

CS+ V8.05.00

統合開発環境

ユーザーズマニュアル GHS CCRH850 ビルド・ツール操作編

対象デバイス

RH850 ファミリ

本資料に記載の全ての情報は発行時点のものであり、ルネサス エレクトロニクスは、予告なしに、本資料に記載した製品または仕様を変更することがあります。
ルネサス エレクトロニクスのホームページなどにより公開される最新情報をご確認ください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
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、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。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、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。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、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。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。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。
10. 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。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
12. 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.4.0-1 2017.11)

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。

このマニュアルの使い方

このマニュアルは、RH850 ファミリ用アプリケーション・システムを開発する際の統合開発環境である CS+ について説明します。

CS+ は、RH850 ファミリの統合開発環境（ソフトウェア開発における、設計、実装、デバッグなどの各開発フェーズに必要なツールをプラットフォームである IDE に統合）です。統合することで、さまざまなツールを使い分ける必要がなく、本製品のみを使用して開発のすべてを行うことができます。

対象者 このマニュアルは、CS+ を使用してアプリケーション・システムを開発するユーザを対象としています。

目的 このマニュアルは、CS+ の持つソフトウェア機能をユーザに理解していただき、これらのデバイスを使用するシステムのハードウェア、ソフトウェア開発の参照用資料として役立つことを目的としています。

構成 このマニュアルは、大きく分けて次の内容で構成しています。

- 1. 概 説
- 2. 機 能
- A. ウィンドウ・リファレンス

読み方 このマニュアルを読むにあたっては、電気、論理回路、マイクロコンピュータに関する一般知識が必要となります。

凡例	データ表記の重み	: 左が上位桁, 右が下位桁
	アクティブ・ロウの表記	: XXX (端子, 信号名称に上線)
	注	: 本文中についた注の説明
	注意	: 気をつけて読んでいただきたい内容
	備考	: 本文中の補足説明
	数の表記	: 10 進数 ... XXXX
		16 進数 ... 0xFFFF

目次

1.	概 説	5
1.1	概 要	5
1.2	特 長	5
2.	機 能	6
2.1	概 要	6
2.1.1	ロード・モジュールを作成する	6
2.1.2	ユーザ・ライブラリを作成する	7
2.1.3	既存の GHS プロジェクト・ファイルを使用する	8
2.2	出力ファイルの種類を設定する	9
2.2.1	出力ファイル名を変更する	9
2.2.2	マップ情報を出力する	10
2.3	コンパイル・オプションを設定する	11
2.3.1	コード・サイズを優先した最適化を行う	11
2.3.2	実行速度を優先した最適化を行う	11
2.3.3	インクルード・パスを追加する	11
2.3.4	定義マクロを設定する	13
2.4	アセンブル・オプションを設定する	14
2.4.1	インクルード・パスを追加する	14
2.4.2	定義マクロを設定する	15
2.5	リンク・オプションを設定する	17
2.5.1	ユーザ・ライブラリを追加する	17
2.6	ライブラリ生成オプションを設定する	19
2.6.1	ライブラリ・ファイルの出力を設定する	19
2.7	個別にビルド・オプションを設定する	20
2.7.1	プロジェクト単位でビルド・オプションを設定する	20
2.7.2	ファイル単位でコンパイル／アセンブル・オプションを設定する	20
2.8	I/O ヘッダ・ファイルを自動更新する	22
A.	ウインドウ・リファレンス	23
A.1	説 明	23
改訂記録		C - 1

1. 概 説

この章では、GHS CCRH850 ビルド・ツール・プラグインの概要について説明します。

1.1 概 要

ビルド・ツール・プラグインでは、ロード・モジュール、またはユーザ・ライブラリを作成するためのビルド・オプションを設定することができます。

1.2 特 長

以下に、ビルド・ツール・プラグインの特長を示します。

- ビルド・オプション設定
よく使うビルド・オプションを GUI（Graphical User Interface）上で設定することができます。

2. 機 能

この章では、CS+ を使用したビルドの手順、およびビルドに関する主な機能について説明します。

2.1 概 要

ここでは、ロード・モジュール、およびユーザ・ライブラリの作成手順について説明します。

2.1.1 ロード・モジュールを作成する

ロード・モジュールの作成手順を以下に示します。

- 備考 (1), (3), (4), (8), (9) についての詳細は、「CS+ 統合開発環境 ユーザーズマニュアル プロジェクト操作編」を参照してください。
- (1) プロジェクトの作成／読み込み
プロジェクトの新規作成、または既存のプロジェクトの読み込みを行います。
新規作成する場合は、プロジェクトの種類に「空のアプリケーション (GHS CCRH850)」を指定してください。
 - (2) ツールのパスの確認
プロパティ パネルの「[共通オプション](#)」タブの「ツールのパス」カテゴリの設定を確認し、必要に応じて設定し直してください。
 - (3) ビルド対象プロジェクトの設定
ビルド対象とするプロジェクトを設定します。
 - (4) ビルド対象ファイルの設定
ビルド対象ファイルの追加／削除、依存関係の更新などを行います。
 - (5) 出力ファイルの種類の設定
生成するロード・モジュールの種類を設定します（「[2.2 出力ファイルの種類を設定する](#)」参照）。
 - (6) ビルド・オプションの設定
コンパイラ、アセンブラ、リンカなどに対するオプションを設定します（「[2.3 コンパイル・オプションを設定する](#)」, 「[2.4 アセンブル・オプションを設定する](#)」, 「[2.5 リンク・オプションを設定する](#)」など参照）。
備考 ビルド・ツールのプロパティがサポートしている GHS コンパイラのバージョンは 2020.5.5, 2020.1.5, 2019.5.5, 2019.1.5, 2018.5.5, 2018.1.5, 2017.5.5, 2017.1.5, 2016.5.5, 2015.1.7, 2015.1.5, 2014.1.7, 2013.5.5, 2013.1.5, 2012.5.5 です。
ビルド・ツールがサポートしている GHS コンパイラのオプションでも、GHS コンパイラのバージョンによっては CS+ のデバッグ・ツールでサポートしていないものがあります。詳細は「CS+ 統合開発環境 ユーザーズマニュアル RH850 デバッグ・ツール編」の「機 能」→「ダウンロード／アップロード」→「応用的なダウンロード方法」を参照してください。
 - (7) I/O ヘッダ・ファイルの更新方法の設定
デバイス・ファイルのアップデートに合わせて、I/O ヘッダ・ファイルを更新します（「[2.8 I/O ヘッダ・ファイルを自動更新する](#)」参照）。
 - (8) ビルドの実行
ビルドを実行します。
備考 ビルド処理前、およびビルド処理後に実行したいコマンドがある場合は、[プロパティ パネルの「共通オプション」](#)タブの「その他」カテゴリにおいて、「ビルド前に実行するコマンド」プロパティ、および「ビルド後に実行するコマンド」プロパティを設定してください。
ファイル単位でビルド処理前、およびビルド処理後に実行したいコマンドがある場合は、「[個別コンパイル・オプション](#)」タブ（C ソース・ファイルの場合）、および「[個別アセンブル・オプション](#)」タブ（アセンブリ・ソース・ファイルの場合）において設定することができます。
 - (9) プロジェクトの保存
プロジェクトの設定内容をプロジェクト・ファイルに保存します。

2.1.2 ユーザ・ライブラリを作成する

ユーザ・ライブラリの作成手順を以下に示します。

備考 (1), (3), (4), (6), (7) についての詳細は、「CS+ 統合開発環境 ユーザーズマニュアル プロジェクト操作編」を参照してください。

- (1) プロジェクトの作成／読み込み
プロジェクトの新規作成、または既存のプロジェクトの読み込みを行います。
プロジェクトを新規作成する際は、プロジェクトの種類に「ライブラリ (GHS CCRH850)」を指定してください。

- (2) ツールのパスの確認
プロパティ パネルの「共通オプション」タブの「ツールのパス」カテゴリの設定を確認し、必要に応じて設定し直してください。

- (3) ビルド対象プロジェクトの設定
ビルド対象とするプロジェクトを設定します。

- (4) ビルド対象ファイルの設定
ビルド対象ファイルの追加／削除、依存関係の更新などを行います。

- (5) ビルド・オプションの設定
コンパイラ、アセンブラ、ライブラリアンに対するオプションを設定します（「2.3 コンパイル・オプションを設定する」、「2.4 アセンブル・オプションを設定する」、「2.6 ライブラリ生成オプションを設定する」参照）。

備考 ビルド・ツールのプロパティがサポートしている GHS コンパイラのバージョンは 2020.5.5, 2020.1.5, 2019.5.5, 2019.1.5, 2018.5.5, 2018.1.5, 2017.5.5, 2016.5.5, 2015.1.7, 2015.1.5, 2014.1.7, 2013.5.5, 2013.1.5, 2012.5.5 です。
ビルド・ツールがサポートしている GHS コンパイラのオプションでも、GHS コンパイラのバージョンによっては CS+ のデバッグ・ツールでサポートしていないものがあります。詳細は「CS+ 統合開発環境 ユーザーズマニュアル RH850 デバッグ・ツール編」の「機能」→「ダウンロード／アップロード」→「応用的なダウンロード方法」を参照してください。

