

この度は、RL78 ファミリ リアルタイム OS RI78V4 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.	RI78V4 V1.00.02 との相違点	11

6.1.	カーネルのバージョン情報	11
6.2.	対応コンパイラ	11
6.3.	対応デバイス	11
6.4.	メモリモデルの変更	11
6.5.	アセンブリ言語における 16 進数の書き方の変更	11
6.6.	構造体パッキング機能を使用可能	11
6.7.	カーネル初期化処理のアドレス変更	12
6.8.	セクション名の変更/追加	12
6.9.	データ・キュー機能追加	13
6.10.	周期ハンドラに PHS 属性追加	14
6.11.	割り込みハンドラの定義方法・記述方法の改善	14
6.12.	タイマ・ハンドラの定義方法	16
6.13.	コンフィギュレータ (CF78V4) 関連	17
6.13.1.	コンフィギュレータ (CF78V4) の変更点	17
6.13.2.	システム・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	17
6.14.	リアルタイム OS タスク・アナライザ対応	19
7.	RI78V4 V1.00.02 から RI78V4 V2.00.00 に移植する場合	21
7.1.	タイマ割り込みハンドラ	21
7.2.	割り込みハンドラの定義	22
7.3.	周期ハンドラの定義の項目	23
7.4.	サービス・コールの引数の <code>_far</code> 修飾子	23
7.5.	リンク・ディレクティブ・ファイル	23
8.	アップデートした製品の変更点	24
8.1.	リアルタイム OS タスク・アナライザ・プラグイン	24
9.	注意事項	25
9.1.	CS+ のプロジェクト作成	25
9.1.1.	本製品添付のサンプル・プロジェクトを使用する	25
9.1.2.	新しいプロジェクトを流用する	26
9.1.3.	RI78V4 のプロジェクトを流用する	27
9.2.	プラグインの有効化	28
9.3.	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	29
9.4.	リアルタイム OS リソース情報パネルに関する注意事項	30
9.4.1.	参照はリアルタイム OS 初期化後に行う	30
9.4.2.	デバッグ情報を生成したプログラムを使用する	30
9.5.	リアルタイム OS タスク・アナライザに関する注意事項	31
9.5.1.	トレース・モード変更	31
9.5.2.	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	31
9.5.3. E1/E20 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」する場合	33
9.5.4. デバッグ・ツールの設定	33
9.5.5. トレースのタイムスタンプについて	33
9.6. セクション・スキップ機能	34
9.7. カーネル・ソース・コードのビルド方法	35
9.8. サンプル・プログラムをビルドしたときのワーニングについて	35
10. 制限事項	36
10.1. RI78V4 V2	36
10.1.1. トレース機能	36
10.2. CS+ for CC 使用時の制限事項	36
10.2.1. リアルタイム OS ビルド設定プラグイン	36
10.2.2. リアルタイム OS リソース情報表示プラグイン	36
10.2.3. リアルタイム OS タスク・アナライザ・プラグイン	36
11. サンプル・プログラム	37
11.1. ボードサポートパッケージモジュールを組み込んだサンプル・プログラム	37
11.1.1. 概要	37
11.1.2. BSP 対応サンプル・プログラムの構成	37
11.1.3. BSP 対応サンプル・プロジェクトのフォルダ構成	38
11.1.4. BSP 対応サンプル・プロジェクトの設定	38
11.1.5. BSP 対応サンプル・プロジェクトの注意点	39
11.1.6. RI78V4 V2.00.00 パッケージ付属サンプル・プロジェクトとの相違点	39
改訂記録	40

1. 製品構成

RI78V4 V2.00.00 は型名により、契約形態と提供物が異なります。

- トライアル版

型名	契約形態
RTRRL7800TR01ERRZZ	トライアル版、インストール可能な PC は 1 台

提供物は、リアルタイム OS RI78V4 V2.00.00 カーネル オブジェクトのトライアル版（使用制限あり）、および、コマンドライン・コンフィギュレータ CF78V4 で、Web サイトからのダウンロードによる提供となります。

なお、CS+ for CC プラグインを使用する際は、Web サイトから個別にダウンロードしてください。

- 評価契約・量産契約

型名	契約形態	提供物
RTRRL7800TR01ERR	評価契約、インストール可能な PC は 1 台	A
RTRRL7800TR01ERRLU	評価契約、インストール可能な PC は無制限	A
RTRRL7800TR01RRRJUL	量産契約、量産数は 3000 台まで	A
RTRRL7800TR01RRRUJ	量産契約、量産数は無制限	A
RTRRL7800TR01SRRUJ	量産契約、量産数は無制限、ソース・コード付き	B

提供物は以下となります。

提供物	ツール名	バージョン	
B	A	リアルタイム OS RI78V4 カーネル・オブジェクト	V2.00.00
		コマンドライン・コンフィギュレータ CF78V4	V2.01.00.01
		CS+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）	V3.01.00.01
		リアルタイム OS ビルド設定プラグイン（RI78V4 依存部）	V1.00.00.04
		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）	V3.01.00.01
		リアルタイム OS 解析制御プラグイン（RI78V4 依存部）	V1.00.00.04
		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）	V3.01.00.01
		リアルタイム OS タスク・アナライザ・プラグイン（共通部）	V3.00.01.01
		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（RI78V4 依存部）	V1.00.00.03
			リアルタイム OS RI78V4 カーネル ソース・コード

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 ユーザーズマニュアル 起動編	R20UT0751JJ0104
RI78V4 V2.00.00 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	R20UT3375JJ0100
RI78V4 V2.00.00 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	R20UT3374JJ0100
RI78V4 V2.00.00 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	R20UT3373JJ0100
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	R20UT0756JJ0104

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。

- RL78 ファミリ (S2 コア、S3 コア)

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)
- ・ メイン・メモリ : 1G バイト以上 (Windows 10, および 64 ビット版の Windows は 2G バイト以上), 推奨 2G バイト以上
- ・ 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 8.1 (32bit 版, 64bit 版)
- ・ Windows 10 (32bit 版, 64bit 版)
- ・ Microsoft .NET Framework 4.5.2 + 言語パック(推奨)
- ・ Microsoft Visual C++ 2015 Update 3 ランタイム・ライブラリ(x86)

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+	ルネサス エレクトロニクス	V8.05.00 以降
C コンパイラ CC-RL	ルネサス エレクトロニクス	V1.10.00 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+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。

コントロールパネルの「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 32 ビット版で、システムドライブが C:の場合
C:¥Program Files¥Common Files¥Renesas Electronics CubeSuite+¥
- Windows®が 64 ビット版で、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。「9.2 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+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 RI78V4 Plugins
- CS+ Realtime OS RI78V4 Object Release（量産契約、ソース・コード付き「以外」の場合）
- CS+ Realtime OS RI78V4 Source Release（量産契約、ソース・コード付きの場合）

6. RI78V4 V1.00.02 との相違点

6.1. 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ようになります。

項目	変更前	変更後
TKERNEL_PRVER	0x0102	0x0200

6.2. 対応コンパイラ

RI78V4 V1.00.02 の対応コンパイラは CA78K0R ですが、RI78V4 V2.00.00 では CC-RL になりました。

6.3. 対応デバイス

RI78V4 V1.00.02 では 78K0R マイクロコントローラ、および、RL78 ファミリに対応していましたが、RI78V4 V2.00.00 から RL78 ファミリのみの対応となり、78K0R は非対応となりました。

6.4. メモリモデルの変更

RI78V4 V1.00.02 ではラージ・モデル対応でしたが、CC-RL ではラージ・モデルのコードを出力せず、ミディアム・モデル、スモール・モデルのコードを出力する仕様であるため、RI78V4 V2.00.00 はミディアム・モデル対応となりました。なお、RI78V4 V2.00.00 はスモール・モデルのアプリケーション上でも動作します。

6.5. アセンブリ言語における 16 進数の書き方の変更

RI78V4 V1.00.02 では、16 進数は suffix 形式（例：012ab）での記述方法でしたが、RI78V4 V2.00.00 は prefix 形式（0x12ab）での記述方法に変更しました。

6.6. 構造体パッキング機能を使用可能

RI78V4 V1.00.02 では、構造体パッキング機能（CA78K0R の -rc オプション）を使用できませんでしたが、RI78V4 V2.00.00 は使用可能（CC-RL の -pack オプション）になりました。

