

この度は、統合開発環境 CS+をご使用いただきまして、誠にありがとうございます。

この添付資料では、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等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。

目次

第1章	対象デバイスについて.....	2
第2章	ユーザーズ・マニュアルについて.....	3
第3章	アンインストール方法.....	4
第4章	機能改善・変更点.....	5
第5章	注意事項.....	9
第6章	制限事項.....	16

第1章 対象デバイスについて

統合開発環境 CS+がサポートする対象デバイスに関しては、WEB サイトに掲載しています。

こちらをご覧ください。

CS+製品ページ：

<https://www.renesas.com/cs+>

第2章 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・マニュアルは、次のようになります。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
CS+ V8.02.00 インストーラ編	R20UT4525JJ0100
CS+ V8.06.00 エディタ編	R20UT4972JJ0100
CS+ V8.07.00 Python コンソール編	R20UT5056JJ0100
CS+ V5.00.00 アップデータ編	R20UT3942JJ0100
CS+ V8.07.00 メッセージ編	R20UT5057JJ0100
CS+ V8.07.00 プロジェクト操作編	R20UT5058JJ0100
CS+ V8.06.00 解析ツール編	R20UT4979JJ0100
CS+ V8.07.00 RH850 デバッグ・ツール編	R20UT5061JJ0100
CS+ V8.07.00 RX デバッグ・ツール編	R20UT5060JJ0100
CS+ V8.07.00 RL78 デバッグ・ツール編	R20UT5059JJ0100
CS+ V8.07.00 GHS CCRH850 ビルド・ツール操作編	R20UT4971JJ0100

* GHS : Green Hills Software, LLC

第3章 アンインストール方法

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、Windows の設定の「アプリと機能」や、コントロール・パネルの「プログラムと機能」から、「CS+ for CC」を選択してください。

第4章 機能改善・変更点

本章では、CS+の V8.06.00 から V8.07.00 の機能改善・変更点について説明します。

4.1 CS+の全体の機能改善・変更点

4.1.1 メイン・ウインドウのパネル表示エリアに表示されるボタン描画を改善

特定の環境下において、メイン・ウインドウのパネル表示エリアの右端に表示されるボタンの背景が見づらくなる現象を改善しました。

4.2 プロジェクト管理機能の機能改善・変更点

4.2.1 環境依存情報ファイル (*.mtud) の保存と共有方法を選択できるよう改善

環境依存情報ファイル (*.mtud) をユーザ間で共有する、または保存しないように設定するためのプロパティを追加しました。これにより情報ファイルのファイル名にユーザ名を含む/含まないなどを選択できるようになります。

4.2.2 マイクロコントローラの表示順の改善【RL78】【RH850】

以下のダイアログでのマイクロコントローラの表示順を、アルファベット順、ROM サイズ順にし、マイクロコントローラを見つけやすくしました。

- ・プロジェクト作成 ダイアログ
- ・プロジェクト変換設定 ダイアログ
- ・マイクロコントローラの変更 ダイアログ

4.3 ビルド・ツールの機能追加・機能改善

4.3.1 ルネサス製コンパイラのサポート拡充【RL78】【RH850】【RX】

次のコンパイラのサポートを追加しました。

- ・CC-RL V1.11.00
- ・CC-RH V2.04.00
- ・CC-RX V3.04.00

4.3.2 使用するコンパイラ・パッケージのメジャー・バージョンの変化に気づくよう改善

コンパイラのメジャー・バージョンの変化に気付きやすくするため、ビルドに使用するコンパイラ・パッケージのメジャー・バージョンに変化があった場合に出カパネルに情報メッセージを表示するようにしました。

4.3.3 [ビルド成功時に品質記録情報をファイル出力する(Q)]のデフォルト値の変更

オプション ダイアログ [全般 - ビルド] の [ビルド成功時に品質記録情報をファイル出力する(Q)] チェック・ボックスのデフォルト値をチェックありとするとともに、品質記録情報ファイルにビルドの出力ファイル (*.abs, *.mot) のファイルパスとそのハッシュ値 (SHA1) を出力するようにしました。これにより、品質記録情報ファイルと出力ファイルの紐付けが可能となります。

