS情報の解析と管理をするプラグインです。 デバイスの端子配置を行うプラグインです。

図 8-8 プラグイン管理



8.3. コンフィギュレーション・ファイルのプロパティ

コンフィギュレーション・ファイルのプロパティにて、コンフィギュレータに指定するオプションを指定します。

コンフィギュレーション・ファイルのプロパティは [プロジェクト・ツリー] 内にあるコンフィギュレーション・フ ァイルを右クリックし, [プロパティ]を選択することでオープンします。

RH850_F1L_RI850V4RH - CS+ for CC			-				X
ファイル(F) 編集(E) 表示(V) プロジェクト(P) ビルド(B) デバッグ(D) ツール(「) ウインドウ(W	/) ヘルプ(H)				
🚳 スタート(S) 🔒 🗃 🍯 🔏 🐚 🚳	る職業家	- 100%	🔹 😽 😽 DefaultE	uild 🗾 🖈	- 🐻 🗅 🗂 I 🖲) () () () () () () () () () () () () ()	
プロジェクト・ツリー	21 プロパティ						- I - I
2 @ 2 2	ed sys.cfg のプロパティ						P
<u>RH850 F1L RI850V4RH (プロジェクト)</u>	▲ システム情報テーブル・ファイル						
■ R7F701007xAFP (マイクロコントローラ)	ファイルを生成する 出力フォルダ			しましい(ctgファイル)が %BuildModeNam	と更時(と更新する)(−1) e%		
や RI850V4 (リアルタイムOS)	ファイル名 4 システム情報へッダ・ファイル			sits			
🔊 RH850 シミュレータ (デバッグ・ツール)	ファイルを生成する			はい(cfgファイル達	で更時(こ更新する)(-d)		
ファイル	コアイル名			%BuildModeName kernel_idh	3%		
boot.s	エントリ・ファイル ファイルを生成する			はい(cfgファイル落	で更時(ご更新する)(-e)		
- cychdr.c	出力フォルダ			%BuildModeName	e%		
	リアイル名出力方式			entrys 直接ベクタ方式			
	例外ハンドラ・ベクタ・アドレスを指定す コンフィギュレータ記動設定	3		いいえ			
usr stkovr.s	ユーザ指定オブション			-peid=1			
🔤 🔤 sys.cfg							
Ⅲ····ジリアルタイムOS関連ファイル							
	ユーザ指定オブション 任音のオブションを指定する際に、オブション	, 立字列发直接职行	tl.+7				
	11.25 (0/1 / 2 32) C18/2 9 (D/mile, 1) / 23.		206.98				
	システム・エンフィギュレーション・ファイル	関連情報、ファ	イル情報/				
	出力						4 ×
	[EOF]						*
							E
							-
	すべてのメッセージ						-
۰	🔜 出力 🎦 エラー一覧						
F1 F2 F3	FH F5	FE	F7	FB	F9	FID F11	FI2
							<mark>入</mark> 非接続

図 8-9 コンフィギュレーション・ファイルのプロパティ

[システム情報テーブル・ファイル] と [システム情報ヘッダ・ファイル] は RI850V4 (V1) と設定内容,設定項目 は同じです。

[エントリ・ファイル] は

- [ファイルを生成する]
- [出力フォルダ]
- [ファイル名]

までは RI850V4 (V1) と設定内容,設定項目は同じですが, RI850V4 V2 では

- [出力方式]
- [割り込みハンドラ・アドレス・テーブルのベース・アドレス]
- [例外ハンドラ・ベクタ・アドレスを指定する]
- [例外ハンドラ・ベクタ・アドレス]

の設定が新たに追加されています。

[出力方式] として

- [直接ベクタ方式]
- [テーブル参照方式]

のどちらかを選択でき、[テーブル参照方式]を選択した場合は[割り込みハンドラ・アドレス・テーブルのベース・ アドレス]を指定します。

[例外ハンドラ・ベクタ・アドレスを指定する] で

- [はい]
- [いいえ]