- (6) ビルドの実行
ビルドを実行します。

備考 ビルド処理前、およびビルド処理後に実行したいコマンドがある場合は、プロパティ パネルの「共通オプション」タブの「その他」カテゴリにおいて、[ビルド前に実行するコマンド] プロパティ、および [ビルド後に実行するコマンド] プロパティを設定してください。
ファイル単位でビルド処理前、およびビルド処理後に実行したいコマンドがある場合は、「個別コンパイル・オプション」タブ (C ソース・ファイルの場合)、および「個別アセンブル・オプション」タブ (アセンブリ・ソース・ファイルの場合) において設定することができます。

- (7) プロジェクトの保存
プロジェクトの設定内容をプロジェクト・ファイルに保存します。

2.1.3 既存の GHS プロジェクト・ファイルを使用する

既存の GHS プロジェクト・ファイルを使用して CS+ のプロジェクトを作成する方法について説明します。

- デバッグ専用プロジェクトと同様に、GHS コンパイラでビルドしたロード・モジュールを使用したデバッグを CS+ 上で行うことができます。
- 既存の GHS プロジェクト・ファイルを使用して簡単にビルドを行うことができます。

備考 GHS のダウンロード・モジュール・ファイルを使用する場合の注意点などは「CS+ 統合開発環境 ユーザーズマニュアル RH850 デバッグ・ツール編」の「機能」→「ダウンロード／アップロード」→「応用的なダウンロード方法」を参照してください。

既存の GHS プロジェクト・ファイルを使用した CS+ のプロジェクトの作成手順を以下に示します。

備考 (1), (4), (5), (6) についての詳細は、「CS+ 統合開発環境 ユーザーズマニュアル プロジェクト操作編」を参照してください。

- (1) プロジェクトの作成／読み込み
CS+ プロジェクトの新規作成、または GHS プロジェクト・ファイルを使用して作成した既存の CS+ プロジェクトの読み込みを行います。
新規作成する場合は、プロジェクトの種類に「既存 GHS プロジェクト・ファイル使用 (GHS CCRH850)」を指定してください。

- (2) ツールのパスの確認
プロパティ パネルの「ビルド・オプション」タブの「ツールのパス」カテゴリの設定を確認し、必要に応じて設定し直してください。

- (3) 使用する GHS プロジェクト・ファイルの設定
プロパティ パネルの「ビルド・オプション」タブの「ビルド」カテゴリにおいて、使用する GHS プロジェクト・ファイルを設定します。

注意 ビルド・オプションを設定する場合は、あらかじめ Green Hills Software, LLC の MULTI を使用して設定しておいてください。

- (4) ダウンロード・ファイルの設定
エクスプローラなどからロード・モジュール・ファイルをドラッグし、プロジェクト・ツリーのダウンロード・ファイル・ノード上でドロップしてください。

備考 詳細は「CS+ 統合開発環境 ユーザーズマニュアル プロジェクト操作編」の「外部ビルド・ツールの使用」→「プロジェクトにファイルを追加する」→「ダウンロード・ファイルを追加する」を参照してください。その際、「デバッグ専用プロジェクト」は「既存 GHS プロジェクト・ファイル使用プロジェクト」に読み替えてください。

- (5) ビルドの実行
ビルドを実行します。
- (6) プロジェクトの保存
プロジェクトの設定内容をプロジェクト・ファイルに保存します。

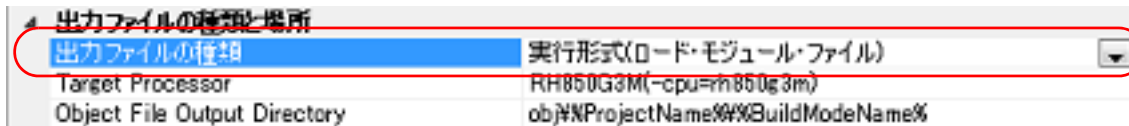
2.2 出力ファイルの種類を設定する

ビルドの生成物として出力するファイルの種類を設定します。

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**〔共通オプション〕**タブを選択します。

〔出力ファイルの種類と場所〕カテゴリの**〔出力ファイルの種類〕**プロパティにおいて、ファイルの種類を選択してください。

図 2.1 〔出力ファイルの種類〕プロパティ



- (1) 〔実行形式 (ロード・モジュール・ファイル)〕を選択した場合 (デフォルト)
ロード・モジュール・ファイルがデバッグ対象となります。

- (2) 〔実行形式 (ヘキサ・ファイル)〕を選択した場合
ヘキサ・ファイルがデバッグ対象となります。

注意 ライブラリ用のプロジェクトの場合、本プロパティは常に**〔ライブラリ形式〕**となり、変更することはできません。

2.2.1 出力ファイル名を変更する

ビルド・ツールが出力するロード・モジュール・ファイル、ヘキサ・ファイル、ライブラリ・ファイルは、デフォルトで次の名前を設定しています。

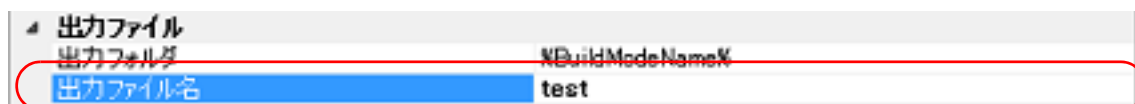
ロード・モジュール・ファイル名	: %ProjectName%
ヘキサ・ファイル名	: %ProjectName%.run
ライブラリ・ファイル名	: lib%ProjectName%.a

備考 “%ProjectName%” はプレースホルダで、プロジェクト名に置換します。

これらのファイル名の変更方法を、以下に示します。

- (1) ロード・モジュール・ファイル名を変更する場合
プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**〔リンク・オプション〕**タブを選択します。
〔出力ファイル〕カテゴリの**〔出力ファイル名〕**プロパティにおいて、変更するロード・モジュール・ファイル名を入力してください。

図 2.2 〔出力ファイル名〕プロパティ



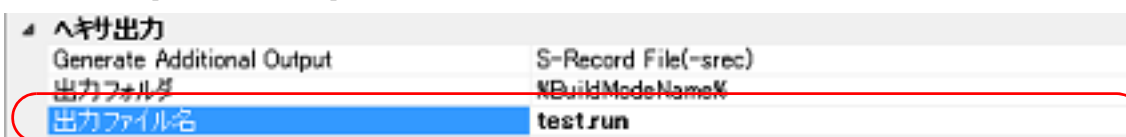
本プロパティは、次のプレースホルダに対応しています。

%ActiveProjectName%	: アクティブ・プロジェクト名に置換します。
%MainProjectName%	: メイン・プロジェクト名に置換します。
%ProjectName%	: プロジェクト名に置換します。

備考 **〔共通オプション〕**タブの**〔よく使うオプション (リンク)〕**カテゴリの**〔出力ファイル名〕**プロパティでも、同様に変更することができます。

- (2) ヘキサ・ファイル名を変更する場合
プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**〔リンク・オプション〕**タブを選択します。
〔ヘキサ出力〕カテゴリの**〔出力ファイル名〕**プロパティにおいて、変更するヘキサ・ファイル名を入力してください。

図 2.3 「出力ファイル名」 プロパティ



本プロパティは、次のプレースホルダに対応しています。

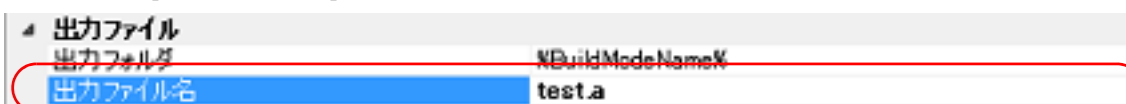
%ActiveProjectName% : アクティブ・プロジェクト名に置換します。

%MainProjectName% : メイン・プロジェクト名に置換します。

%ProjectName% : プロジェクト名に置換します。

- (3) ライブラリ・ファイル名を変更する場合
プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネルの [ライブラリ生成オプション] タブ**を選択します。
[出力ファイル] カテゴリの [出力ファイル名] プロパティにおいて、変更するライブラリ・ファイル名を入力してください。

図 2.4 「出力ファイル名」 プロパティ



本プロパティは、次のプレースホルダに対応しています。

%ActiveProjectName% : アクティブ・プロジェクト名に置換します。

%MainProjectName% : メイン・プロジェクト名に置換します。

%ProjectName% : プロジェクト名に置換します。

備考 **[共通オプション] タブ**の [よく使うオプション (ライブラリ生成)] カテゴリの [出力ファイル名] プロパティでも、同様に変更することができます。