4.3.4 マルチコアのプロジェクト作成時の-Xcpu オプションの初期値の改善【RH850】

スタート パネルの [新しいマルチコアプロジェクトを作成する] からマルチコアのプロジェクトを [コア数分のアプリケーション・プロジェクトとして作成] で作成する場合において、指定したマイクロコントローラの各コアのコア種別が異なる場合に、マルチコア用アプリケーション プロジェクトの-Xcpu オプション初期値を適切にセットするようにしました。そのようなデバイスには RH850/E1L (PE1: -Xcpu=g3m PE3: -Xcpu=g3k) などがあります。詳細はデバイスのマニュアルをご参照ください。

4.3.5 [出力開始アドレスのアライメント] プロパティの設定可能値の改善【RL78】 【RH850】 【RX】

ビルド・ツールのプロパティ [ヘキサ出力オプション] タブの [出力開始アドレスのアライメント] プロパティの値に 16 進数固有の文字 (A~F) を入力できるようにしました。

4.3.6 [利用可能な場合、依存関係の更新にコンパイラを使用し、解析速度より解析精度を上げる(C)] 有効時のラピッド・ビルドの改善

オプション ダイアログ [全般 - ビルド] の [利用可能な場合、依存関係の更新にコンパイラを使用し、解析速度より解析精度を上げる(C)] が有効時、C/C++ソース・ファイルでインクルードされているヘッダファイルを上書き保存したときも、ラピッド・ビルドが発動するようにしました。

4.4 デバッグ・ツールの機能追加・機能改善

文中において以下の略称を使用しています。

OCD(Serial)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータ, E2 エミュレータ Lite の Serial / FINE 接続

OCD(JTAG)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータ, E2 エミュレータ Lite の JTAG 接続

OCD(LPD)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータの LPD 接続

4.4.1 周辺・消費電流シミュレーション機能の対応品種・周辺機能拡充【RL78】

【対象】シミュレータ, RL78/G10, RL78/G11, RL78/G12, RL78/G13, RL78/G13A, RL78/G14, RL78/L12, RL78/G1F, RL78/G23

周辺シミュレーション機能の対応周辺機能を追加し、消費電流シミュレーションの対応品種を追加しました。本機能を使用することで、実際にマイコン(ボード)に接続せずに、パソコン上の仮想的な環境だけで周辺機能や消費電流をシミュレーションすることができます。

次の周辺機能に対応しました。

- ・シリアル・アレイ・ユニットの簡易 I²C 機能

次のマイクロコントローラの消費電流シミュレーションの対応品種を追加しました。

- ・RL78/G23 (注)

注：消費電流シミュレーション機能は一部の品種のみ対応しています。

4.4.2 仮想ボードパネルのサポート部品拡充【RL78】

【対象】シミュレータ, RL78/G10, RL78/G11, RL78/G12, RL78/G13, RL78/G13A, RL78/G14, RL78/L12, RL78/G1F, RL78/G23

仮想ボードパネルでサポートする部品を追加しました。

- ・UART コンソール

4.4.3 リードのみ可能な I/O レジスタへの書き込み動作の改善【RH850】

【対象】シミュレータ, RH850

RH850 命令シミュレータでは I/O レジスタのアクセス属性を R/W (リード/ライト可能) とし、一部のレジスタを除き、リードのみ可能な I/O レジスタへの書き込みができるようになりました。これにより IOR パネル上や Python コンソール経由でリードのみ可能な I/O レジスタに疑似的に値を書き込み、シミュレーションを行うことができます。