のどちらかを選択でき、 [はい]を選択した場合は [例外ハンドラ・ベクタ・アドレス]を指定します。

[コンフィギュレータ起動設定]は、RI850V4 V2 で新規に追加された項目です。

[コンフィギュレータ起動設定] で

[ユーザ指定オプション]

を指定でき,-peid=<*id*>オプションを指定することで, RI850V4 を組み込む対象となる PE を選択することができます。



8.4. シミュレータ使用時の注意事項

8.4.1. シミュレータ設定ファイルについて

CS+ for CC のシミュレータを使用してアプリケーションを動作させる場合,シミュレータの設定を行うための 「rh850_simulator.setini」ファイルが必要になります。このファイルを CS+ for CC のプロジェクト・ファイル (mtpj ファイル)と同じ階層に配置してください。サンプル・プロジェクトを使用している場合は自動的に配置されますが, 新規にプロジェクトを作成した場合は,下記から該当ファイルをコピーしてご使用ください。

- Windows®が 32bit 版で、システムドライブが C:の場合 C:¥Program Files¥Renesas Electronics¥CS+¥CC¥SampleProjects¥RH850¥RH850_F1L_RI850V4RH
- Windows®が 64bit 版で, システムドライブが C:の場合 C:¥Program Files (x86)¥Renesas Electronics¥CS+¥CC¥ SampleProjects¥RH850¥RH850_F1L_RI850V4RH

8.4.2. シミュレータを使用した場合の OS タイマについて

CS+ for CC のシミュレータを使用した場合、使用可能な OS タイマに制限がある場合があります。詳細はシミュレータの制限事項を参照してください

8.5. リアルタイム OS リソース情報表示パネルに関する注意事項

8.5.1. 参照はリアルタイム OS 初期化後に行う

リアルタイム OS リソース情報パネルを参照する場合は、リアルタイム OS 初期化後に参照してください。リアル タイム OS の初期化完了前は、リアルタイム OS リソース情報パネルの表示が不定となります。

8.5.2. デバッグ情報を生成したプログラムを使用する

リアルタイム OS リソース情報パネルを使用する際は、デバッグ情報を生成したプログラムをダウンロードしてください。デバッグ情報がないプログラムをダウンロードして、リアルタイム OS リソース情報パネルを表示しようとした場合、エラーが発生します。

デバッグ情報を生成するには「ビルド・ツール」の「リンク・オプション」のプロパティで「デバッグ情報を出力 する」を「はい」に設定してください。 8.6. リアルタイム OS タスク・アナライザに関する注意事項

8.6.1. トレース・モード変更

RI850V4 のプロパティの「タスク・アナライザ」タブで、「トレース・モードの選択」を選択します。「トレース・モードの選択」のデフォルト値は「トレースしない」です。リアルタイム OS タスク・アナライザを使用する場合は「トレースしない」以外 を選択してください。また「トレース・モードの選択」を変更した場合は、必ずビルドを行ってください。トレース・モードごとに使用するモニタが違うため、ビルドを行うことで正しいモニタを組み込みます。