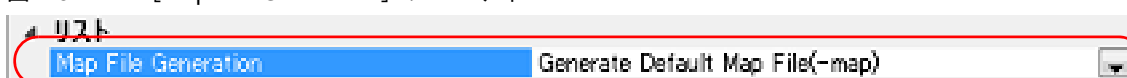
2.2.2 マップ情報を出力する

マップ情報（リンク結果の情報）は、マップ・ファイルに出力します。

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネルの [リンク・オプション] タブ**（アプリケーション・プロジェクトの場合）、または **[ライブラリ生成オプション] タブ**（ライブラリ・プロジェクトの場合）を選択します。

マップ・ファイルを出力するには、[リスト] カテゴリの [Map File Generation] プロパティで [Generate Default Map File(-map)] を選択してください。

図 2.5 「Map File Generation」 プロパティ



マップ・ファイルの出力先は、[出力フォルダ] プロパティで指定したフォルダです。
ファイル名は、リンクの出力ファイル名の拡張子を“.map”で置き換えた名前となります。

2.3 コンパイル・オプションを設定する

コンパイル・フェーズに対するオプションを設定するには、プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［コンパイル・オプション］**タブを選択してください。

タブ上で各プロパティを設定することにより、対応するコンパイル・オプションを設定することができます。

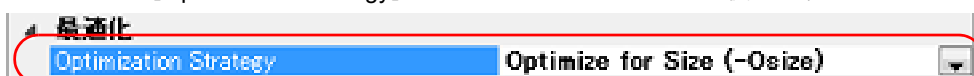
備考 よく使うオプションについては、**［共通オプション］**タブの**［よく使うオプション（コンパイル）］**カテゴリにまとめられています。

2.3.1 コード・サイズを優先した最適化を行う

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［コンパイル・オプション］**タブを選択します。

コード・サイズを優先した最適化を行うには、**［最適化］**カテゴリの**［Optimization Strategy］**プロパティで**［Optimize for Size (-Osize)］**を選択してください。

図 2.6 **［Optimization Strategy］** プロパティ（コード・サイズ優先の場合）



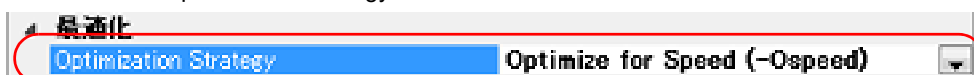
備考 **［共通オプション］**タブの**［よく使うオプション（コンパイル）］**カテゴリの**［最適化レベル］**プロパティでも、同様に設定することができます。

2.3.2 実行速度を優先した最適化を行う

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［コンパイル・オプション］**タブを選択します。

実行速度を優先した最適化を行うには、**［最適化］**カテゴリの**［Optimization Strategy］**プロパティで**［Optimize for Speed (-Ospeed)］**を選択してください。

図 2.7 **［Optimization Strategy］** プロパティ（実行速度優先の場合）



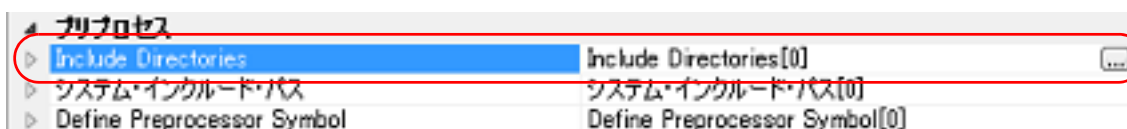
備考 **［共通オプション］**タブの**［よく使うオプション（コンパイル）］**カテゴリの**［最適化レベル］**プロパティでも、同様に設定することができます。

2.3.3 インクルード・パスを追加する

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［コンパイル・オプション］**タブを選択します。

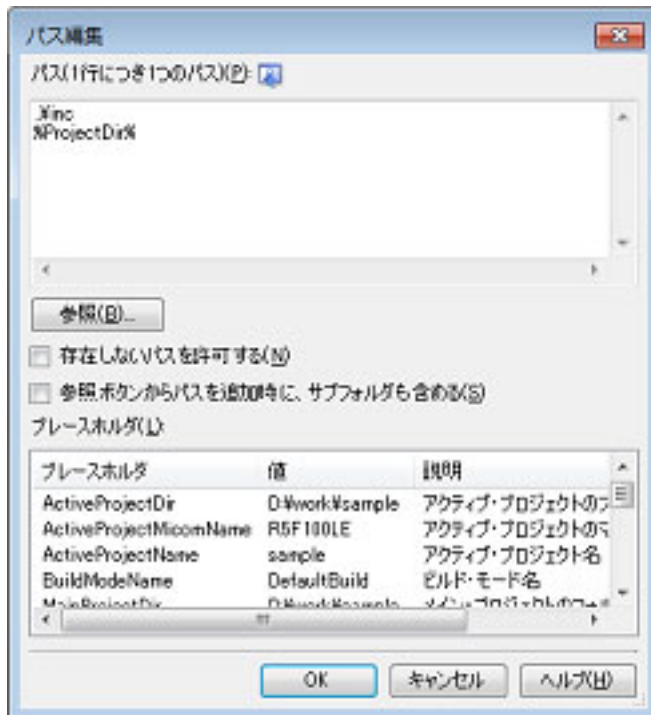
インクルード・パスの設定は、**［プリプロセッサ］**カテゴリの**［Include Directories］**プロパティで行います。

図 2.8 **［Include Directories］** プロパティ



[...] ボタンをクリックすると、パス編集 ダイアログがオープンします。

図 2.9 パス編集 ダイアログ

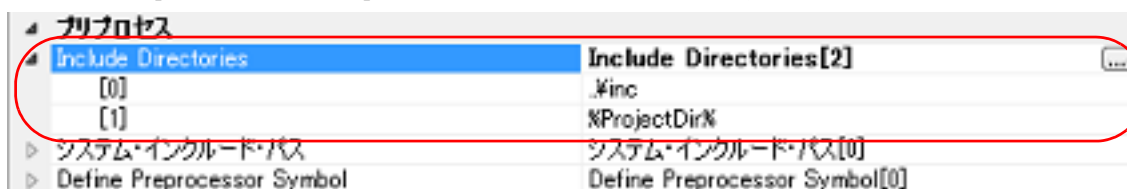


「パス (1 行につき 1 つのパス)」にインクルード・パスを 1 行に 1 つずつ入力します。
1 行に 247 文字まで、256 行まで指定可能です。

- 備考 1. 本プロパティは、プレースホルダに対応しています。
「プレースホルダ」において行をダブルクリックすると、プレースホルダが「パス (1 行につき 1 つのパス)」に反映されます。
- 備考 2. インクルード・パスは、以下のいずれかの方法で指定することも可能です。
- エクスプローラなどからフォルダをドラッグ・アンド・ドロップ
 - 「参照 ...」ボタンをクリックし、フォルダの参照 ダイアログによるフォルダの選択
 - 「プレースホルダ」において行をダブルクリック
- 備考 3. 「参照ボタンからパスを追加時に、サブフォルダも含める」をチェックしたのち、「参照 ...」ボタンからパスの指定を行うと、指定したパスとそのサブフォルダ 5 階層分までのパスを「パス (1 行につき 1 つのパス)」に追加します。

「OK」ボタンをクリックすると、入力したインクルード・パスがサブプロパティとして表示されます。

図 2.10 「Include Directories」プロパティ (インクルード・パス追加後)



インクルード・パスの変更は、「...」ボタン、またはサブプロパティのテキスト・ボックスへの直接入力により行うことができます。

また、プロジェクト・ツリーにインクルード・ファイルを追加すると、そのインクルード・パスをサブプロパティの一番最初に自動で追加します。

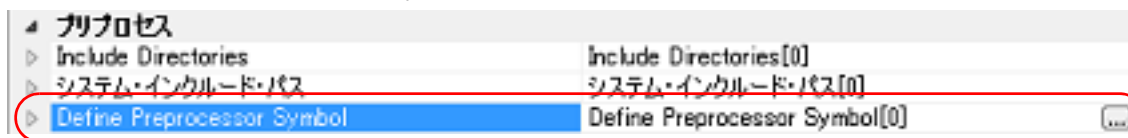
- 備考 [【共通オプション】](#) タブの「よく使うオプション (コンパイル)」カテゴリの「Include Directories」プロパティでも、同様に設定することができます。