6.7. カーネル初期化処理のアドレス変更

カーネル初期化処理のアドレスを変更しました。

V1.00.02	V2.00.00
__urx_start	__kernel_start

6.8. セクション名の変更／追加

RI78V4 が使用するメモリ領域名（V1.00.02 ではセグメント、V2.00.00 ではセクション）を変更しました。また機能拡充のために V2.00.00 で新規に追加した領域も存在します。

V1.00.02 と V2.00.00 におけるメモリ領域名の対応表は以下のようになります。

V1.00.02	V2.00.00
k_system	.kernel_system
k_info	.kernel_info
k_const	.kernel_const / .kernel_const_f
k_data	.kernel_data
k_stack	.kernel_stack
k_work0、k_work1、k_work2、k_work3	.kernel_work0、.kernel_work1、.kernel_work2、.kernel_work3

V2.00.00 で追加になったメモリ領域名と、その領域に配置するものは以下のようになります。

追加領域名	
.kernel_system_timer_n	システム・タイマ割り込み領域 / FAR 分岐情報領域
.kernel_data_init	カーネル初期化データ領域
.kernel_data_trace_n	トレースデータ領域
.kernel_const_trace_f	トレース取得用データ領域
.kernel_system_trace_f	トレース取得用コード
.kernel_sbss	カーネル使用 SADDR 領域 ※

※ RI78V4 V1.00.02 でもカーネルが使用している SADDR 領域はありましたが、RI78V4 V2.00.00 ではその領域をセクションで区切りました。

6.9. データ・キュー機能追加

RI78V4 V2.00.00 ではデータ・キュー機能が追加になりました。以下のサービス・コールが追加になっています。

サービス・コール名	機能
snd_dtq	データ・キューへの送信を待つ
psnd_dtq	データ・キューへの送信を待つ（ポーリング）
ipsnd_dtq	データ・キューへの送信を待つ（ポーリング・非タスクコンテキスト）
tsnd_dtq	データ・キューへの送信を待つ（タイムアウト指定）
fsnd_dtq	データ・キューへの強制送信
ifsnd_dtq	データ・キューへの強制送信（非タスクコンテキスト）
rcv_dtq	データ・キューからの受信を待つ
prcv_dtq	データ・キューからの受信を待つ（ポーリング）
trcv_dtq	データ・キューからの受信を待つ（タイムアウト指定）
ref_dtq	データ・キューの状態を参照する

また、システム・コンフィギュレーション・ファイルにおいて、以下のデータ・キューの定義が追加になります。

```

CRE_DTQ (
    dtqid {                // ID
        dtqatr,           // 属性
        dtqcnt [:sec_nam] , // データ数、メモリ領域名
        dtq               // システム予約
    }
)

```

6.10. 周期ハンドラに PHS 属性追加

RI78V4 V2.00.00 では周期ハンドラの属性に PHS 属性（起動位相保存属性）が追加になりました。属性はシステム・コンフィギュレーション・ファイルにおいて設定します。

```
CRE_CYC (  
    cycid {          // ID  
        cycatr,      // 属性  
        exinf,       // システム予約  
        cychdr,      // 起動アドレス  
        cyctim,      // 起動周期  
        cycphs      // 初期起動位相  
    }  
)
```

6.11. 割り込みハンドラの定義方法・記述方法の改善

割り込みハンドラの定義方法、および、記述方法を改善しました。

- RI78V4 V1.00.02 での割り込みハンドラの定義方法と記述方法

【C ソースファイル】

```
#include "kernel_id.h"  
  
#pragma rtos_interrupt INTP0 intp0_hdr  
  
void  
intp0_hdr(void) {  
    (割り込みハンドラ本体処理)  
    return;  
}
```

#pragma rtos_interrupt 指令と割り込みハンドラ本体をユーザが記述する必要があります。

- RI78V4 V2.00.00 での割り込みハンドラの定義方法と記述方法

【システム・コンフィギュレーション・ファイル】	【C ソースファイル】
<pre>DEF_INH (INTP0 { TA_HLNG, intp0_hdr })</pre>	<pre>#include "kernel_id.h" void intp0_hdr(void){ (割り込みハンドラ本体処理) return; }</pre>

システム・コンフィギュレーション・ファイルにて割り込みハンドラを定義することにより、対応する #pragma rtos_interrupt 指令は、コンフィギュレータ (CF78V4) が kernel_id.h に出力します。よって、ユーザは kernel_id.h のインクルードと割り込みハンドラ本体を記述します。

また、システム・コンフィギュレーション・ファイルの割り込みハンドラ定義において FAR 属性を指定し、割り込みハンドラ本体に `__far` 修飾子をつけることによって FAR 領域への配置が可能です。この場合、割り込みベクタと割り込みハンドラ本体の間に位置する中間地点 (ベクタから分岐し、そこで割り込みハンドラへさらに分岐するコード) はコンフィギュレータ (CF78V4) が自動的に出力します。

【システム・コンフィギュレーション・ファイル】	【C ソースファイル】
<pre>DEF_INH (INTP0 { TA_HLNG TA_FAR, intp0_hdr })</pre>	<pre>#include "kernel_id.h" __far void intp0_hdr(void){ (割り込みハンドラ本体処理) return; }</pre>

6.12. タイマ・ハンドラの定義方法

RI78V4 V1.00.02 では、タイマ・ハンドラをコールするための「タイマ割り込みハンドラ」ユーザが記述する必要がありました。V2.00.00 では記述する必要がなくなりました。ただし、システム・コンフィギュレーション・ファイルにて、基本クロック要因割り込みを定義する必要があります。

- RI78V4 V1.00.02 でのタイマ・ハンドラの定義方法

```
#pragma rtos_interrupt INTTM00 int_timer

void
int_timer(void){

    Timer_Handler();
    return;

}
```

#pragma rtos_interrupt 指令とタイマ割り込みハンドラ本体をユーザが記述する必要があります。

- RI78V4 V2.00.00 での定義方法

【システム・コンフィギュレーション・ファイル】

```
CLK_INTNO (INTTM00) // 割り込み要因 INTTM00 を RI78V4 のタイマとする場合
```

これにより、コンフィギュレータが、対応する#pragma rtos_interrupt 指令を kernel_id.h に、タイマ割り込みハンドラ本体を割り込み情報定義ファイルに出力します。

6.13. コンフィギュレータ（CF78V4）関連

6.13.1. コンフィギュレータ（CF78V4）の変更点

コンフィギュレータの変更点は以下のとおりです。

(1) **【追加】** -cpu オプション追加

使用するデバイス名を指定します。

【指定例】

```
-cpu R5F10A6A
```

なお、CS+を使用する場合は、本オプションを自動的に使用します。

(2) **【追加】** -devpath オプション追加

デバイス・ファイルの存在するパスを指定します。

【指定例】

```
-devpath="C:\Program Files\Renesas Electronics\CS+\CC\Device\RL78\Devicefile"
```

なお、CS+を使用する場合は、本オプションを自動的に使用します。

6.13.2. システム・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

システム・システム・コンフィギュレーション・ファイルの変更点は以下のとおりです。

(1) **【追加】** 基本クロック用タイマ割り込み要因 CLK_INTNO

RI78V4 で使用するタイマ割り込みを指定します。

【指定例】

```
CLK_INTNO ( INTTM00 );
```

(2) **【追加】** データ・キュー情報

データ・キューの情報を指定します。

(3) **【追加】** 割り込みハンドラ情報

割り込みハンドラ情報を指定します。

(4) **【追加】** 周期ハンドラの PHS 属性

周期ハンドラの属性として PHS 属性が追加となりました。

(5) 【変更】 ID に指定できる文字数

キーワード「ID」に指定できる文字数を 24 文字以内から 255 文字以内に変更しました。

(6) 【変更】 シンボル名やオブジェクト名に指定できる文字数

シンボル名やオブジェクト名に指定できる文字数を 30 文字以内から 4095 文字以内に変更しました。

6.14. リアルタイム OS タスク・アナライザ対応

RI78V4 V1.00.02 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「AZ78K0R」を提供していましたが、RI78V4 V2.00.00 からは「リアルタイム OS タスク・アナライザ」を提供します（AZ78K0R は提供しません）。リアルタイム OS タスク・アナライザは、CS+用プラグインとして提供します。詳しい使用方法は「RI78V4 V2.00.00 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 6-1 AZ78K0R のイメージ図