O RH850_F1L_RI850V4RH - CS+ for CC - [7	「ロパティ]	Carden &					
ファイル(F) 編集(E) 表示(V) プロジェクト(F	?) ビルド(B) デバッ?	グ(D) ツール(T) ウインドウ((W) ヘルプ(H)		
🚳 スタート(S) 🔒 🍟 🎖 🖻 🖻 🔘	0 0 8 4 4	Ç 6				۱) () († () () () () () () () () () () () () ()
·							
ブロジェクト・ツリー	🚰 プロパティ 📑 task)	c					- x
2 🕜 🙎 🔳	₽ RI850V4 のプロパテ	í.					P - +
RH850 F1L RI850V4RH (プロジェクト) 電 R7F701007xAFP (マイクロコントローラ)	トレース トレース・モードの選手	択	٢	レースしない			
く CC-RH (ビルド・ツール) そ RI850V4 (リアルタイムOS)							
 RH850 シミュレータ (デバッグ・ツール) ファイル boot.s task.c cychdr.c inirtn.c idirtn.c inithdr.c usr_stkovr.s vs.sfg wy.stkovr.s wy.stkovr.s 	トレース・モードの選択 トレースを信頼するバシフ ハードウエア・トレース・モ ソフトウエア・トレース・モ	マと、トレースで取り ード:トレースをデリ ード:トレースをター	<mark>手したい)情報服を没 ッグ・ツールの持ち デットのRAM へ1:</mark>	試択します。 つトレース・メモビ 呆持します。この	リルに保持します。 リ項目を選択し	。 た場合,トレ	
kernel_id.h ● entry.s ● 雪 sit.s ● 切 リアルタイムOS関連ファイル 「「」 trcinf.c	(RB5004 入 3次). 出力 [EOF]						+ ×
< III •	すべてのメッセージ/ → 出力 <u>■</u> エラー一覧						+
F1 F2 F3 F4	FS F6	F7	FB	F9	FHB	F11	FH2
e vio vio vio vio						3	<mark>~</mark> 非接続

図 8-10 トレース・モードの選択



シミュレータを使用して「ハードウェア・トレース・モードで、トレース・チャートを取得」する場合 RH850 のシミュレータを使用してアプリケーションを動作させる場合、リアルタイム OS タスク・アナライザで表 示する時間を正しくするために、次の設定値を適切に設定する必要があります

- メイン・クロック周波数 [MHz]
 RH850 シミュレータ(デバッグ・ツール)プロパティの「接続用設定」タブ内の「クロック」カテゴリ(図 8-11 参照)
- トレース・タイム・タグの分周率
 RH850 シミュレータ(デバッグ・ツール)プロパティの「デバッグ・ツール」タブ内の「トレース」カテゴリ
 (図 8-12 参照)

図 8-11 RH850 シミュレータ(デバッグ・ツール)プロパティの「接続用設定」タブ

O RH850_F1L_RI850V4RH - CS+ for CC - [7	プロパティ]							. D X
ファイル(F) 編集(E) 表示(V) プロジェクト(F) ビルド(B)	デバッグ(D)	ツール(T)	ウインドウ(\	N) ヘルプ(H))		
🏟 スタート(S) 📮 🍟 🎖 🖻 🛍 🖻	@ #4 ₽	A.,	÷ 🖓	T	=	👧 🗅 🐂) H1 🚳 🏅
: _ 무 목 및 주 : 운 : 삶 삶								
プロジェクト・ツリー 中 🗙	🚰 ว่อパริง	🗹 task.c						- x
2 3 2 2	🚑 RH850 핫ミ	ュレータ のプロ/	(न्र					+ – ۹
<u>RH850 F1L RI850V4RH (プロジェクト)*</u>	<u>⊿ 50%5</u>							
- 尾 R7F701007xAFP (マイクロコントローラ)	メイン・クロッ	り周波数[MHz]	80	.00			-
CC-RH (ビルド・ツール)	タイマノトレ タイマノトレ	- ス用クロックル - ス用クロックル	山灰奴の進か 記波数の単位	MH	ロジロッジルIIの東知 セ			
🖗 RI850V4 (リアルタイムOS)	タイマ/トレ	ース用クロック履	訂波数					
🔺 RH850 シミュレータ (デバッグ・ツール)	▲ コンフィギュ シミュレータ	レーション ニアンコンペーレー	ວ່າວ ແລະ ແມ່ນ	に使用する ハイ	17			
□ ファイル	2410.3	1221411	JEJ JRIN	20011920 00	<i>v</i> .			
boot.s								
task.c								
idlrtn.c								
inthdr.c	メイン・クロック周	波鼓[MHz]					122400241-1-1-1	a/# a IB 4/4
usr_stkovr.s	 クロックのジミュレ 直接キーボード 	~トの為に、使 吟入力して(想	用マイクロコンI さい。	・ローラのメイン・	・クロックの周辺変	にを入力します。	遮积胶以外	の他の場合は
□ □ ··································	接続用設定	デバッグ・ツ	ール設定 🗸 !	ダウンロード・フ	777 Julian / フラ	ッシュ・オプショ	ン / フック	処理設定 / ₹
kernel_id.h	出力							д X
entry.s	[EOF]							*
sit.s								
□								=
								-
	すべてのメッヤ	-9/						-
• III • •	🔜 出力 🛅 エ:	-一覧						
F1 F2 F3 F4	FS	FG	F7	FB	F9	FHD	FTI	FI2
							_ <mark>↓</mark> 非	接続