2.3.4 定義マクロを設定する

プロジェクト・ツリーでビルド・ツール・ノードを選択し、プロパティパネルの[コンパイル・オプション]タブを選択します。

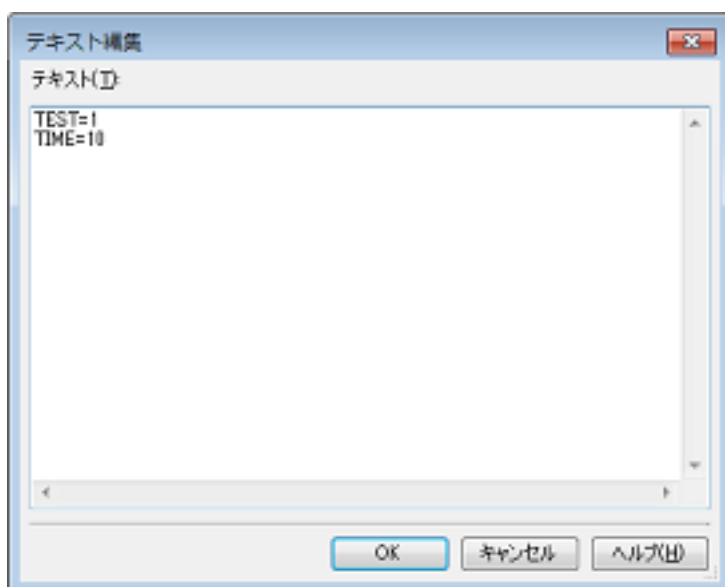
定義マクロの設定は、[プリプロセス]カテゴリの[Define Preprocessor Symbol]プロパティで行います。

図 2.11 [Define Preprocessor Symbol] プロパティ



[...] ボタンをクリックすると、テキスト編集 ダイアログがオープンします。

図 2.12 テキスト編集 ダイアログ



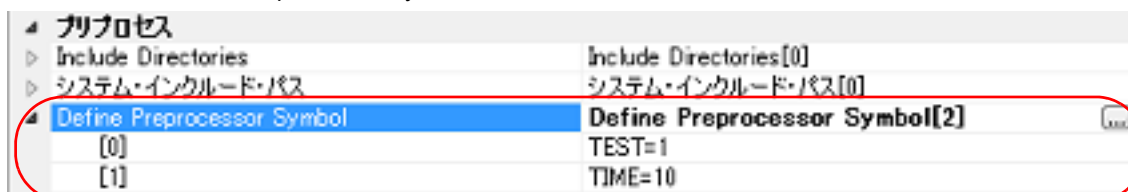
[テキスト] に定義マクロを「マクロ名 = 定義値」の形式で 1 行に 1 つずつ入力します。

1 行に 256 文字まで、256 行まで指定可能です。

「= 定義値」の部分は省略可能で、省略した場合、定義値を 1 とします。

[OK] ボタンをクリックすると、入力した定義マクロがサブプロパティとして表示されます。

図 2.13 [Define Preprocessor Symbol] プロパティ (定義マクロ設定後)



定義マクロの変更は、[...] ボタン、またはサブプロパティのテキスト・ボックスへの直接入力により行うことができます。

備考 [共通オプション] タブの [よく使うオプション (コンパイル)] カテゴリの [Define Preprocessor Symbol] プロパティでも、同様に設定することができます。

2.4 アセンブル・オプションを設定する

アセンブル・フェーズに対するオプションを設定するには、プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**「アセンブル・オプション」**タブを選択してください。

タブ上で各プロパティを設定することにより、対応するアセンブル・オプションを設定することができます。

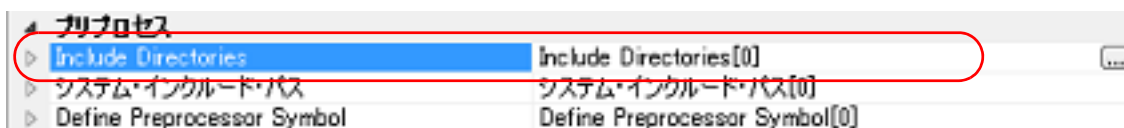
備考 よく使うオプションについては、**「共通オプション」**タブの**「よく使うオプション（アセンブル）」**カテゴリにまとめられています。

2.4.1 インクルード・パスを追加する

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**「アセンブル・オプション」**タブを選択します。

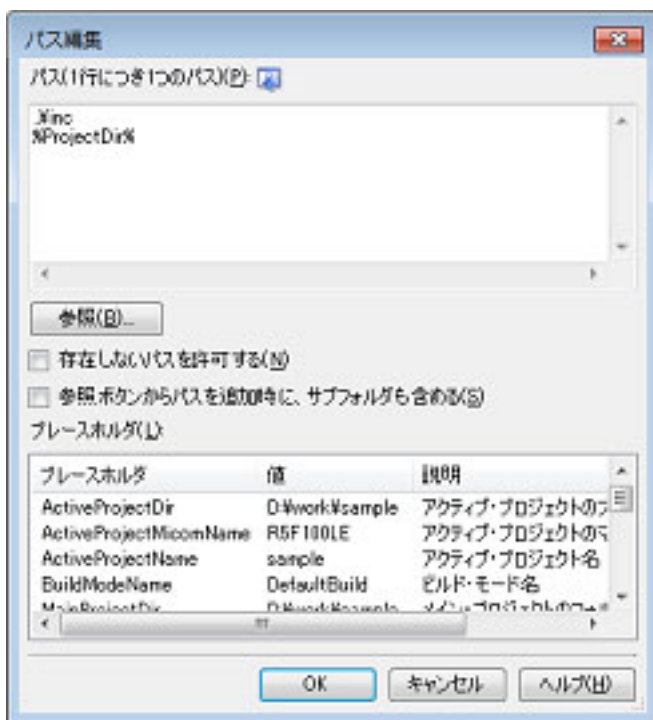
インクルード・パスの設定は、**「プリプロセッサ」**カテゴリの**「Include Directories」**プロパティで行います。

図 2.14 「Include Directories」プロパティ



[...] ボタンをクリックすると、パス編集 ダイアログがオープンします。

図 2.15 パス編集 ダイアログ



「パス (1 行につき 1 つのパス)」にインクルード・パスを 1 行に 1 つずつ入力します。

1 行に 247 文字まで、256 行まで指定可能です。

備考 1. 本プロパティは、プレースホルダに対応しています。
「プレースホルダ」において行をダブルクリックすると、プレースホルダが「パス (1 行につき 1 つのパス)」に反映されます。

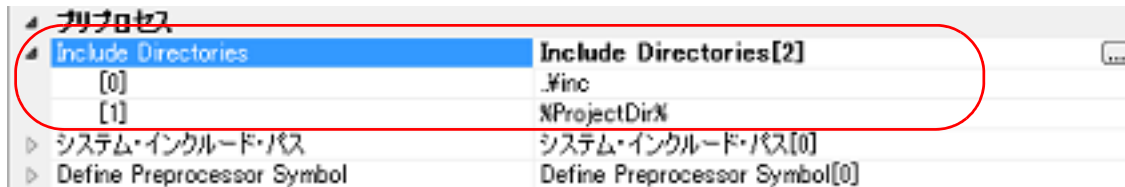
備考 2. インクルード・パスは、以下のいずれかの方法で指定することも可能です。

- エクスプローラなどからフォルダをドラッグ・アンド・ドロップ
- 「参照 ...」ボタンをクリックし、フォルダの参照 ダイアログによるフォルダの選択
- 「プレースホルダ」において行をダブルクリック

備考 3. [参照ボタンからパスを追加時に、サブフォルダも含める] をチェックしたのち、[参照 ...] ボタンからパスの指定を行うと、指定したパスとそのサブフォルダ 5 階層分までのパスを [パス (1 行につき 1 つのパス)] に追加します。

[OK] ボタンをクリックすると、入力したインクルード・パスがサブプロパティとして表示されます。

図 2.16 [Include Directories] プロパティ (インクルード・パス追加後)



インクルード・パスの変更は、[...] ボタン、またはサブプロパティのテキスト・ボックスへの直接入力により行うことができます。

また、プロジェクト・ツリーにインクルード・ファイルを追加すると、そのインクルード・パスをサブプロパティの一番最初に自動で追加します。

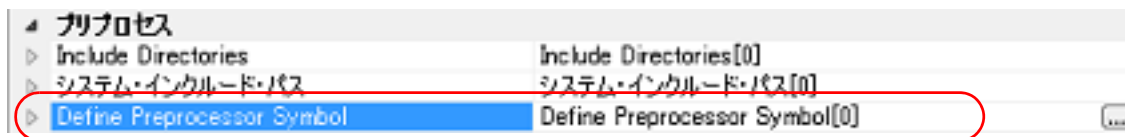
備考 [共通オプション] タブの [よく使うオプション (アセンブル)] カテゴリの [Include Directories] プロパティでも、同様に設定することができます。