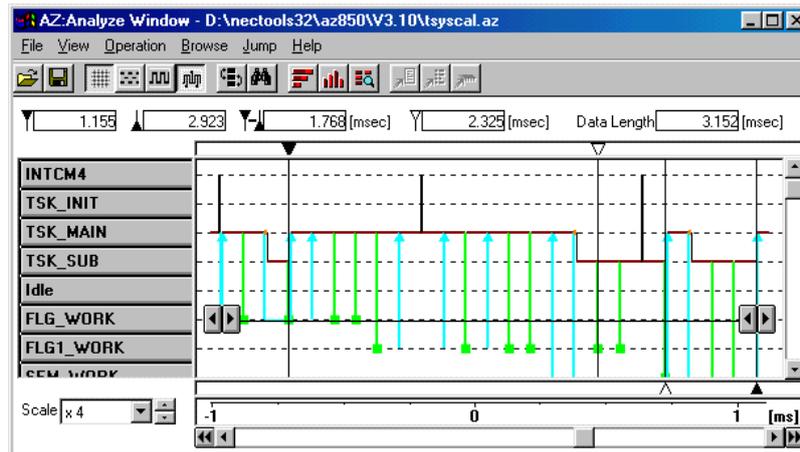
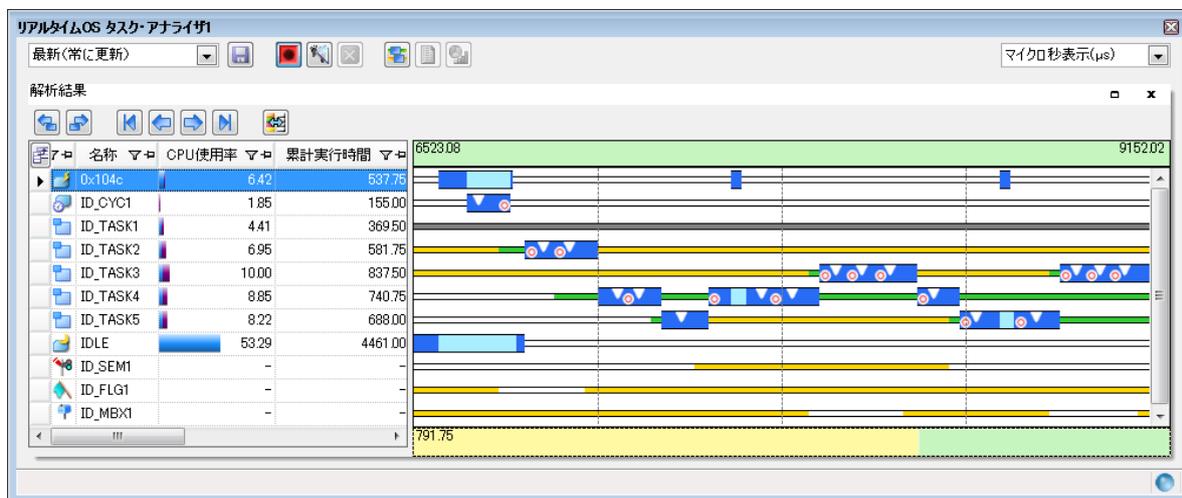
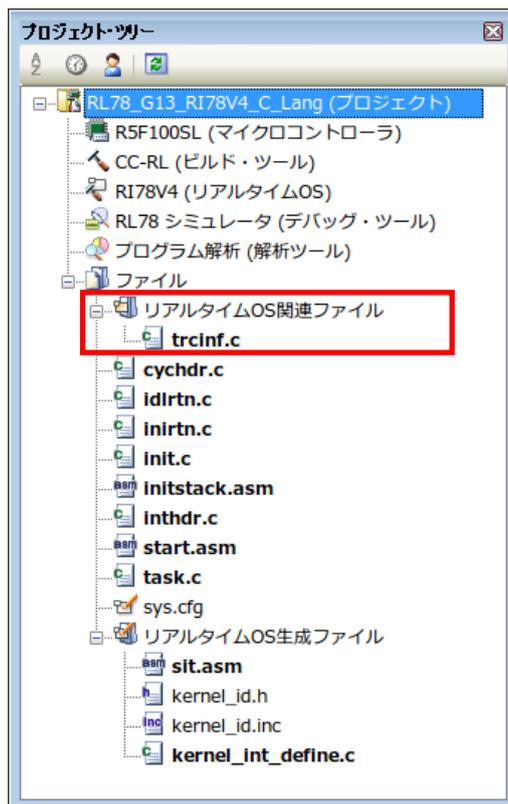