() RH850_F1L_RI850V4RH - CS+ for CC - [7	「ロパティ]							x
ファイル(F) 編集(E) 表示(V) プロジェクト(F	?) ビルド(B)	デバッグ(D) ツール	(T) ウインドウ(W) ヘルプ(H)				
🚳 スタート(S) 🛄 🍟 🐰 🖻 💼 🖷	? (~) 品 🖗	A			" 🖏 🗅 I	ה 🗈 🖟	• • • • • • •	** -
プロジェクト・ツリー	🚰 プロパティ	🗹 task.c						- x
2 7 2 2	<u>(</u> RH850 シミ	ゴ ュレータ のプロパティ					٩	- +
<u>RH850 F1L RI850V4RH (プロジェクト)*</u>	▲ メ モ リ							
R7F701007xAFP (マイクロコントローラ)	▷ メモリ・マッピン またけのよ	ノヴ エリーマ わわつ		[27]				
▲ CC-RH (ビルド・ツール)	* 天口中の入 実行中に表	モリ・アクビス 示更新を行う		(ttt)				
冬 RI850V4 (リアルタイムOS)	表示更新間	那高[ms]		500				
🔐 RH850 シミュレータ (デバッグ・ツール)	▲ トレース トレーフ 部分	とち /市田オス		14/2				
シリファイル	ドレース1級能 実行前にト	こを1920日9つ レース・メモリをクリアする		はい				
boot.s	トレース・メモ	りを使い切った後の動作	:	トレース・メモリを上	書きし実行を続	ける		
task.c	トレース・タイ	(ム・タグを積算する」		いいえ				
	<u> </u>	リーックスレレース」 (ム・タグの分周率		1/2				
	P 21Y							
	▷ カバレッジ ▲ シミュレータ(3111						
	シミュレータの	JUIを表示する		いいえ				
usr_stkovr.s	▲ ステップ実行	ŕ						
	指定での73	ンを入ギップする		いいえ				-
□ □···································	指定セクションを	スキップする						
kernel_id.h	指定したセクンヨ	ンをステッフ・イン実行中	にスキッフするかとつ	の指定します。				
entry.s								
	接続用設定	↓ テバッグ・ツール設定	ダウンロード・ご	ファイル設定 / フ:	ラッシュ・オプショ	ン設定 <u>人</u> こ	フック処理設定	/ -
□	出力							чх
	[EOF]							•
								=
								-
	すべてのメッセ	->						-
< III +	🔜 出力 🛅 工	5-一覧						
F1 F2 F3 F4	F5	F6 F7	FB	F9	FHD	FTI	FIE	
							<mark>▲</mark> 非接続	