2.4.2 定義マクロを設定する

プロジェクト・ツリーでビルド・ツール・ノードを選択し、プロパティ パネルの [アセンブル・オプション] タブを選択します。

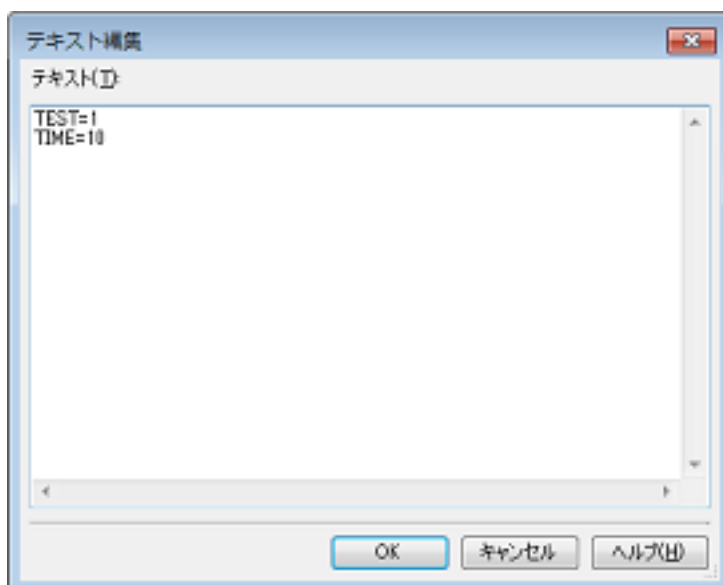
定義マクロの設定は、[プリプロセス] カテゴリの [Define Preprocessor Symbol] プロパティで行います。

図 2.17 [Define Preprocessor Symbol] プロパティ



[...] ボタンをクリックすると、テキスト編集 ダイアログがオープンします。

図 2.18 テキスト編集 ダイアログ



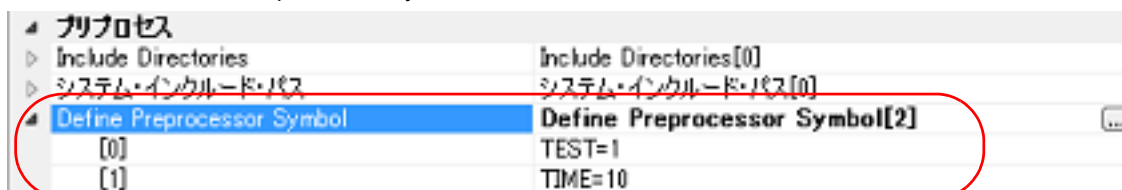
[テキスト] に定義マクロを「マクロ名 = 定義値」の形式で 1 行に 1 つずつ入力します。

1 行に 256 文字まで、256 行まで指定可能です。

「= 定義値」の部分は省略可能で、省略した場合、定義値を 1 とします。

[OK] ボタンをクリックすると、入力した定義マクロがサブプロパティとして表示されます。

図 2.19 [Define Preprocessor Symbol] プロパティ（定義マクロ設定後）



定義マクロの変更は、[...] ボタン、またはサブプロパティのテキスト・ボックスへの直接入力により行うことができます。

備考 [共通オプション] タブの [よく使うオプション (アセンブル)] カテゴリの [Define Preprocessor Symbol] プロパティでも、同様に設定することができます。

2.5 リンク・オプションを設定する

リンク・フェーズに対するオプションを設定するには、プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の **[リンク・オプション]** タブを選択してください。

タブ上で各プロパティを設定することにより、対応するリンク・オプションを設定することができます。

注意 本タブは、ライブラリ用のプロジェクトの場合は表示しません。

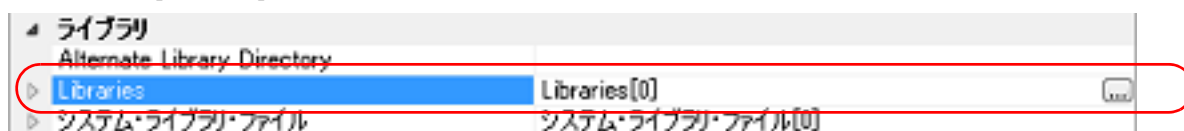
備考 よく使うオプションについては、**[共通オプション]** タブの [よく使うオプション (リンク)] カテゴリにまとめられています。

2.5.1 ユーザ・ライブラリを追加する

ユーザ・ライブラリの追加は、プロパティ、またはプロジェクト・ツリーで行います。

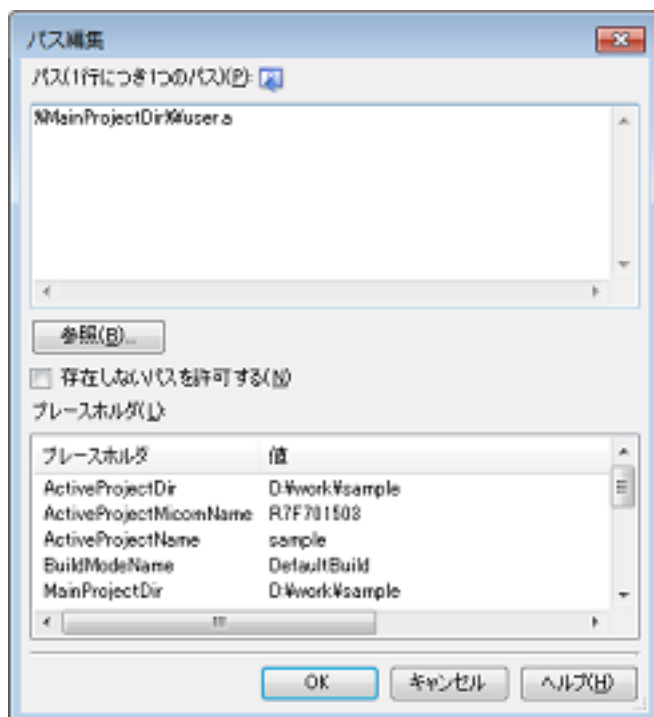
- (1) プロパティで追加する場合
プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の **[リンク・オプション]** タブを選択します。
ユーザ・ライブラリの追加は、**[ライブラリ]** カテゴリの **[Libraries]** プロパティで行います。

図 2.20 [Libraries] プロパティ



[...] ボタンをクリックすると、パス編集 ダイアログがオープンします。

図 2.21 パス編集 ダイアログ



[パス (1 行につき 1 つのパス)] にライブラリ・ファイル (パスを含む) を 1 行に 1 つずつ入力します。
1 行に 259 文字まで、65536 行まで指定可能です。

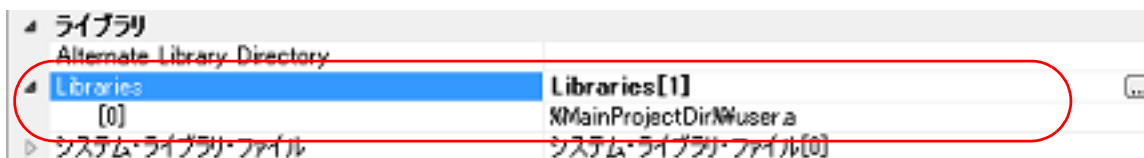
備考 1. 本プロパティは、プレースホルダに対応しています。
[プレースホルダ] において行をダブルクリックすると、プレースホルダが [パス (1 行につき 1 つのパス)] に反映されます。

備考 2. ライブラリ・ファイルは、以下のいずれかの方法で指定することも可能です。
- エクスプローラなどからフォルダをドラッグ・アンド・ドロップ

- [参照 ...] ボタンをクリックし、使用するライブラリ・ファイルを指定ダイアログによるフォルダの選択
- [プレースホルダ] において行をダブルクリック

[OK] ボタンをクリックすると、入力したライブラリ・ファイルがサブプロパティとして表示されます。

図 2.22 [Libraries] プロパティ (ライブラリ・ファイル設定後)



ライブラリ・ファイルの変更は、[...] ボタン、またはサブプロパティのテキスト・ボックスへの直接入力により行うことができます。

備考 [共通オプション] タブの [よく使うオプション (リンク)] カテゴリの [Libraries] プロパティでも、同様に設定することができます。

- (2) プロジェクト・ツリーで追加する場合
プロジェクト・ツリーへのライブラリ・ファイルの追加は、既存のファイルを追加ダイアログから行います。または、プロジェクト・ツリーにライブラリ・ファイルをドロップすることでも行うことができます。プロジェクト・ツリーで追加した場合、ビルド時にロード・モジュールとのタイムスタンプ比較の対象となり、追加されたライブラリ・ファイルが更新された時には、リンク処理が実行されます。

2.6 ライブラリ生成オプションを設定する

ライブラリ生成フェーズに対するオプションを設定するには、プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［ライブラリ生成オプション］**タブを選択してください。

タブ上で各プロパティを設定することにより、対応するライブラリ生成オプションを設定することができます。

注意 本タブは、ライブラリ用のプロジェクトの場合のみ表示します。

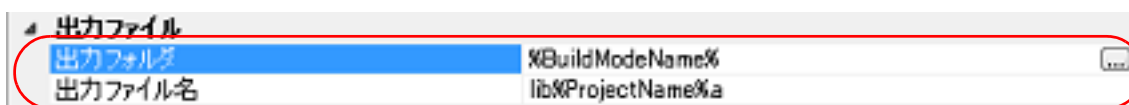
備考 よく使うオプションについては、**［共通オプション］**タブの**［よく使うオプション（ライブラリ生成）］**カテゴリにまとめられています。