リードのみ可能な I/O レジスタの書き込みに関する注意事項は、CS+ RH850 命令シミュレータ V5.05.00 リリースノートをご参照ください。

4.5 Python コンソールの機能改善

4.5.1 IronPython を 2.7.11 に更新

4.5.2 Python プロパティの追加

以下の Python プロパティを追加しました。

プロパティ名	機能概要
build.Common.DataEndian	データのエンディアンの設定／参照を行います。【CC-RX】
build.Common.PrecisionOfDoubleType	double 型、および long double 型の精度の設定／参照を行います。【CC-RX】
build.Common.UseDPFPU	倍精度浮動小数点処理命令を使用するかどうかの設定／参照を行います。【CC-RX】
build.Library.EnableMathH	mathf.h(C89/C99) を有効にするかどうかの設定／参照を行います。【CC-RX】
build.Library.EnableMathfH	math.h(C89/C99) を有効にするかどうかの設定／参照を行います。【CC-RX】
build.Link.RangeOfDebugMonitorArea	デバッグ・モニタ領域の範囲の設定／参照を行います。【CC-RL】
build.Link.SetDebugMonitorArea	デバッグ・モニタ領域の設定／参照を行います。【CC-RL】
debugger.Option.CpuEndian	CPUエンディアン(シミュレータ)またはエンディアン(シミュレータ以外)を設定／参照します。【RX】

以下の Python 関数は CS+ for CC には対応していないため、ヘルプおよびユーザーズ・マニュアルから削除しました。

プロパティ名	機能概要
debugger.Upload.IntelIdTag	メモリ・データを ID タグ付きインテル形式で保存します。
debugger.Upload.MotorolaIdTag	メモリ・データを ID タグ付きモトローラ形式で保存します。
debugger.Upload.Tektronix	メモリ・データを テクトロニクス形式で保存します。
debugger.Upload.TektronixIdTag	メモリ・データを ID タグ付き テクトロニクス形式で保存しません。

第5章 注意事項

本章では、注意事項について説明します。

5.1 CS+全体の注意事項

5.1.1 ファイル名に関する注意事項

フォルダ名、ファイル名に関しては次の注意事項があります。

- ・ フォルダ名、ファイル名

Windows のエクスプローラーで作成することのできないフォルダ名とファイル名は、使用しないでください。

ネットワークパス名は使用できません。ドライブ名に割り当てて使用してください。

- ・ ソース・ファイル名とロード・モジュール・ファイル名とプロジェクト・ファイル名

ファイル名は、a-z, A-Z, 0-9, .(ピリオド), _(アンダスコア), +, - のいずれかの文字で構成されます。ファイル名の先頭と最後に、.(ピリオド)の文字は使えません。

ファイル名の先頭に「+」(プラス) / 「-」(マイナス)は使えません。

英大文字(A-Z), 英小文字(a-z)は区別されません。

ファイル名は、パスを含めて最大 259 文字です。

ファイル名が同じソース・ファイルは使用しないで下さい。異なるパスに存在していても区別できません。

- ・ 上記以外のファイル名

Windows のファイル名規約に準拠します。

なお、ファイル名には次の文字は使えません。

¥ / : * ? " < > | ;

ファイル名の先頭と最後に.(ピリオド) とスペースは使えません。

英大文字(A-Z), 英小文字(a-z)は区別されません。

ファイル名は、パスを含めて最大 259 文字です。

- ・ フォルダ名

Windows のファイル名規約に準拠します。

なお、ファイル名には次の文字は使えません。

() , =

5.1.2 パネル表示に関する注意事項

使用するハードウェア環境が CS+ の推奨サポート環境を下回るスペックである場合、[プロパティ] パネルのサイズを小さくすると表示内容が乱れることがあります。

その場合には、分割パネル領域から [プロパティ] パネルを外に出してください。

- ・ ドッキング可能を ON にして、ドッキング・パネル化する
- ・ フローティングを ON にして、フローティング・パネル化する

5.1.3 Windows の更新プログラムに関する注意事項

Windows 用の更新プログラムによっては、Windows 8.1 で CS+ for CC を起動するとエラー発生または PC がクラッシュする場合があります。