図 8-12 RH850 シミュレータ(デバッグ・ツール)プロパティの「デバッグ・ツール」タブ



そして, デバッガのトレーススイッチを ON にしてください(足跡マークのアイコン)。

	to CS, for CC. [traje a]	
		1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
ブロジェクト・ツリー 平 🗙	● 逆アセンブル1 ≦ boots ≦ taskc	ローカル変数 4 ×
2 🕜 🙎 🗷		図 表記(N)▼ 100 エンコード(C)▼
RH850 F1L RI850V4RH (プロジェクト)*		カレント 💌
■ R7F701007xAFP (マイクロコントローラ)	115 000061ba ext_tsk();	名前(
- CC-RH (ビルト・ツール) - 2 RT850V4 (リアルタイムOS)	117	
▲ RH850 シミュレータ (デバッグ・ツール)	118 void 119 000061c2 task5(VP_INT exinf)	🗊 😜 strings
ファイル	120 ⊟{ 121 SMP MSG task as a:	
▶ 🗊 ビルド・ツール生成ファイル	122 000061c8 char strings[] = {"TASK5"};	
boot.s	123 124 000061e6 err = wai_sem(ID_SEM1);	
Cuchdr c	125 000061f4 if (err != E_0K) { 126 000061f8 while(1) {}	
inirtn.c	127 199 00000160	
- idirtn.c		
	リアルタイムOS リソース情報 キ ×	
usr_stkovr.s	- 弐.固定長… 😤 可変長… 😌 周期八… 🧃 割り込… 🦽 初期化… 🧬 拡張サ… 🧐 レディ… 🧐 タイマ・…	
kernel id.h	RTOS名 RI850V4	
entry.s		
- em sit.s		
□ □ ··································	戦和(市に更新)	
End treinf.c	解析結果 ロ ×	
	ぼ7-9 名称 マロ PE番 280.75 2870.76	
	▶ 🛃 0x104c	③ ローカル変数 「第 CPUレジスタ」
	[] [] [] [] [] [] [] [] [] [] [] [] []	出力
		Renesas Uptimizing Linker Completed」 ビルド終了(エラー:0個, 警告:0
		d),
	1 ID_TASK4	************************************
	D_TASK5	
	8250.00	nts = 0, Miss Counts = 0
×	0	↓ t / *ビ / *デ / ▼ / ▼
F1 F2 F3	F4 F5 F6 F7 F8 F9 F8	FT FP
	122 行 1 桁 挿入 日本語 (シフト JIS) BREAK 🚱 0x000061c8 🚥 RH8	50 シミュレータ 🍈 未計湯 🛐 🚺 🖽

図 8-13 トレーススイッチ

8.6.2. デバッグ・ツールの設定

デバッグ・ツールのプロパティにおいて「デバッグ・ツール」タブ内の「トレース」カテゴリを以下の組み合わせの 設定にしないでください。

- 実行前にトレース・メモリをクリアする:いいえ
- トレース・タイム・タグを積算する:はい

8.6.3. トレースのタイムスタンプについて

トレースのタイムスタンプは、カーネルのタイマ機能を使用して実現しています。カーネルのタイマは OS タイマ割 り込みを使用して実現しているため、割り込み禁止状態の場合は、タイマ割り込み処理が保留されます。タスクなどで 割り込み禁止にし、その期間が 1ms 以上であった場合は正しい時間を表示できません(処理順は正しく表示します)。 8.7. カーネル・ソース・コードのビルド方法

RI850V4 カーネルはライブラリで提供しているため,通常はカーネル・ソース・コードをビルドしてカーネル・ラ イブラリを再生成する必要はありません。

購入された製品が「RTRRH8500TR01SRRUU(量産契約,量産数は無制限,ソース・コード付き)」の場合,カー ネルのソース・コードが「<インストール・フォルダ>¥source¥kernel」に格納されます。また,カーネルをビルドす るために必要な makefile が「<インストール・フォルダ>¥source¥project¥rh850_ccrh¥r32」に格納されています。こ の makefile は Cygwin 環境の make コマンドに対応していますので,カーネルを再生成するためには, Cygwin がイン ストールされた環境が必要となります。

● 32 ビット版 Windows の場合

Cygwin ウインドウ上で上記 makefile が格納されているパスへ移動し,

make [return]

と入力してください。これにより下記フォルダにビルドされたカーネルが格納されます。

<インストール・フォルダ>¥library¥rh850_ccrh¥r32

なお、インストール・フォルダに対する書き込み権限がない場合、インストール・フォルダを書き込み可能なフォル ダにコピーしてビルドしてください。ビルド後、インストール・フォルダに対する書き込み権限のあるユーザにて、生 成されたライブラリを「<インストール・フォルダ>¥library¥rh850_ccrh¥r32」にコピーしてください。