図 6-2 タスク・アナライザのイメージ図



また、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 6-3 リアルタイム OS 関連ファイル



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+\SampleProjects\RL78\RL78_G13_RI78V4_C_Lang\appli\include
```

このファイル内では、基本クロック用タイマのカウンタ・レジスタの I/O アドレス、および基本クロック用タイマのコンペア・レジスタの I/O アドレスをマクロ定義します。詳細については「RI78V4 V2.00.00 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。

7. RI78V4 V1.00.02 から RI78V4 V2.00.00 に移植する場合

RI78V4 V1.00.02 アプリケーションから RI78V4 V2.00.00 アプリケーションに移植する場合の変更点を下記に示します。

なお、CubeSuite+(CS+)で RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを作成していた場合、RI78V4 V2.00.00 をインストールした環境で RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込むことで、変換可能な箇所は自動的に変換します。

7.1. タイマ割り込みハンドラ

RI78V4 V1.00.02 を使用したアプリケーションでは、タイマ割り込みハンドラを記述する必要がありましたが、この仕様に変更になりました。まず、V1.00.02 で作成したタイマ割り込みハンドラを削除します。そして、システム・コンフィギュレーション・ファイルにて基本クロック用割り込み要因 (CLK_INTNO) を指定します。

<p>■ V1.00.02</p> <p>【C ソースファイル】</p> <pre>#pragma rtos_interrupt INTPO timer void timer(void) { TimerHandler(); return; }</pre>	<p>■ V2.00.00</p> <p>【システム・コンフィギュレーション・ファイル】</p> <pre>CLK_INTNO (INTTM00);</pre>
--	--

なお、CS+では、RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込むことで、システム・コンフィギュレーション・ファイルをコンバートしますが、本項目に関しては、使用するタイマ割り込みを特定できないため、コメントにて本項目を指定する旨を入れてあります。コンバート後にコメントのある箇所を上記の指定例にしたがって変更してください。

7.2. 割り込みハンドラの定義

RI78V4 V1.00.02 では、システム・コンフィギュレーション・ファイルで割り込みハンドラ定義を行なっていませんでしたが、RI78V4 V2.00.00 では行う必要があります。RI78V4 V1.00.02 を使用したアプリケーションで #pragma rtos_interrupt 指令で指定していた「割り込み要因」「割り込みハンドラアドレス」を使用し、システム・コンフィギュレーション・ファイルにて「DEF_INH」を使用した定義を行ってください。そして、RI78V4 V1.00.02 を使用したアプリケーションで指定していた #pragma rtos_interrupt 指令は削除してください。

なお、CS+で RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込んでも、本項目は自動的に変換しません。

<p>■ V1.00.02</p> <p>【C ソースファイル】</p> <pre>#pragma rtos_interrupt INTP0 intp0_hdr void intp0_hdr(void){ (割り込みハンドラ本体処理) return; }</pre>	<p>■ V2.00.00</p> <p>【システム・コンフィギュレーション・ファイル】</p> <pre>DEF_INH (INTP0 { TA_HLNG, intp0_hdr })</pre> <p>【C ソースファイル】</p> <pre>#include "kernel_id.h" void intp0_hdr(void){ (割り込みハンドラ本体処理) return; }</pre>
--	---

7.3. 周期ハンドラの定義の項目

RI78V4 V2.00.00 では、周期ハンドラの初期起動位相の保存を行うようになりました。CRE_CYC の項目の予約項目だった箇所に、周期ハンドラの初期起動位相を指定する必要があります。RI78V4 V1.00.02 では、0 を指定する必要がありましたが、RI78V4 V2.00.00 では 0 を指定するとエラーになります。0 以外の値を指定してください（周期ハンドラの起動周期を入力してもよいです）なお、CS+で RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込んでも、本項目は自動的に変換しません。

■ V1.00.02	■ V2.00.00
<p>【システム・コンフィギュレーション・ファイル】</p> <pre> CRE_CYC (ID_CYC1 { TA_HLNG, 0, cycadr 0x1000, 0x0 }) </pre>	<p>【システム・コンフィギュレーション・ファイル】</p> <pre> CRE_CYC (ID_CYC1 { TA_HLNG, 0, cycadr 0x1000, 0x1000 }) </pre>