この場合は、以下のいずれかを実施してください。

- ・最新の VC++ 2015 ランタイム(x86 版)をインストールする。
- ・最新の Windows 用の更新プログラムを適用する。

5.1.4 エディタ・パネルに関する注意事項

- ・ラベルを選択して、コンテキスト・メニューの「関数または変数へジャンプ」機能を使用した場合、ラベルにジャンプしません。
- ・メイン・プロジェクトとサブプロジェクトに、パスの違う同名のソース・ファイルが登録されていて、メイン・プロジェクトとサブプロジェクトのロード・モジュールを両方ダウンロードしたとき、次のようになります。
 - 当該ファイルでは、メイン・プロジェクトのアドレスが表示される
 - 当該ファイルの逆アセンブルから「ソースへジャンプ」を行うと、メイン・プロジェクトに登録されているファイルが開く
 - どちらのプロジェクトから当該ファイルを開いても 1 つのファイルしか開けない
- ・Windows 8.1, Windows 10 では、アンチエイリアスが有効になり表示が不鮮明になる場合があります。
- ・名前を付けて保存ダイアログにおいてファイルを保存する場合、拡張子を入力しない場合には、ファイルの種類ドロップダウン・リストで選択されている最初の拡張子が自動的に付加されます。ただし、ファイルの種類ドロップダウン・リストで選択されている拡張子および Windows で登録されている拡張子を付加してファイル名を入力した場合、拡張子は付加されません。
自動的に拡張子が付加されてしまった場合は、エクスプローラー等でファイルをリネームしてください。
- ・.NET Framework 4.6 がインストールされた環境では、エディタ・パネルに表示した内容を印刷できません。CS+以外のエディタを使用して印刷をしてください。

5.1.5 新規にプロジェクトを作成する際の注意事項

【対象】RX

RX 開発環境にて、” 空のアプリケーション(CC-RX)” のプロジェクトを作成してビルドした場合、以下のエラーが出る場合があります。

- ```
** L2132 (E) Cannot find "D" specified in option "rom"
** L2132 (E) Cannot find "D_1" specified in option "rom"
** L2132 (E) Cannot find "D_2" specified in option "rom"
```

エラーが発生した場合は、リンク・オプションの” ROM から RAM へマップするセクション” の設定を変更してください。

### 5.1.6 チュートリアルに関する注意事項

チュートリアルでは、コード生成プラグイン、端子配置プラグイン、プログラム解析プラグインを使用します。プラグイン管理ダイアログで使用するプラグインを有効にしてください。

### 5.1.7 CS+の複数起動の注意事項

CS+は、同じホストマシン上で複数起動が可能ですが、次の注意事項があります。

- ・ CS+を複数起動した場合、パソコンのユーザ毎の情報ファイルは最後に書き込んだ情報が保存されます。
- ・ CS+を複数起動した場合、スタック見積もりツール（CallWalker 含む）の情報ファイルは最後に書き込んだ情報が保存されます。
- ・ 複数起動した CS+で、同一プロジェクト・ファイルを使用した場合、最後に書き込んだ情報が保存されます。
- ・ 複数起動した CS+で、同一プロジェクト・ファイルを使用した場合、同時にビルドしないでください。出力ファイルが同一のためです。

### 5.1.8 作成したバージョンより古いバージョンで読み込む際の注意事項

作成したプロジェクトを、そのプロジェクトを作成した CS+より古いバージョンの CS+で読み込むと、一部の設定がクリアされる場合があります。これは、新たに追加された機能などの新しいバージョンでのみ認識できる項目が、古いバージョンでは認識できないためです。

### 5.1.9 マイクロソフト株式会社.NET Framework に関する注意事項

マイクロソフト株式会社製の.NET Framework のバージョンが 4.5.2 未満の場合に、次のメッセージを出力して終了します。

E0200010

本製品の起動に失敗しました。

お使いの PC に Microsoft .NET Framework 4.5.2 以上をインストールしてください。

マイクロソフト株式会社のホームページから最新の.NET Framework をダウンロードしてインストールしてから、CS+ for CC を起動してください。

CS+ for CC のアップデート・マネージャでは、.NET Framework のアップデートはできませんので、ご注意ください。

### 5.1.10 コードフラッシュのデュアルバンク機能に関する注意事項

【対象】RX65N-2M, RX651-2M, RX72M グループ

RX65N/RX651 の ROM 2MB 品および 1.5MB 品、RX72M の全製品に、コードフラッシュメモリのデュアルバンク機能があります。リニアモードまたはデュアルモードの選択はプロジェクト設定で行います。