8.8. 可変リセットベクタ機能を利用する場合の注意事項

RH850 ではフラッシュプログラマの可変リセットベクタ機能を利用して, RBASE の値を変更することが可能です (可変リセットベクタ機能の詳細については, ハードウェアのマニュアルを参照してください)。ただし, RI850V4 は デバイス・ファイルから読み込んだ固定値のみを RBASE の値として扱います。このため, 可変リセットベクタ機能を 利用してリセットベクタの値を変更する場合, RBASE 相対のエントリ・ファイルを用いて RI850V4 の割り込み管理機 能を利用することはできません。可変リセットベクタ機能を利用する場合は, 「8.3 コンフィギュレーション・ファイ ルのプロパティ」に示す変更方法に従ってコンフィギュレータが出力するエントリ・ファイルを EBASE 相対, もしく は INTBP 相対に設定する方法を推奨します。

可変リセットベクタ機能を利用し、かつ RI850V4 の割り込み管理機能も RBASE を利用する必要がある場合は、下 記の「8.8.1 可変リセットベクタ機能を利用し、かつ RBASE 相対の割り込み管理を行う方法」に示す方法に従ってエ ントリ・ファイルの書き換えを行ってください。

8.8.1. 可変リセットベクタ機能を利用し, かつ RBASE 相対の割り込み管理を行う方法に ついて

上記の通り、コンフィギュレータは可変リセットベクタ機能によって変更した後の RBASE を基準にしてエント リ・ファイルを出力することはできません。ユーザが手動で変更後の RBASE に書き換える必要があります。

コンフィギュレータが出力するエントリ・ファイルの形式は以下のとおりです。base_addressの部分を変更後の RBASE 値に書き換えて下さい。

.org base_address + offset	分岐先アドレスの設定
jr32kernel_int_entry	割り込み処理への分岐

例:可変リセットベクタ機能によって RBASE の値を 0x00000000 から 0x00010000 に書き換えた場合,優先度 4 に対応する割り込みベクタを下記のように書き換えます。

● エントリ・ファイルの変更前

; 0x0140, call "priority 4"

.org 0x0000000 + 0x140

jr32__kernel_int_entry

● エントリ・ファイルの変更後





エントリ・ファイルを書き換える場合、コンフィギュレータを再実行した際にエントリ・ファイルが上書きされな いように図 8-14のようにオプションの設定変更を行う必要があります。

RH850_F1L_RI850V4RH - CS+ for CC - [出	动]					
ファイル(F) 編集(E) 表示(V) プロジェクト(F	P) ビルド(B) デバッグ(D) ツール(T)	ウインドウ(W) ヘルプ(H)			
🍇 スタート(S) 🔒 🔒 🐉 🐚 🖄	らら 関 🛊 🚽	• 100% • 😽 🗑	DefaultBuild	- 🔬 🐻 🛛	יין ₪ ₪ ₪ №1 @) sa ça ca K
·						
プロジェクト・ツリー 🛛 🕂 🗙	20パティ 🖌 boots 📝 task.c					- x
2 3 2 2	en overste Diffet					
RH850 F1L RI850V4RH (プロジェクト)*	▲ システム情報テーブル・ファイル					
■ R7F701007xAFP (マイクロコントローラ)	ファイルを生成する		はい(cfgファイ)	↓変更時に更新する)(−i)	
▲ CC-RH (ビルド・ツール)	出力フォルダ		%BuildModeNa	ame%		
≈ RI850V4 (リアルタイムOS)	▲ システム情報ヘッダ・ファイル		511.5			
🔊 RH850 シミュレータ (デバッグ・ツール)	ファイルを生成する		はい(cfeファイノ	レ変更時(こ更新する)(-d)	
🗊 ファイル	出力フォルターファイルター		%BuildModeNa kernel idh	ame%		
□- 🛺 ビルド・ツール生成ファイル	▲ Iントリ・ファイル		Kernerjah			
boot.s	- ファイルを生成する		はい(.cfgファ	「ル変更時に更新しな	()(-ne)	
	出力がから		%BuildModeNa entry s	ame%		
-9 cychdr.c	出力方式		直接ベクタ方式	¢		
- 🛀 inirtn.c	例外ハンドラ・ベクタ・アドレスを指定する		いいえ			
- 🛀 idlrtn.c	4 コノノイキュレータ延期設定 フーザ指定オブション		-peid=1			
isr_stkovr.s						
-2 sys.cfg						
□- ³ リアルタイムOS生成ファイル						
entry.s						
and sit.s						
□ · 切 リアルタイムOS関連ファイル						
trcinf.c						
	ファイルを生成する					
	割り込みや例外エントリに配置する命令を記え	もしたエントリ・ファイルを作成す	るかどうかを選択します。			
	シュニナ・ア・フィギュト・ション・フラノル明					
	1EIF					* *
	Iccord					
		-	illatin Lass s			
	9べ(のメッセーシ / *ビルド・ツール / *	テハック・ツール (*ラピッド・	ヒルド人 チャッシュ・レ			-
	▲ 出力 2017ラー一覧				law	1
F1 F2 F3	F4 F5 F6	F7	FB	F9 Fit	FN FN	H2
					1 行 1 桁 編集不可	▲ 非接続