7.4. サービス・コールの引数の__far 修飾子

RI78V4 V1.00.02 を使用したアプリケーションで、サービス・コールの引数に__far 修飾子を付けている場合、RI78V4 V2.00.00 では削除するか__near 修飾子に変更してください。なお、CS+で RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込んでも、本項目は自動的に変換しません。

なお本項目の変更理由は「6.4 メモリモデルの変更」によるものです。

7.5. リンク・ディレクティブ・ファイル

RI78V4 V1.00.02 で対応していたコンパイラ CA78K0R では、配置情報を「リンク・ディレクティブ・ファイル」にて記載していましたが、RI78V4 V2.00.00 で対応する CC-RL はリンクのオプション (-start) で指定する方式に変更になりました。

CS+では、RI78V4 V1.00.02 を使用したアプリケーションのプロジェクトを読み込むことで、リンクのオプションに変換しますが、配置アドレスに関しては見直してください。リンクのオプションの設定方法についてはユーザーズマニュアル「CC-RL コンパイラ」を参照してください。

8. アップデートした製品の変更点

本パッケージからアップデートした製品に関する変更点を下記に示します。なお、本パッケージには含まれていませんので、CS+のアップデート機能によりアップデートを行ってください。

8.1. リアルタイム OS タスク・アナライザ・プラグイン

- (1) トレース・チャート（処理プログラムの実行遷移状況、リアルタイム OS 資源の利用状況）の拡大／縮小方法の変更

「トレース・モードの選択」で、「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」、または「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」を選択した場合に表示されるトレース・チャートの拡大／縮小の仕様を変更しました。詳しい使用方法は「RI78V4 V2.00.00 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。

変更後のバージョンは、以下の通りです。

ツール名	バージョン
リアルタイム OS タスク・アナライザ・プラグイン（パネル部）	V3.01.00.08

9. 注意事項

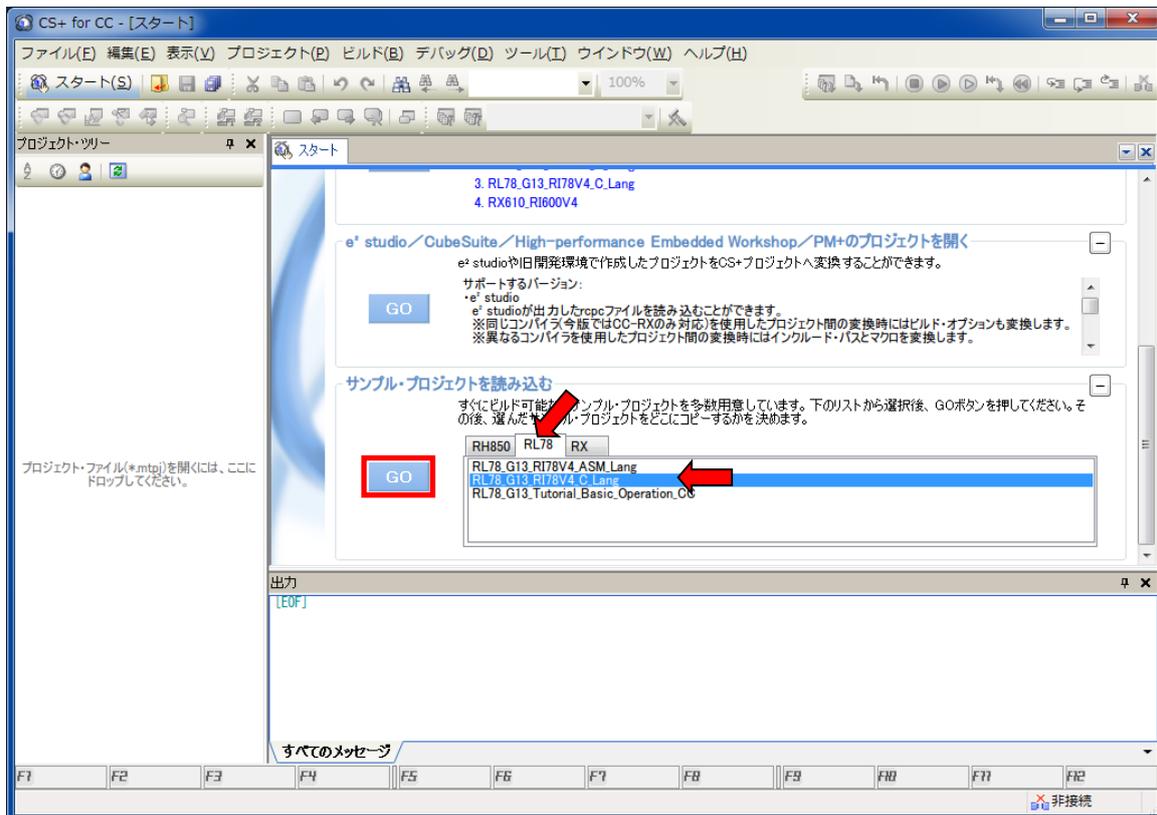
9.1. CS+ のプロジェクト作成

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
- RI78V4 V1.00.02 のプロジェクトを流用する

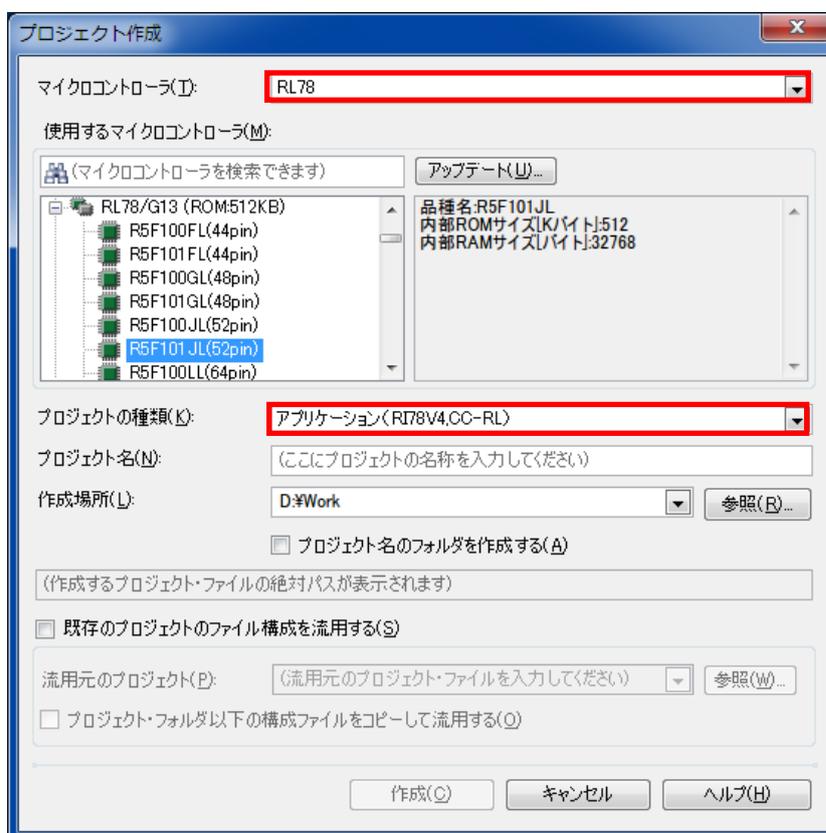
9.1.1. 本製品添付のサンプル・プロジェクトを使用する

CS+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「RL78」タブを選択し、「RL78_G13_RI78V4_C_Lang」または「RL78_G13_RI78V4_ASM_Lang」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。



9.1.2. 新しいプロジェクトを流用する

CS+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。



- [マイクロコントローラ] : 「RL78」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 (RI78V4、 CC-RL)」を選択してください。

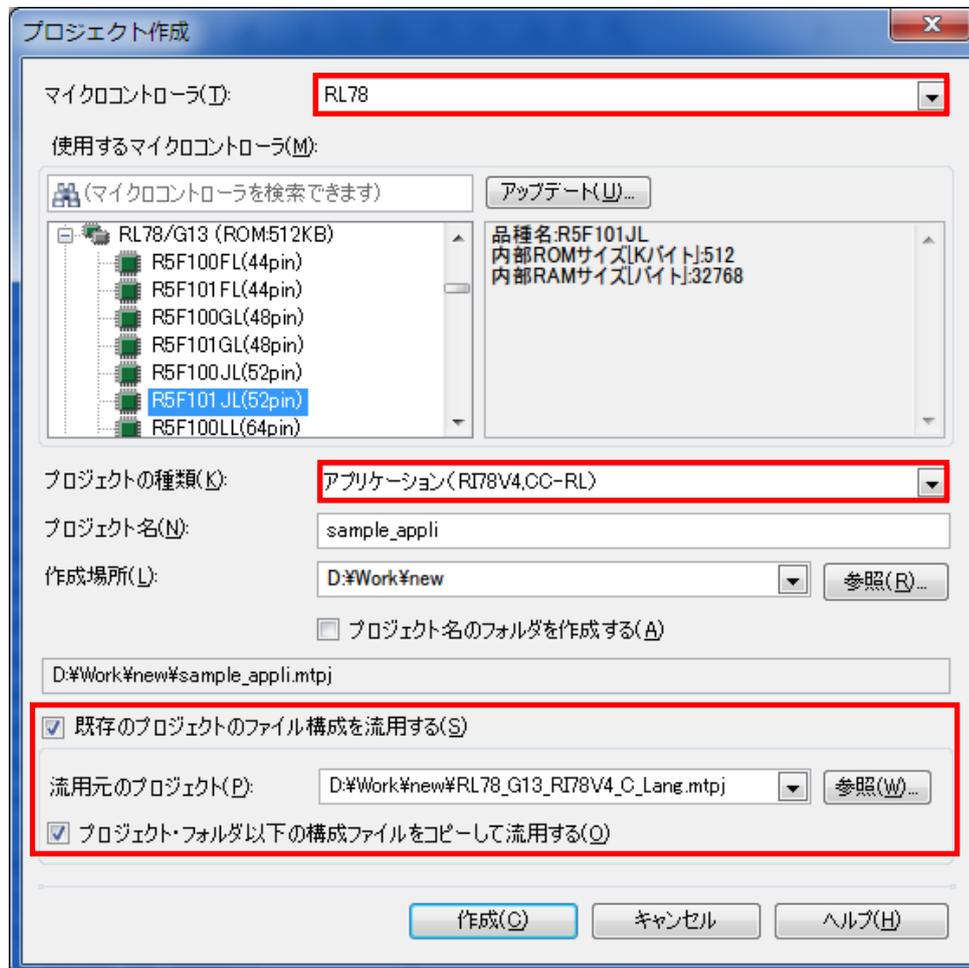
[作成] ボタンを押すと、プロジェクトが作成されます。

※ RI78V4 プロジェクトでは、iodefine.h は、自動的にプロジェクト・ツリーに登録しません。iodefine.h は、必要であればビルド・ツールのプロパティ内の「I/O ヘッダ・ファイル生成オプション」を操作してください。ビルド時にプロジェクトに登録する設定を行うことで使用できるようになります。

9.1.3. RI78V4 のプロジェクトを流用する

CS+で作成した RI78V4 (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+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。



- [マイクロコントローラ] : 「RL78」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 (RI78V4、 CC-RL)」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。

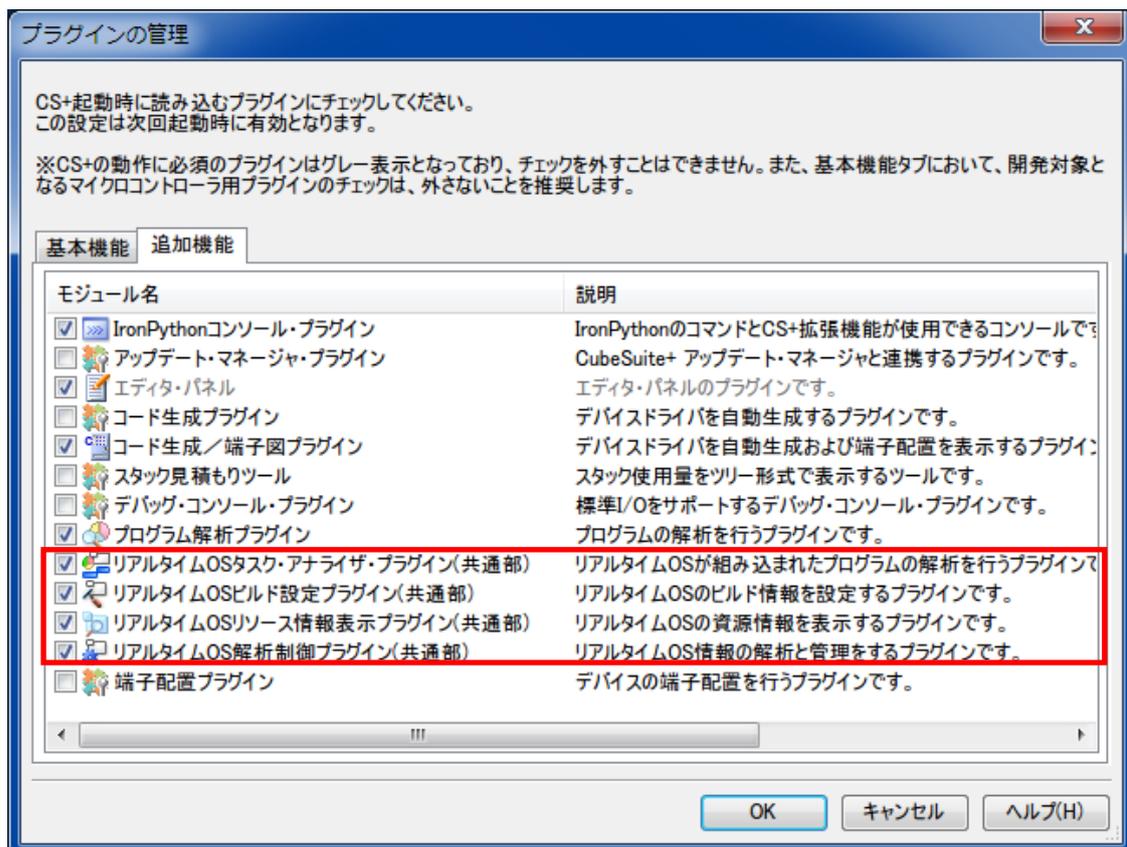
[作成] ボタンを押すと、プロジェクトが作成されます。

9.2. 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+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+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。

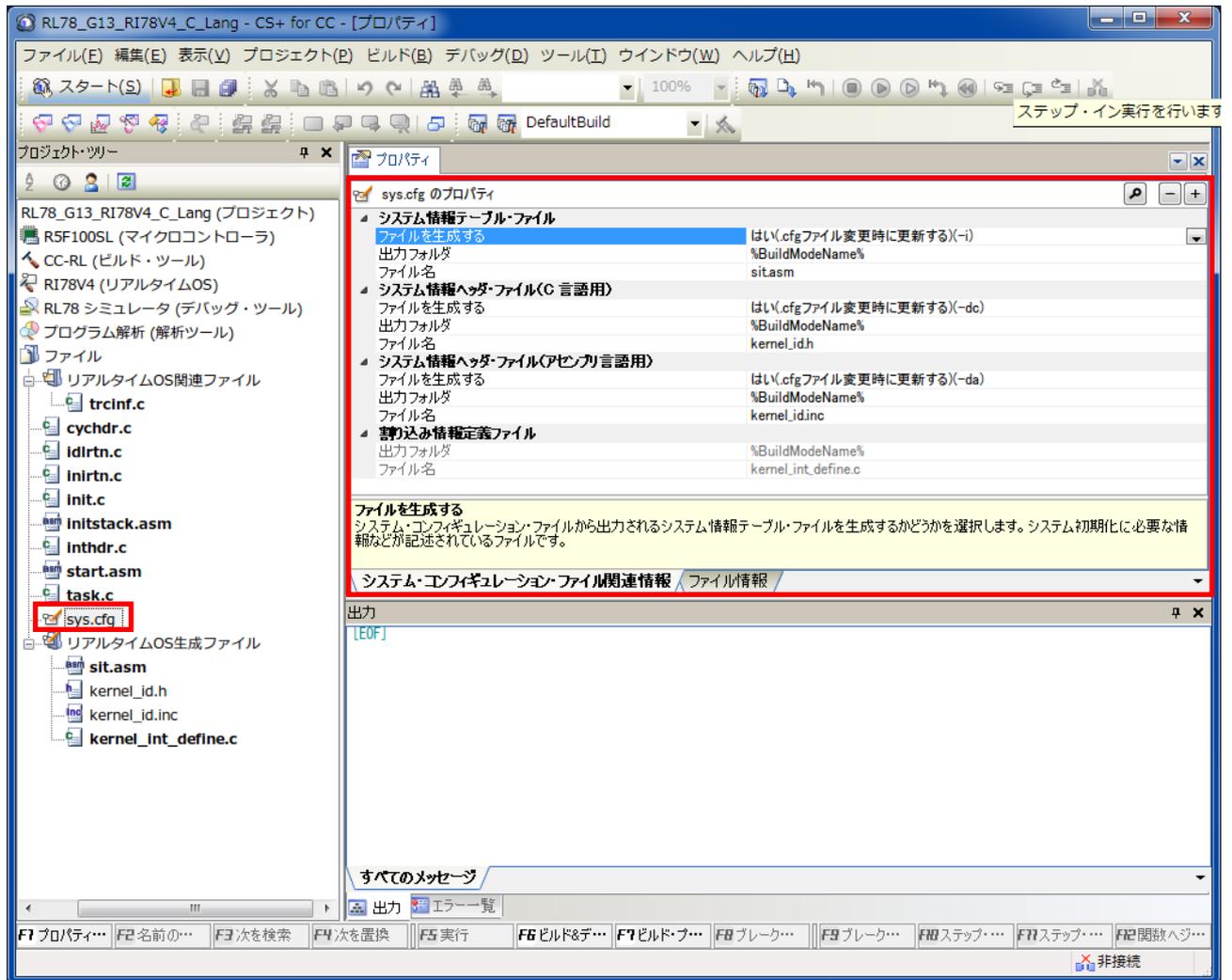
- リアルタイム OS ビルド設定プラグイン (共通部)
- リアルタイム OS 解析制御プラグイン (共通部)