デュアルモードを選択する場合は\_DUAL 付きの品種名を選択してください。

### 5.1.11 スマート・マニュアルに関する注意事項

【対象】RL78・RX

スマート・マニュアルに対応していないマイコンのプロジェクトを開いている場合は、スマート・マニュアルでマニュアルが表示されません。

### 5.1.12 パートナーOS 対応デバッグプラグインに関する注意事項

**【対象】** RL78・RX

パートナーOS 対応デバッグプラグインを有効にして CubeSuite+.exe(メイン・ウインドウ表示なし)で CS+を起動すると「(0202002)プロジェクトの読み込みに失敗しました。」エラーになります。

以下のいずれかの方法で回避できます。

- ・ CubeSuite+.exe にパートナーOS 対応デバッグプラグインを読み込まないオプションを指定して起動する。

例) CubeSuite+.exe /np PartnerOS ...

- ・ パートナーOS 対応デバッグプラグインをオフにしてから CubeSuite+.exe(メイン・ウインドウ表示なし)で起動する。

プラグインのオフの仕方

メイン・ウインドウ表示ありの CS+あるいは CubeSuite+(CubeSuiteW+.exe)を起動する。

[ツール(T)]メニュー[プラグインの管理(P)...]を指定するとプラグイン管理ダイアログが表示される。

[パートナーOS 対応デバッグプラグイン]のチェックを外す。

### 5.1.13 RX72M 使用時の注意事項

**【対象】** 全デバッグ・ツール, RX72M グループ

RX72M グループの全製品で新規プロジェクトを作成した場合、フォルダに配置される[iodef.h]の定義がユーザーズ・マニュアルの記載と異なります。V1.00A 以前の[iodef.h]ファイルが該当します。

1. 誤 : unsigned short ACKCMDER:1;  
正 : unsigned short ACKCMDERR:1;
2. 誤 : unsigned long PSADR:19;  
正 : unsigned long PSADR:17;

## 5.2 ビルド・ツールの注意事項

### 5.2.1 ビルド・オプションのインポートに関する注意事項

インポート先および元のプロジェクトでビルド・ツールのプロパティの[I/O ヘッダ・ファイル生成オプション]タブの[ファイルに出力するモジュールを選択する]プロパティの値を"はい"にした場合、ビルド・オプションのインポートを行うと、インポートに失敗します。

インポート先のプロジェクトで[ファイルに出力するモジュールを選択する]プロパティの値を"いいえ"にしてプロジェクトを保存して、再度プロジェクトを開いた後でビルド・オプションのインポートを行ってください。

### 5.2.2 ビルド・モードに関する注意事項

[ビルド・モード設定]ダイアログでビルド・モードを複製、または名前を変更する際、既存のビルド・モードと大文字小文字のみ異なる名前でも複製または変更できます。

通常、ビルド・ツールのプロパティの 共通オプション タブ [中間ファイル出力フォルダ] プロパティやリンク・オプション タブ/ヘキサ出力オプションタブ [出力フォルダ] プロパティがビルド・モード名になっており、大文字小文字のみ異なる2つのビルド・モードの出力先が重複してしまう点、ご注意ください。

回避するには [中間ファイル出力フォルダ] プロパティ、 [出力フォルダ] プロパティで異なるフォルダを設定してください。

### 5.2.3 e<sup>2</sup> studio からのプロジェクト変換に関する注意事項【RL78】

e<sup>2</sup> studio からのプロジェクト変換において、2 番目以降のビルド・モードで [ヘキサ出力オプション] タブの設定が引き継がれないことがあります。

回避策：プロジェクト変換後、[ヘキサ出力オプション] タブを設定しなおしてください。

## 5.3 設計ツールの注意事項

### 5.3.1 プロジェクト保存に関する注意事項

サブプロジェクトが存在するプロジェクトにて、端子配置図または端子配置表パネルが開いた状態でプロジェクトの保存を行った場合に、メイン・プロジェクトの端子配置図、端子配置表が必ず表示されます。

### 5.3.2 検索・置換ダイアログの表示に関する注意事項

検索・置換ダイアログの表示位置は記憶、復元されます。マルチディスプレイ環境で一時的にサブディスプレイが表示されていない場合などに、検索・置換ダイアログが画面外となり、表示されないように見える場合があります。その場合、以下の方法で見える位置に移動することが可能です。