図 8-14 エントリ・ファイルの出力設定変更

● [ファイルを生成する]:「はい(.cfg ファイル変更時に更新しない)(-ne)」を選択してください。

8.9. E1 エミュレータを使用して「ハードウェア・トレース・モードでトレー ス・チャートを取得する」を選択した場合の注意事項

E1 エミュレータを使用して「ハードウェア・トレース・モードでトレース・チャートを取得する」を使用した場合, 取得できるトレースの容量に制限があるため,測定可能な区間が限定されることに注意してください。取得できるトレ ース容量についての詳細は,E1/E20 エミュレータのユーザーズマニュアルを参照してください。RI850V4 V2 では, 「ソフトウェア・トレース・モードでトレース・チャートを取得する」を使用することを推奨します。

8.10. I/O ヘッダ・ファイル生成ツールを使用する場合の注意事項

I/O ヘッダ・ファイル生成ツールが生成した I/O ヘッダ・ファイルと、RI850V4 V2 のヘッダ・ファイルを同時に使用する場合、ヘッダ・ファイル内の定義が競合する可能性があります。この場合、重複している I/O ヘッダ・ファイル内のマクロの名称を別の名称に変更してください。

8.11. CPU クロックの周波数について

RH850 はクロックコントローラを備えており,所定の手順で CPU クロックの周波数を変更することができます(変更方法の手順については,ハードウェアのユーザーズマニュアルを参照してください)。なお, RI850V4 V2 のサンプル プログラムでは, CPU クロックの変更を行っていないため,デフォルトの周波数 (8MHZ)で動作します。

CPU クロックの周波数を変更し、かつトレース機能を使用する場合はマクロ KERNEL_USR_BASETIME の設定が 必要となります。本設定は、OS タイマの1カウントあたりの経過時間(単位:ns)を指定します。1カウントあたりの 経過時間は、OS タイマを駆動するクロックの周波数から算出します。OS タイマは CPU クロック周波数の2分周で動 作するため、デフォルトでは4MHZ となります。



9. 制限事項

9.1. CS+ for CC 使用時の制限事項

9.1.1. リアルタイム OS ビルド設定プラグイン

下記に現状の制限事項を記載します。

- (1) プリプロセッサを起動する際 一時ファイル名が IDE の出力パネルに表示されてしまう制限事項 プロジェクトに登録されているコンフィギュレーション・ファイルをコンパイラのプリプロセッサに入力し ますが、コンフィギュレーション・ファイルにマクロ定義を行っているとき、かつ、そのマクロ定義に構文 の誤りがある場合、エラー対象ファイル名がプロジェクトに登録されているコンフィギュレーション・ファ イル名ではなく、コンフィギュレータが一時的に作成した一時ファイル(cf850***.c)のファイル名を出力 します。一時ファイルは使用後にすぐ削除するため、出力パネルからエラー行にジャンプすることができま せん。
- (2) ビルド・モード未対応の制限事項