- リアルタイム OS リソース情報表示プラグイン (共通部)
- リアルタイム OS タスク・アナライザ・プラグイン (共通部)



9.3. 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

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。

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は [プロジェクト・ツリー] 内にあるシステム・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。



[システム情報テーブル・ファイル] [システム情報ヘッダ・ファイル (C 言語用)] と [システム情報ヘッダ・ファイル (アセンブリ言語用)] は RI78V4 V1.00.02 と設定内容、設定項目は同じです。

[割り込み情報定義ファイル] は RI78V4 V1.00.02 にはない項目ですが、変更ができない項目です。

9.4. リアルタイム OS リソース情報パネルに関する注意事項

9.4.1. 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。

9.4.2. 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。

9.5. リアルタイム OS タスク・アナライザに関する注意事項

9.5.1. トレース・モード変更

RI78V4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。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。

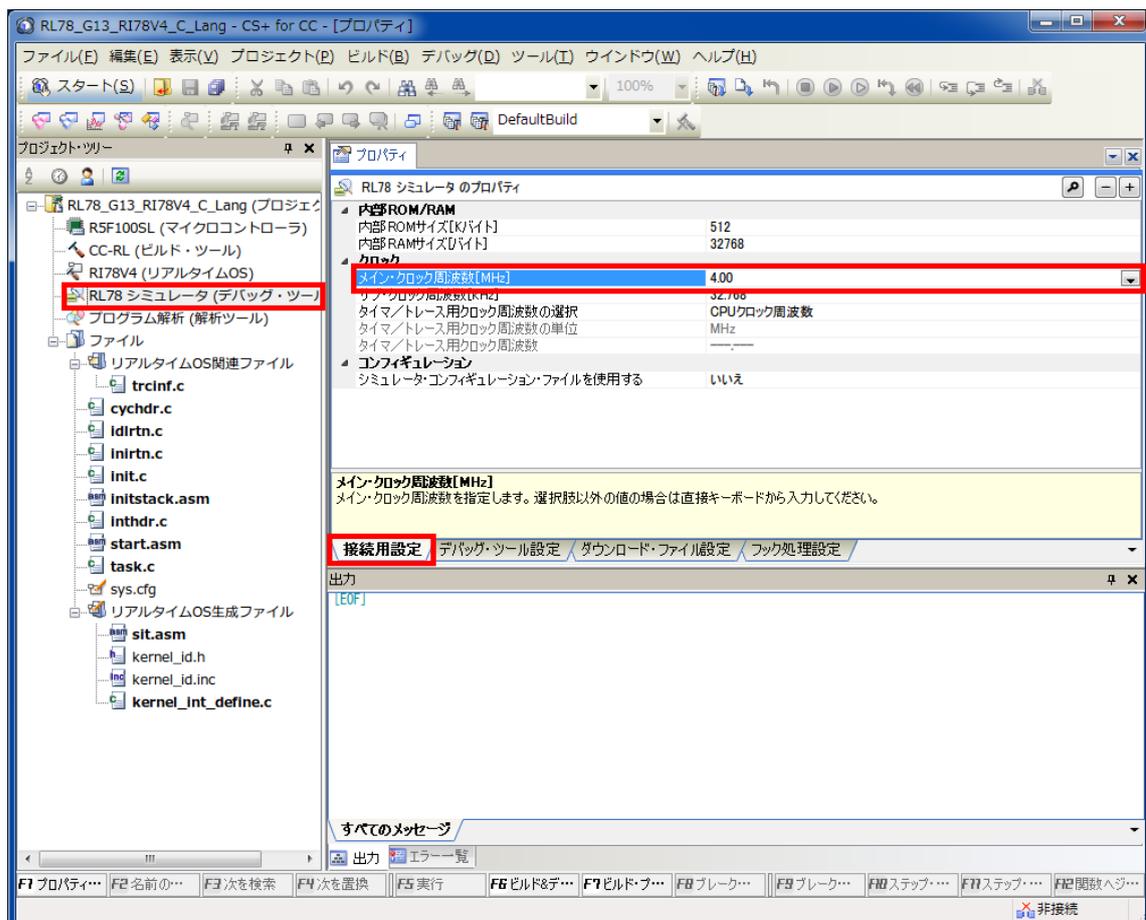
9.5.2. シミュレータを使用して「ハードウェア・トレース・モードで、トレース・チャートを取得」する場合

RL78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、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

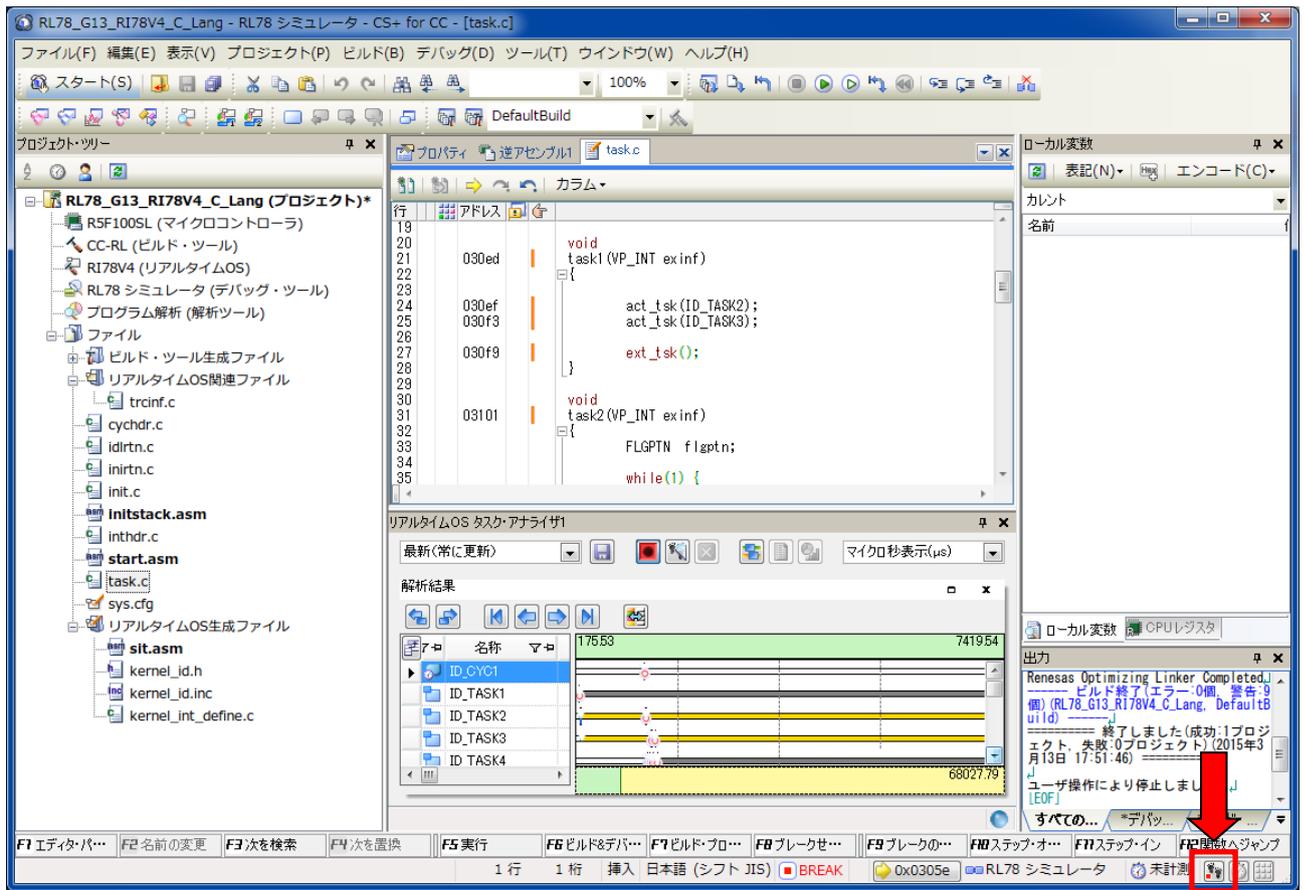
- メイン・クロック周波数 [MHz]

RL78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 9-1 参照）

図 9-1 RL78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



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）。



9.5.3. E1/E20 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」する場合

E1/E20 エミュレータを使用してハードウェア・トレース・モードを使ってトレース・チャートを取得することはできません。E1/E20 エミュレータを使用する場合は、「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」もしくは「ソフトウェア・トレース・モードで、長時間統計を取得」を選択してください。

9.5.4. 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

9.5.5. トレースのタイムスタンプについて

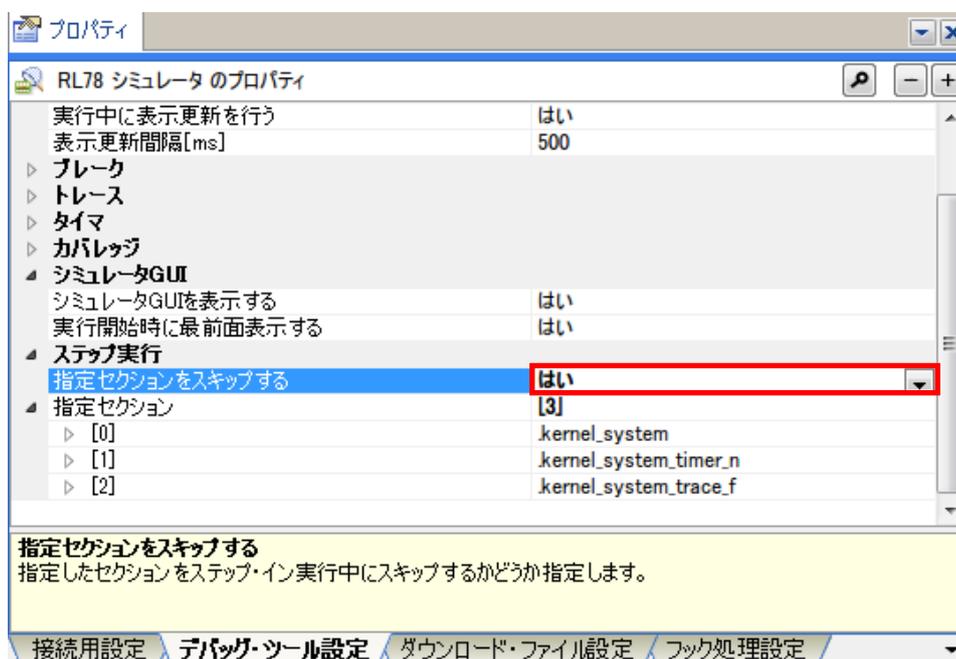
トレースのタイムスタンプは、カーネルのタイマ機能を使用して実現しています。カーネルのタイマは OS タイマ割り込みを使用して実現しているため、割り込み禁止状態の場合は、タイマ割り込み処理が保留されます。タスクなどで割り込み禁止にし、その期間が 1ms 以上であった場合は正しい時間を表示できません（処理順は正しく表示します）。

9.6. セクション・スキップ機能

CS+を使用したデバッグ環境で、ステップ実行を行ったときに、RI78V4 のカーネルコードをスキップする機能を追加しました。RI78V4 のサービス・コール発行箇所でのブレークした後に「ステップイン」を実行した場合、RI78V4 のカーネルコードをスキップし続け、カーネルコード外に到達したときにブレークします。

この機能を使用すると、サービス・コール発行によってディスパッチが発生する場合、サービス・コール発行箇所での「ステップイン」を実行すると、ディスパッチ後のタスクでブレークさせることができます。

ただし、このセクション・スキップ機能は、デフォルトでは OFF になっています。この機能を ON にするためには、デバッグのプロパティの「デバッグ・ツール設定」の「ステップ実行」にある「指定セクションをスキップする」を「はい」に設定してください。なお、指定セクションについては自動的に設定されています。



9.7. カーネル・ソース・コードのビルド方法

R178V4 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「RTRRL7800TR01SRRUU（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%r178_ccrl%medium」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。

- ビルド方法

Cygwin ウィンドウ上で上記 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%r178_ccrl%medium
```