2.6.1 ライブラリ・ファイルの出力を設定する

プロジェクト・ツリーでビルド・ツール・ノードを選択し、**プロパティ パネル**の**［ライブラリ生成オプション］**タブを選択します。

ライブラリ・ファイルの出力設定は、**［出力ファイル］**カテゴリで行います。

図 2.23 「出力ファイル」カテゴリ



(1) 出力フォルダの設定

「出力フォルダ」プロパティにおいて、テキスト・ボックスへの直接入力、または [...] ボタンにより行います。テキスト・ボックスには 247 文字まで指定可能です。

本プロパティは、次のプレースホルダに対応しています。

%ActiveProjectDir%	: アクティブ・プロジェクト・フォルダの絶対パスに置換します。
%ActiveProjectName%	: アクティブ・プロジェクト名に置換します。
%BuildModeName%	: ビルド・モード名に置換します。
%MainProjectDir%	: メイン・プロジェクト・フォルダの絶対パスに置換します。
%MainProjectName%	: メイン・プロジェクト名に置換します。
%MicromToolPath%	: 本製品のインストール・フォルダの絶対パスに置換します。
%ProjectDir%	: プロジェクト・フォルダの絶対パスに置換します。
%ProjectName%	: プロジェクト名に置換します。
%TempDir%	: テンポラリ・フォルダの絶対パスに置換します。
%WinDir%	: Windows システム・フォルダの絶対パスに置換します。

デフォルトでは、“%BuildModeName%”を設定しています。

(2) 出力ファイル名の設定

「出力ファイル名」プロパティにおいて、テキスト・ボックスへの直接入力により行います。

拡張子を省略した場合は、“.a”を自動的に付加します。

テキスト・ボックスには 259 文字まで指定可能です。

本プロパティは、次のプレースホルダに対応しています。

%ActiveProjectName%	: アクティブ・プロジェクト名に置換します。
%MainProjectName%	: メイン・プロジェクト名に置換します。
%ProjectName%	: プロジェクト名に置換します。

デフォルトでは、“lib%ProjectName%.a”を設定しています。

2.7 個別にビルド・オプションを設定する

ビルド・オプションの設定は、プロジェクト単位、またはファイル単位で行います。

- プロジェクト単位 → 「[2.7.1 プロジェクト単位でビルド・オプションを設定する](#)」参照
 ファイル単位 → 「[2.7.2 ファイル単位でコンパイル／アセンブル・オプションを設定する](#)」参照

2.7.1 プロジェクト単位でビルド・オプションを設定する

プロジェクト（メイン・プロジェクト、またはサブプロジェクト）に対するビルド・オプションを設定するには、プロジェクト・ツリーでビルド・ツール・ノードを選択し、[プロパティ パネル](#)を表示します。

フェーズに対する各タブを選択し、必要なプロパティを設定することにより、ビルド・オプションを設定することができます。

- | | |
|-------------------|--|
| コンパイル・フェーズ | → [コンパイル・オプション] タブ |
| アセンブル・フェーズ | → [アセンブル・オプション] タブ |
| リンク・フェーズ | → [リンク・オプション] タブ |
| ライブラリ生成フェーズ | → [ライブラリ生成オプション] タブ |
| I/O ヘッダ・ファイル生成ツール | → [I/O ヘッダ・ファイル生成オプション] タブ |

2.7.2 ファイル単位でコンパイル／アセンブル・オプションを設定する

プロジェクトに追加している各ソース・ファイルに対して、コンパイル・オプション、またはアセンブル・オプションを個別に設定することができます。

- (1) C ソース・ファイルにコンパイル・オプションを設定する場合
 プロジェクト・ツリーでC ソース・ファイルを選択し、[プロパティ パネル](#)の[\[ビルド設定\] タブ](#)を選択します。
 [ビルド] カテゴリの[\[個別コンパイル・オプションを設定する\]](#) プロパティで[はい]を選択すると、[メッセージ ダイアログ](#)がオープンします。

図 2.24 [個別コンパイル・オプションを設定する] プロパティ

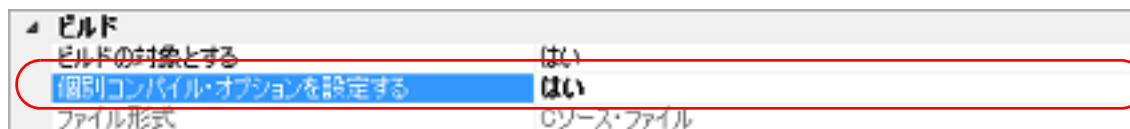
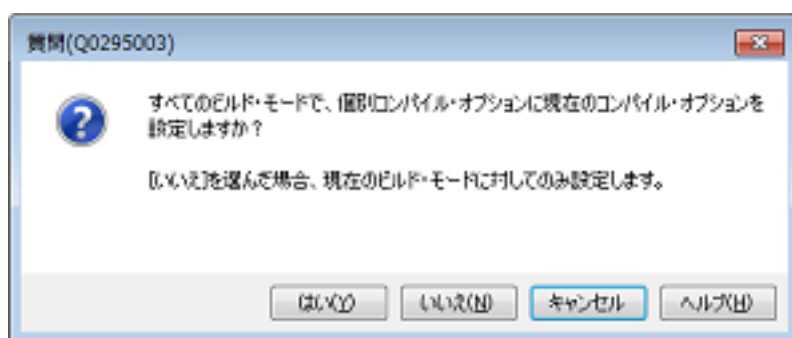


図 2.25 メッセージ ダイアログ



ダイアログ上で[はい]をクリックすると、[\[個別コンパイル・オプション\] タブ](#)を表示します。
 タブ上で必要なプロパティを設定することにより、C ソース・ファイルに対するコンパイル・オプションを設定することができます。

なお、本タブは、以下のプロパティを除いて、デフォルトでは[\[共通オプション\] タブ](#)、および[\[コンパイル・オプション\] タブ](#)の設定内容を継承します。

- [プリプロセス] カテゴリの [Include Directories]、[ビルド・ツールに指定した全体インクルード・パスも使用する]
- [出力ファイル] カテゴリの [オブジェクト・ファイル名]

- (2) アセンブリ・ソース・ファイルにアセンブル・オプションを設定する場合
プロジェクト・ツリーでアセンブリ・ソース・ファイルを選択し、プロパティパネルの[ビルド設定]タブを選択します。
[ビルド] カテゴリの[個別アセンブル・オプションを設定する] プロパティで[はい]を選択すると、メッセージダイアログがオープンします。

図 2.26 [個別アセンブル・オプションを設定する] プロパティ

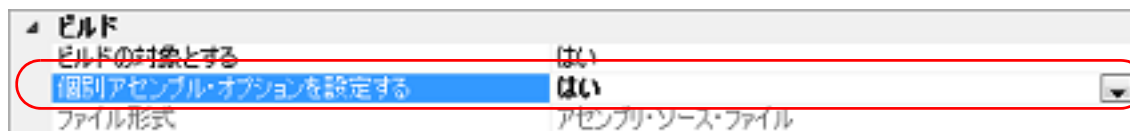
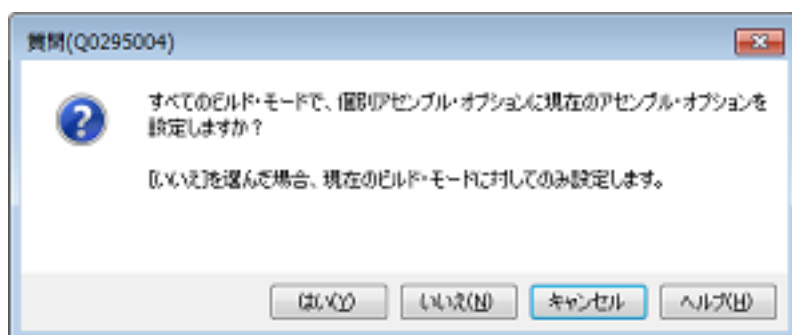


図 2.27 メッセージ ダイアログ



ダイアログ上で[はい]をクリックすると、[個別アセンブル・オプション]タブを表示します。
タブ上で必要なプロパティを設定することにより、アセンブリ・ソース・ファイルに対するアセンブル・オプションを設定することができます。

なお、本タブは、以下のプロパティを除いて、デフォルトでは[共通オプション]タブ、および[アセンブル・オプション]タブの設定内容を継承します。

- [プリプロセス] カテゴリの [Include Directories]、[ビルド・ツールに指定した全体インクルード・パスも使用する]
- [出力ファイル] カテゴリの [オブジェクト・ファイル名]