下記の制限により、複数のビルド・モードを使用しないでください。

- ビルド・モードごとにコンフィギュレータのオプションを保存しません。そのため、複数のビルド・
 モードを作成しても、すべてのビルド・モードで同じコンフィギュレータ・オプションで起動します。
- ビルド・モードを切り替えるたびに、ビルド・ツールの「追加のインクルード・パス」に kernel_id.h へのパスが追加されてしまいます。正しいパスはビルド設定プラグインが「システム・インクルード・パス」に設定していますが、IDE が「追加のインクルード・パス」に、ビルド・モードを切り替える前のパスを設定してしまい、ビルド時に IDE が設定したパスを先行して参照します。ビルド・モードを切り替えた後に kernel_id.h が変更されるようなコンフィギュレーション・ファイル編集を行った場合、その変更がビルドに反映されないことになります。
- (3) 流用プロジェクト機能に関する制限

流用元のプロジェクトに sit.s などの情報ファイルが存在しない(クリーンされている状況)かつ,流用元 のファイルを「コピーして流用プロジェクトを作成する」という操作が行われた場合,本来グレー表示でプ ロジェクト・ツリーに登録されている sit.s ファイルなどがプロジェクト・ツリーから削除されてしまいま す。 9.1.2. リアルタイム OS 解析制御プラグイン

特になし。

- 9.1.3. リアルタイム OS リソース情報表示プラグイン
 - (1) 待ちタスク表示(子ノード表示)で表示リセットを選択すると、タスク・タブの表示がリセットされる制限
 待ちタスクのカラム情報をリセットすると、タスクのカラム情報もリセットします。ただし、表示情報の内容としては問題ありません。
- 9.1.4. リアルタイム OS タスク・アナライザ・プラグイン
 - (1) CPU使用率カラムに対してフィルタリング操作を行ったとき、正しい結果が得られないことがある制限 フィルタリング内容によって、正しくフィルタリングされた結果が表示されないことがあります。例えば CPU使用率を「80%以上の表示」を指定したとき、フィルタリングされずにすべてのカラムが表示されて しまいます。

以上



改訂記録

		改訂内容		
Rev.	発行日	ページ	ポイント	
1.00	2015.9.30		新規発行	

	ご注意書き
 本資料 て、回訳 者に生 	こ記載された回路、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 各、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、お客様または第三 こた損害に関し、当社は、一切その責任を負いません。
2. 本資料(の誤り)	こ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 こ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。
3. 本資料(に対す・ 諾する [:]	こ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 る侵害に関し、当社は、何らの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、著作権その他の知的財産権を何ら許 ものではありません。
4. 当社製 5. 当社は、	品を改造、改変、複製等しないでください。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。 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	k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、 家雷、工作機械、パーソナル機器、産業用ロボット等
	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、各種安全装置等
当社製 せるお 途に当 合わせ	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 それのある機器・システム(原子力制御システム、軍事機器等)に使用されることを意図しておらず、使用することはできません。 たとえ、意図しない用 ±製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。 なお、ご不明点がある場合は、当社営業にお問い ください。
6. 当社製 品をご	いここ。 品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の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 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。
7. 当社は、 す。また ないよ [、] を行っ [・]	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 と、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、社会的損害等を生じさせ 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 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 RoHS排 関して、	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 i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 当社は、一切その責任を負いません。
9. 本資料(た、当 国為替)	こ記載されている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 1±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に使用しないでください。当社製品または技術を輸出する場合は、「外 &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。
10. お客様の 担して]	D転売等により、本ご注意書き記載の諸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、当社は何らの責任も負わず、お客様にてご負 頁きますのでご了承ください。
11. 本資料(D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。
注1. 本資 をī	【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 直接または間接に保有する会社をいいます。
注2. 本資	5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。

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

■営業お問合せ窓口

Г

© 2014, 2015 Renesas Electronics Corporation

Colophon 2.0