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%r178_ccrl%medium」にコピーしてください。

9.8. サンプル・プログラムをビルドしたときのワーニングについて

付属のサンプル・プログラムをビルドしたとき、数個のワーニングが出力されます。これらのワーニングは、サンプル・プログラムをベースに開発する等、このサンプルを他の品種で動作させやすくするために、あえて除去していません。最終的にプログラムを確定する際には、ワーニングをすべて除去することを推奨します。

10. 制限事項

10.1. RI78V4 V2

10.1.1. トレース機能

RI78V4 V2 とリアルタイム OS タスク・アナライザ・プラグインによって提供されているトレース機能のうち、長時間統計トレース・モードは使用してはいけません。長時間統計トレース・モードを使用すると、不特定の RAM 領域が破壊されてしまうことがあります。

10.2. CS+ for CC 使用時の制限事項

10.2.1. リアルタイム OS ビルド設定プラグイン

(1) 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。正しいパスはリアルタイム OS 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システム・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となります。

10.2.2. 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。

10.2.3. 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。

(2) ハードウェア・トレース・モードに関する制限事項

E2 エミュレータ/E2 エミュレータ Lite を使用してハードウェア・トレース・モードを使ってトレース・チャートを取得することはできません。

11. サンプル・プログラム

11.1. ボードサポートパッケージモジュールを組み込んだサンプル・プログラム

本章では、ルネサスボードサポートパッケージ（以下 BSP と称す）モジュールを組み込んだサンプル・プログラムについて説明します。

11.1.1. 概要

本サンプル・プログラムは、RI78V4 V2.00.00 パッケージ付属のサンプル・プログラムに、BSP モジュールを組み込んで構成されています。本サンプル・プログラムの動作は、BSP モジュールを使用してクロックの発振を行った後、RI78V4 V2.00.00 パッケージ付属のサンプル・プログラムと同様のマルチタスク処理を行います。