2.8 I/O ヘッダ・ファイルを自動更新する

空のアプリケーション・プロジェクト新規作成時に、選択したデバイスに対応する I/O ヘッダ・ファイルが自動で生成されます。

デバイス・ファイルのアップデートなどに対し、自動で I/O ヘッダ・ファイルを更新する必要がある場合は、以下の方法で更新してください。

I/O ヘッダ・ファイルは、空のアプリケーション・プロジェクト新規作成時に "iodefine.h" として自動で生成されます。

備考 I/O ヘッダ・ファイルはプロジェクト・ファイルと同じフォルダに生成されます。すでに同名ファイルが存在していた場合、既存のファイルを "iodefine.bak" にリネームしてバックアップします。

I/O ヘッダ・ファイルを更新するタイミング、および更新方法を以下に示します。

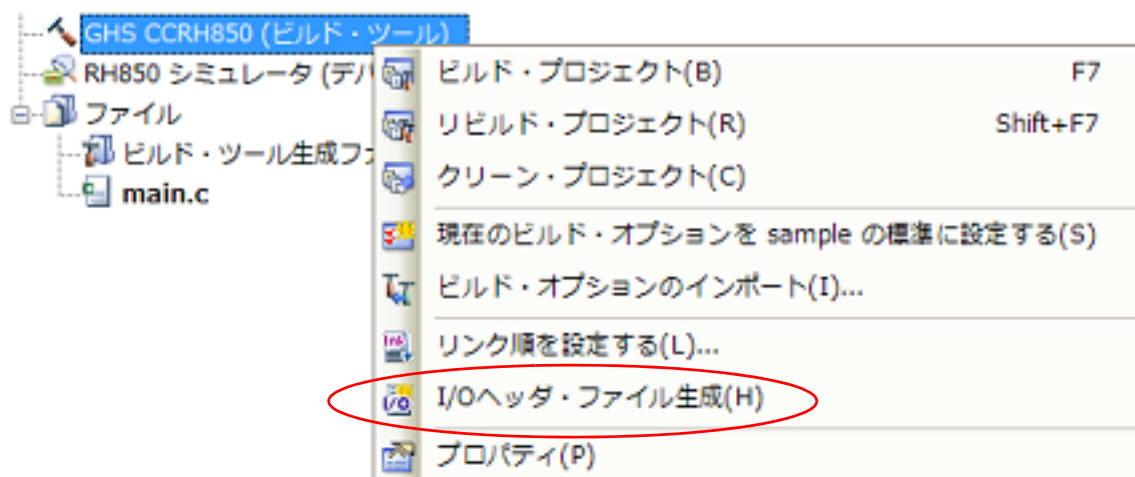
- プロジェクトのオープン時

CS+ は、プロジェクトのオープン時にデバイス・ファイルのバージョンをチェックします。

デバイス・ファイルが更新されており、I/O ヘッダ・ファイルの更新が必要な可能性がある場合、出力パネルにメッセージが表示されます。必要に応じて、以下の方法で I/O ヘッダ・ファイルを更新してください。

- プロジェクト・ツリーパネルにおいて、ビルド・ツール・ノードを選択したのち、コンテキスト・メニュー → [I/O ヘッダ・ファイル生成] を選択

図 2.28 [I/O ヘッダ・ファイル生成] 項目

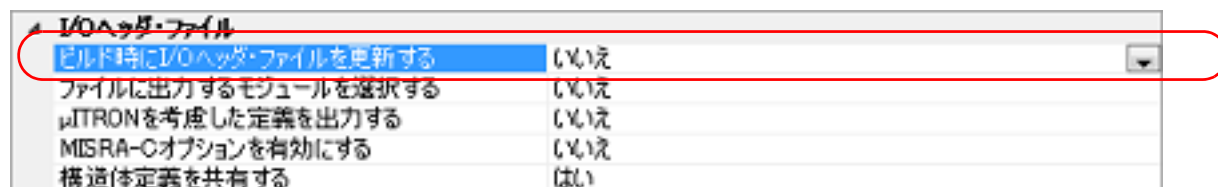


- ビルド時

ビルド操作を行ったとき、およびビルドの直前にも自動で I/O ヘッダ・ファイルを更新することができます。

プロパティパネルの [I/O ヘッダ・ファイル生成オプション] タブの [ビルド時に I/O ヘッダ・ファイルを更新する] プロパティを設定してください。更新する条件も同じカテゴリのプロパティにて変更することができます。

図 2.29 [ビルド時に I/O ヘッダ・ファイルを更新する] プロパティ



A. ウィンドウ・リファレンス

ここでは、ビルド・ツールに関するパネル／ダイアログについての詳細を説明します。

A.1 説 明

以下に、ビルド・ツールに関するパネル／ダイアログの一覧を示します。

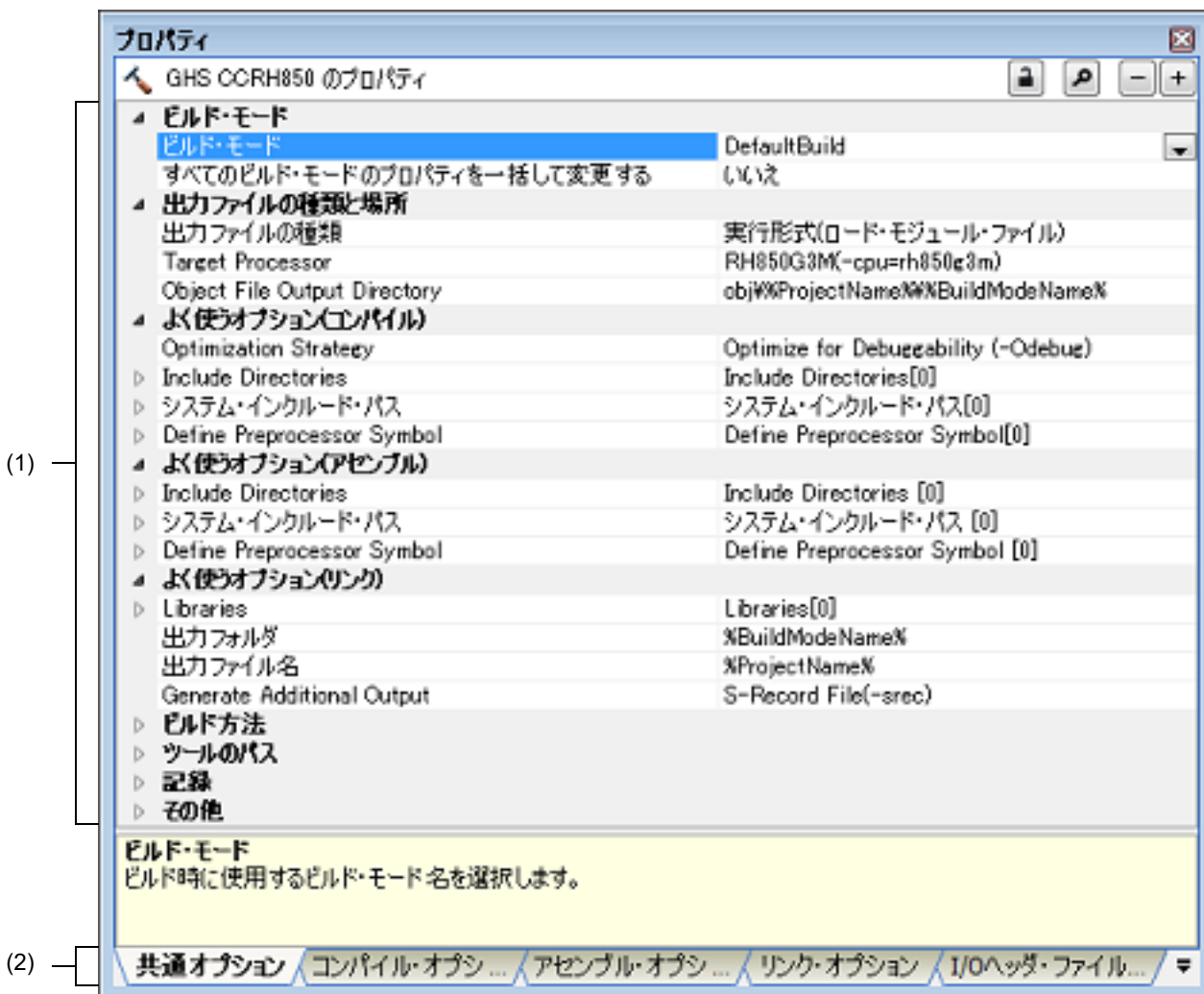
表 A.1 パネル／ダイアログ一覧

パネル／ダイアログ名	機能概要
プロパティ パネル	プロジェクト・ツリー パネルで選択しているビルド・ツール・ノード、ファイルについて、詳細情報を表示、および設定を変更
システム・インクルード・パス順設定 ダイアログ	コンパイラに対して指定するシステム・インクルード・パスを参照、および指定順を設定
ファイルに出力するモジュールの選択 ダイアログ	I/O ヘッダ・ファイルに出力するモジュールを設定