1. Ctrl+F 入力直後、つまり検索・置換ダイアログにフォーカスがある状態で Alt+Space キーを入力します。コントロールメニューが表示されるので、M キーを入力し「移動(M)」を選択します。
2. その状態で矢印キーを入力、またはマウスカーソルを動かすと、ダイアログが表示されます。

## 5.4 デバッグ・ツールの注意事項

文中において以下の略称を使用しています。

OCD(Serial)

： E1 エミュレータ、E20 エミュレータ、E2 エミュレータ、E2 エミュレータ Lite の Serial / FINE 接続

OCD(JTAG)

： E1 エミュレータ、E20 エミュレータ、E2 エミュレータ、E2 エミュレータ Lite の JTAG 接続

OCD(LPD)

： E1 エミュレータ、E20 エミュレータ、E2 エミュレータの LPD 接続

### 5.4.1 サブプロジェクトの追加について

【対象】全デバッグ・ツール、全デバイス

デバッグ・ツール接続中にサブプロジェクトを追加すると、ダウンロード等に失敗することがあります。サブプロジェクトの追加は、デバッグ・ツール切断中にしてください。

### 5.4.2 レジスタ割付された共用体の取り扱いに関する注意事項

【対象】全デバッグ・ツール、RX

共用体がレジスタに割り付いている場合、共用体のメンバはレジスタの下位バイトから割り付いているとみなします。このため、ビッグエンディアンの場合はメンバの値を正しく表示できません。

### 5.4.3 char 型の引数を持つ同名の関数の取り扱いに関する注意事項

【対象】全デバッグ・ツール、RX

下記のように char 型を使用した 3 つの関数を定義した場合、"Func(signed char)" のアドレスを正しく表示でき

ません。 ("Func(char)"のアドレスを表示します。)

```
void Func(char);
```

```
void Func(signed char);
```

```
void Func(unsigned char);
```

#### 5.4.4 オーバーレイ・セクションの優先セクションの変更に関する注意事項

【対象】全デバッグ・ツール, RX

オーバーレイ・セクションの優先セクションを変更しても、デバッガの機能には直ぐには反映されません。

例えば、エディタ上のアドレス表示については、ファイルを一旦閉じ、再度開くことにより反映されます。また、ウォッチ・パネル上の変数表示については、1回ステップを実行することにより反映されます。

#### 5.4.5 CC-RX コンパイラのリンク・オプションの注意事項

【対象】全デバッグ・ツール, RX

CC-RX コンパイラはリンク・オプション-sdebugには対応していません。

デバッグの際は、CC-RX コンパイラプロパティの[リンク・オプション]タブ→[出力カテゴリ]→[デバッグ情報]を出力する]-debugに設定してください。

#### 5.4.6 for 文やインライン関数内のブレークポイント設定に関する注意事項

【対象】全デバッグ・ツール, 全デバイス共通

以下のプログラムをCソースに記述するとソースプログラム1行に対する命令が複数箇所に配置されますがエディタ上にはそのうちの1箇所のアドレスのみを表示しています。

この行に対しブレークポイントを設定した場合、ブレークするのはエディタに表示したアドレス実行時のみとなります。

1. インライン関数 (\*1)
2. テンプレート関数
3. for 文, do-while 文の先頭行

\*1: 最適化によりインライン展開された関数も含まれます

#### 5.4.7 デュアルモードのプロジェクトに関する注意事項

【対象】全デバッグ・ツール, RX65N-2M, RX651-2M グループ

デュアルモードのプロジェクトの場合、1.5MB以下のROMサイズではバンク0とバンク1のROMのアドレス空間が不連続のため、本来隙間が存在しますが、以下の機能では隙間の領域のデータが存在するものとして扱います。

- ・メモリパネル
- ・ウォッチ・パネル
- ・逆アセンブル・パネル

[E1/E20/E2/E2Lite]

該当領域に対してのRead結果は0x00となり、Writeは無視されます。

[Simulator]