BSP モジュールの使用方法は、「RL78 ファミリ ボードサポートパッケージモジュール Software Integration System アプリケーションノート (R01AN5522JJxxxx)」を参照してください。

11.1.2. BSP 対応サンプル・プログラムの構成

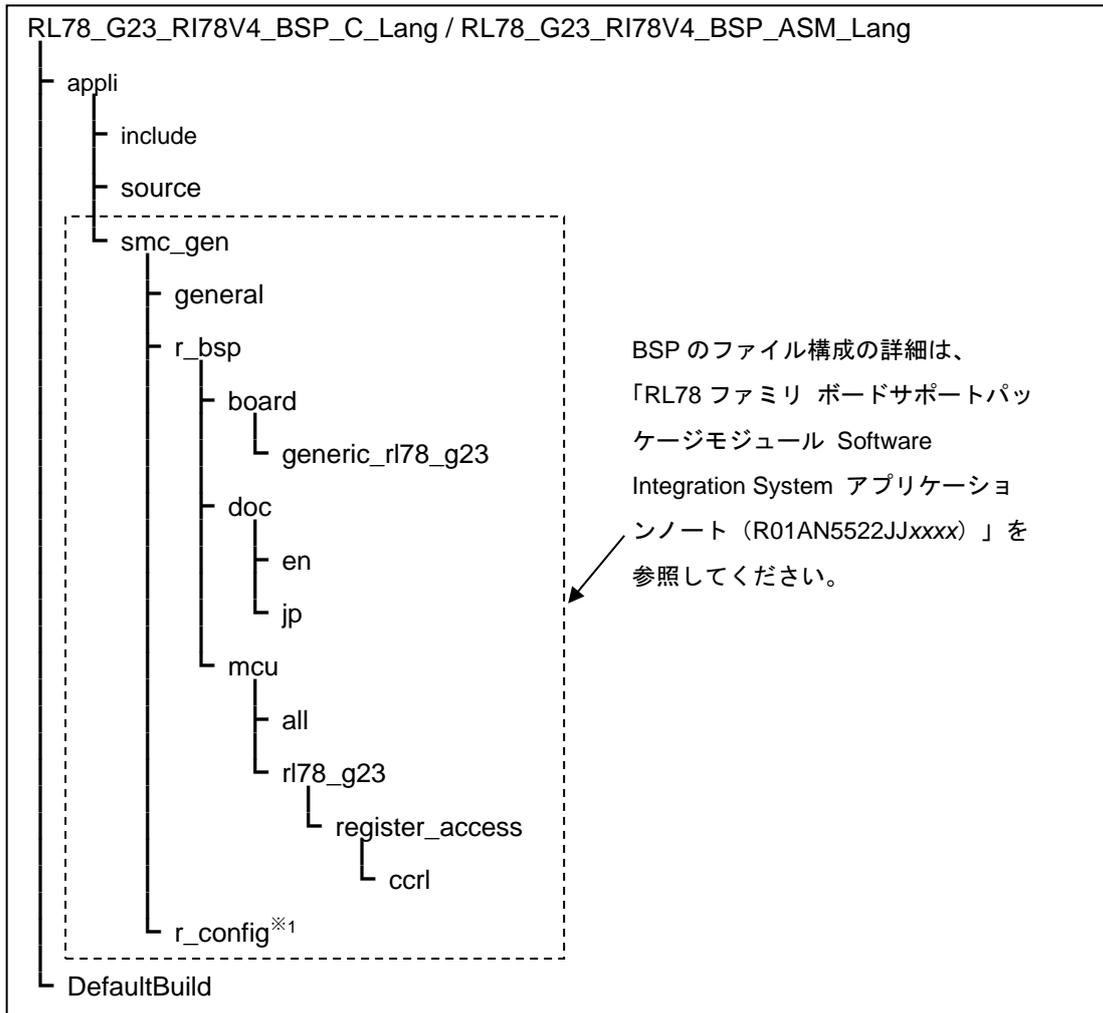
BSP 対応サンプル・プログラムは、CS+のサンプル・プロジェクトとして提供されています。BSP 対応サンプル・プロジェクトの構成は以下の通りです。

サンプル・プロジェクト名	内容
RL78_G23_RI78V4_BSP_C_Lang	タスク等、RI78V4 V2.00.00 管理オブジェクトのソースコードを C 言語で記述
RL78_G23_RI78V4_BSP_ASM_Lang	タスク等、RI78V4 V2.00.00 管理オブジェクトのソースコードをアセンブリ言語で記述

モジュール名	ファイル名	リビジョン
BSP	r_bsp	v1.10

11.1.3. BSP 対応サンプル・プロジェクトのフォルダ構成

BSP 対応 RI78V4 V2.00.00 サンプル・プロジェクトのフォルダ構成を以下に示します。



※1 : BSP コンフィグレーションファイルの配置フォルダ

11.1.4. BSP 対応サンプル・プロジェクトの設定

(1) 定義マクロ

アセンブル・オプションの定義マクロに「BSP_RTOS」を設定してください。

(2) BSP コンフィグレーションファイルへの設定

ファイル名	定義	値	設定内容
r_bsp_config.h	BSP_CFG_RTOS_USED	4	ルネサス エレクトロニクス製 uITRON OS RI78V4 V2
	BSP_CFG_CONFIGURATOR_SELECT	0	Smart Configurator を使用しない
r_bsp_config.inc	BSP_CFG_ASM_RTOS_USED	4	ルネサス エレクトロニクス製 uITRON OS RI78V4 V2

11.1.5. BSP 対応サンプル・プロジェクトの注意点

- (1) BSP 対応サンプル・プロジェクトは、Smart Configurator を使用することはできません。
- (2) BSP 対応サンプル・プロジェクトは、RI78V4 V2.00.00 パッケージには同梱されていません。入手方法は、ルネサス エレクトロニクスの Web サイトからダウンロードしてください

11.1.6. RI78V4 V2.00.00 パッケージ付属サンプル・プロジェクトとの相違点

相違点	パッケージ付属サンプル・プロジェクト	BSP 対応サンプル・プロジェクト
マイクロコントローラ	RL78/G13	RL78/G23
デバッグ・ツール	シミュレータ	E2 Lite
スタートアップ	パッケージ付属のスタートアップを使用	BSP 付属のスタートアップを使用
クロック発振の設定	hdwinit 関数でクロック制御レジスタへの設定	初期化ルーチン inthdr1 で R_BSP_StartClock を使用して設定
C 言語ファイルでの kernel.h / kernel_id.h インクルード方法	kernel.h / kernel_id.h を直接インクルード	platform.h をインクルード ※ kernel.h / kernel_id.h は r_rtos.h でインクルードされている。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2015.3.25	—	新規発行
1.01	2016.3.31	24	リアルタイム OS タスク・アナライザ・プラグインのアップデート情報を追加しました。 トレース・チャート（処理プログラムの実行遷移状況、リアルタイム OS 資源の利用状況）の拡大／縮小方法を変更しました。
1.02	2021.4.28	7	ハードウェア環境を最新の情報に更新しました。
		7	ソフトウェア環境を最新の情報に更新しました。
		7	対応ツールを最新のバージョンに更新しました。
		36	リアルタイム OS タスク・アナライザ・プラグインの長時間統計トレース・モードを使用できない制限事項を追加しました。
		36	RL78/G23 の環境では、リアルタイム OS タスク・アナライザ・プラグインのハードウェア・トレース・モードを使用できない制限事項を追加しました。
		37	BSP 対応サンプルプログラムを新規に追加しました。

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。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、お客様の責任において、お客様の機器・システムを設計ください。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. 当社製品または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。
3. 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。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング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。
6. 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、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）、交通管制（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、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、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（宇宙機器と、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想定していません。たとえ、当社が想定してい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。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 8. 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。
 9. 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、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っていません。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。
 10. 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。
 11. 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。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。
 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
 13. 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。
 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。