該当領域に対しての Read, Write とともに正常終了します。

#### 5.4.8 IE850A に関する注意事項

【対象】 IE850A

トレース・メモリ・サイズ[フレーム]プロパティで 512K より大きな値を指定できますが、指定した場合トレース・パネルの表示や Python コンソールでは最新の 512K のトレース・データを使用します。

#### 5.4.9 RL78/G1M, RL78/G1N に関する注意事項

【対象】 OCD(Serial), RL78/G1M, RL78/G1N

RL78/G1M, RL78/G1N では以下のプロパティは使用できません。

接続用設定 - ターゲット・ボードとの接続 - 低電圧 OCD ボードを使用する

#### 5.4.10 周辺シミュレーション機能に関する注意事項

【対象】 シミュレータ, RL78/G10, RL78/G11, RL78/G12, RL78/G13, RL78/G13A, RL78/G14, RL78/L12, RL78/G1F, RL78/G23

対象デバイスのユーザーズ・マニュアルで禁止されている SFR の設定を、ユーザ・プログラムまたはデバッグ操作で行った場合、CS+が正しく動作しない場合があります。

#### 5.4.11 シミュレータのスタンバイ・モード中の時間計測に関する注意事項

【対象】 シミュレータ, RL78, RH850

プログラムがスタンバイ・モード (RL78:HALT/STOP/SNOOZE, RH850:HALT) に入る場合に、次の 2 つの機能の時間は正しく計測できません。

(1) Run-Break タイマ機能 (RL78 シミュレータ, RH850 シミュレータ)

以下の場合、計測される Run-Break 時間は正しくありません。

- ・スタンバイ・モード中に強制ブレークした場合
- ・スタンバイ・モード中に強制ブレーク後のプログラム実行を開始した場合

(2) Python 関数の `debugger.Interrupt.SetTimer` (RH850 シミュレータのみ)

`debugger.Interrupt.SetTimer` で設定するブレーク時間が、スタンバイ・モード (HALT) 中にマッチしても、スタンバイ・モードが解除されるまでブレークしません。

なお、`debugger.XRunBreak.Set` も同じ機能を持つので同様です。

## 第6章 制限事項

本章では、制限事項について説明します。

### 6.1 デバッグ・ツールの制限事項

文中において、以下の略称を使用しています。

OCD(Serial)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータ, E2 エミュレータ Lite の Serial / FINE 接続

OCD(JTAG)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータ, E2 エミュレータ Lite の JTAG 接続

OCD(LPD)

: E1 エミュレータ, E20 エミュレータ, E2 エミュレータの LPD 接続

#### 6.1.1 デバッグ・ツールの制限事項一覧

| No. | 対象ツール     | 対象デバイス             | 制限事項                                      | 備考 |
|-----|-----------|--------------------|-------------------------------------------|----|
| 1   | 全デバッグ・ツール | RL78, RX,<br>RH850 | ロード・モジュール分割時の注意事項                         |    |
| 2   | 全デバッグ・ツール | RL78, RX,<br>RH850 | 変数情報表示機能の注意事項                             |    |
| 3   | 全デバッグ・ツール | RL78, RX,<br>RH850 | デバッグ対象のロード・モジュール内に同じ名称のファイルが複数存在する場合の制限事項 |    |

#### 6.1.2 デバッグ・ツールの制限事項詳細

##### No.1 ロード・モジュール分割時の注意事項

【対象】 全デバッグ・ツール, RL78, RX, RH850

【内容】 ロード・モジュールを分割生成した場合に、下記の制限があります。

- a. ソースレベルデバッグができません。
- b. 2 番目以降の出力ファイルは、デバッグ・ツールに自動登録されません。

【回避策】 ロード・モジュールは分割しないでください。

##### No.2 変数情報表示機能の注意事項

【対象】 全デバッグ・ツール, RL78, RX, RH850

【内容】 関数内で同じ名称の変数を複数定義している場合、各行で停止時に参照できる変数の情報が期待値と異なる場合があります。なお、本現象はコンパイル時の最適化レベル(注)に依存します。

注：設定項目(プロパティ)の位置



[ビルド・ツール]-[共通オプション]-[よく使うオプション(リンク)]。

【例】 以下のようなプログラムを記述した場合、生成コード上は、

(\*1)の位置で char 型の変数"a"が最も内側のスコープに生存し、

(\*2)の位置で int 型の変数"a"が最も内側のスコープに生存します。

しかし、本制限に該当する場合には、(\*1)、(\*2)のどちらの位置からも一方の変数の情報しか参照できません。

```
void main()
{
 int a = 100;
 {
 char a = 'A';
 a++; <-(*1)
 }
 a++; <-(*2)
}
```

●(\*1)の位置でのウォッチ・パネル上の表示結果

"a" 'A' (0x41) "signed char" "0xfeb1004" //期待値

または "a" 100 (0x00000064) "int" "0xfeb1000"

●(\*2)の位置でのウォッチ・パネル上の表示結果

"a" 'B' (0x42) "signed char" "0xfeb1004"

または "a" 100 (0x00000064) "int" "0xfeb1000" //期待値

【発生条件】 コンパイル時に最適化を行うと発生することがあります。

【回避策】 コンパイル時の最適化レベルを「デバッグ優先」にしてください。以下の条件を全て満たした場合、認証切れエラーが発生し、デバッグを継続することができなくなります。

### No.3 デバッグ対象のロード・モジュール内に同じ名称のファイルが複数存在する場合の制限事項

【対象】 全デバッグ・ツール, RL78, RX, RH850

【内容】 デバッグ対象のロード・モジュール内に同じ名称のファイルが複数存在する場合、エディタで行アドレスが正しく表示されません。また、イベント機能が正しく設定されません。

フォルダ構成例：

C:\Work\CS+\ProjA\ProjA.mtpj\Src\main.c -> A.abs

C:\Work\CS+\ProjB\ProjB.mtpj\Src\main.c -> B.abs

上記 2 つのロード・モジュールを同時にデバッグする場合

※上記例では複数のロード・モジュールを使用していますが、同一ロード・モジュール内にファイルが複数存在する場合も同様です。

【発生条件】 発生条件：ファイルのコンパイルディレクトリ以降の相対パス(ファイル名含む)が一致する

CS+上でビルドする場合：

コンパイルディレクトリ=プロジェクト・ファイル(\*.mtpj(注))位置

注：サブプロジェクトの場合は "\*.mtpj"ではなく、 "\*.mtsp"になります

MakeFile 等でビルドする場合：

コンパイルディレクトリ=カレントディレクトリ

【回避策】以下のいずれかを実施いただくことで、デバッグ中も同名ファイルが区別できるようになります。

- a. コンパイルディレクトリからの相対パスが異なるようにフォルダ構成を変更してください。

Before : ProjA¥Src¥main.c

          : ProjB¥Src¥main.c

After : ProjA¥SrcA¥main.c

          : ProjB¥SrcB¥main.c

→変更を実施することで相対パスはそれぞれ

"SrcA¥main.c"

"SrcB¥main.c"になります。

- b. デバッグ対象ファイルに同名ファイルがなくなるようファイル名を変更してください。

Before : ProjA¥Src¥main.c

          : ProjB¥Src¥main.c

After : ProjA¥Src¥mainA.c

          : ProjB¥Src¥mainB.c

## 6.2 設計ツールの制限事項

### 6.2.1 設計ツールの注意事項一覧

| No. | 対象ツール | 対象デバイス | 制限事項                                     | 備考 |
|-----|-------|--------|------------------------------------------|----|
| 1   | コード生成 | RH850  | アクティブ・プロジェクトを変更した場合、プロジェクト・ツリーの表示がちらつく問題 |    |

### 6.2.2 設計ツールの制限事項の詳細

No.1 アクティブ・プロジェクトを変更した場合、プロジェクト・ツリーの表示がちらつく問題

【対 象】 RH850

【内 容】プロジェクト・ツリーに2つ以上のプロジェクトが存在し、それぞれが「コード生成」を含む場合、アクティブ・プロジェクトを変更するとプロジェクト・ツリーの表示がちらつきます。

【回避策】 回避策はありません。

## 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、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。
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といたします。
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](http://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/](http://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。