プロパティ パネル

プロジェクト・ツリー パネルで選択しているビルド・ツール・ノード、ファイルについて、カテゴリ別に詳細情報の表示、および設定の変更を行います。

図 A.1 プロパティ パネル



ここでは、次の項目について説明します。

- [オープン方法]
- [各エリアの説明]
- [[編集] メニュー (プロパティ パネル専用部分)]
- [コンテキスト・メニュー]

[オープン方法]

- プロジェクト・ツリー パネルにおいて、ビルド・ツール・ノード、ファイルを選択したのち、[表示] メニュー→[プロパティ] を選択、またはコンテキスト・メニュー→[プロパティ] を選択

備考 すでにプロパティ パネルがオープンしている場合、プロジェクト・ツリー パネルにおいて、ビルド・ツール・ノード、ファイルを選択することで、選択した項目の詳細情報を表示します。

[各エリアの説明]

(1) 詳細情報表示／変更エリア

プロジェクト・ツリー パネルで選択しているビルド・ツール・ノード、ファイルの詳細情報を、カテゴリ別のリスト形式で表示し、設定の変更を直接行うことができるエリアです。

■マークは、そのカテゴリ内に含まれているすべての項目を展開表示していることを示し、また、▶マークは、カテゴリ内の項目を折りたたみ表示していることを示します。展開／折りたたみ表示の切り替えは、このマークのクリック、またはカテゴリ名のダブルクリックにより行うことができます。

HEXマークは、そのプロパティのテキスト・ボックスが 16 進数入力専用であることを示します。

カテゴリ、およびそれに含まれる項目の表示内容／設定方法についての詳細は、該当するタブの項を参照してください。

(2) タブ選択エリア

タブを選択することにより、詳細情報を表示するカテゴリが切り替わります。

本パネルには、次のタブが存在します（各タブ上における表示内容／設定方法についての詳細は、該当するタブの項を参照してください）。

備考 プロジェクト・ツリー パネルで複数の構成要素を選択している場合は、その構成要素に共通するタブのみ表示します。

プロパティの値の変更は、選択している複数の構成要素に共通に反映します。

- (a) プロジェクト・ツリー パネルでビルド・ツール・ノードを選択している場合、かつプロジェクトの種類が「空のアプリケーション (GHS CCRH850)」または「ライブラリ (GHS CCRH850)」の場合

- [共通オプション] タブ
- [コンパイル・オプション] タブ
- [アセンブル・オプション] タブ
- [リンク・オプション] タブ
- [ライブラリ生成オプション] タブ
- [I/O ヘッダ・ファイル生成オプション] タブ

- (b) プロジェクト・ツリー パネルでビルド・ツール・ノードを選択している場合、かつプロジェクトの種類が「既存 GHS プロジェクト・ファイル使用 (GHS CCRH850)」の場合

- [ビルド・オプション] タブ

- (c) プロジェクト・ツリー パネルでファイルを選択している場合

- [ビルド設定] タブ (C ソース・ファイル、アセンブリ・ソース・ファイル、オブジェクト・ファイル、ライブラリ・ファイルの場合)
- [個別コンパイル・オプション] タブ (C ソース・ファイルの場合)
- [個別アセンブル・オプション] タブ (アセンブリ・ソース・ファイルの場合)
- [ファイル情報] タブ^注

注 [ファイル情報] タブについての詳細は、「CS+ 統合開発環境 ユーザーズマニュアル プロジェクト操作編」を参照してください。

[[編集] メニュー (プロパティ パネル専用部分)]

元に戻す	直前に行ったプロパティの値の編集作業を取り消します。
切り取り	プロパティの値を編集中の場合、選択している文字列を切り取ってクリップ・ボードに移動します。
コピー	選択しているプロパティの値文字列をクリップ・ボードにコピーします。
貼り付け	プロパティの値を編集中の場合、クリップ・ボードの内容を挿入します。
削除	プロパティの値を編集中の場合、選択している文字列を削除します。
すべて選択	プロパティの値を編集中の場合、選択しているプロパティの値文字列をすべて選択します。

[コンテキスト・メニュー]

元に戻す	直前に行ったプロパティの値の編集作業を取り消します。
切り取り	プロパティの値を編集中の場合、選択している文字列を切り取ってクリップ・ボードに移動します。
コピー	選択しているプロパティの値文字列をクリップ・ボードにコピーします。
貼り付け	プロパティの値を編集中の場合、クリップ・ボードの内容を挿入します。
削除	プロパティの値を編集中の場合、選択している文字列を削除します。
すべて選択	プロパティの値を編集中の場合、選択しているプロパティの値文字列をすべて選択します。
デフォルトに戻す	選択している項目の設定値をプロジェクトに設定しているデフォルト値に戻します。 ただし、 [個別コンパイル・オプション] タブ 、 [個別アセンブル・オプション] タブ においては、全体オプションの設定値に戻します。
すべてデフォルトに戻す	現在表示しているタブの設定値をすべてプロジェクトに設定しているデフォルト値に戻します。 ただし、 [個別コンパイル・オプション] タブ 、 [個別アセンブル・オプション] タブ においては、全体オプションの設定値に戻します。

〔共通オプション〕 タブ

本タブでは、ビルド・ツールに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [ビルド・モード]
- (2) [出力ファイルの種類と場所]
- (3) [よく使うオプション (コンパイル)]
- (4) [よく使うオプション (アセンブル)]
- (5) [よく使うオプション (リンク)]
- (6) [よく使うオプション (ライブラリ生成)]
- (7) [ビルド方法]
- (8) [ツールのパス]
- (9) [記録]
- (10) [その他]

備考 1. [よく使うオプション] カテゴリのプロパティを変更した場合、それらに対応するタブの同名のプロパティの値も連動して変更します。

〔共通オプション〕 タブのカテゴリ	対応するタブ
[よく使うオプション (コンパイル)] カテゴリ	[コンパイル・オプション] タブ
[よく使うオプション (アセンブル)] カテゴリ	[アセンブル・オプション] タブ
[よく使うオプション (リンク)] カテゴリ	[リンク・オプション] タブ
[よく使うオプション (ライブラリ生成)] カテゴリ	[ライブラリ生成オプション] タブ

備考 2. コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 1. 本タブはプロジェクトの種類に [空のアプリケーション (GHS CCRH850)]、または [ライブラリ (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

注意 2. GHS コンパイラをバージョンアップした場合は、[ツールのパス] カテゴリの [コンパイラ・パッケージのフォルダ] プロパティの設定を確認してください。

〔各カテゴリの説明〕

- (1) [ビルド・モード]
ビルド・モードに関する詳細情報の表示、および設定の変更を行います。

ビルド・モード	ビルド時に使用するビルド・モードを選択します。 なお、本プロパティには、コンテキスト・メニュー→ [すべてデフォルトに戻す] は適用されません。		
	デフォルト	DefaultBuild	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	DefaultBuild	プロジェクトの新規作成時にデフォルトで設定するビルド・モードでビルドを行います。
		プロジェクトに登録しているビルド・モード	プロジェクトに登録しているビルド・モード (DefaultBuild 以外) でビルドを行います。

すべてのビルド・モードのプロパティを一括して変更する	プロパティに値を設定する際、設定した値をすべてのビルド・モードに反映するかどうかを選択します。 設定した値が他のビルド・モードの値として適していない可能性もあるため、充分ご注意ください。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	プロパティに値を設定する際、設定した値をすべてのビルド・モードに反映します。
		いいえ	プロパティに値を設定する際、設定した値を他のビルド・モードに反映しません。

(2) [出力ファイルの種類と場所]

出力ファイルの種類と場所に関する詳細情報の表示、および設定の変更を行います。

出力ファイルの種類	ライブラリ用のプロジェクト以外の場合は、ここで設定したファイルの種類がデバッグ対象となります。 なお、ライブラリ用のプロジェクト以外の場合は[実行形式（ロード・モジュール・ファイル）]、[実行形式（ヘキサ・ファイル）]のみを表示します。 ライブラリ用のプロジェクトの場合は[ライブラリ形式]のみを表示します。		
	デフォルト	- ライブラリ用のプロジェクト以外の場合 実行形式（ロード・モジュール・ファイル） - ライブラリ用のプロジェクトの場合 ライブラリ形式	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	実行形式（ロード・モジュール・ファイル）	ビルド時にロード・モジュール・ファイルを生成します。 ロード・モジュール・ファイルがデバッグ対象となります。
		実行形式（ヘキサ・ファイル）	ビルド時にヘキサ・ファイルを生成します。 ヘキサ・ファイルがデバッグ対象となります。 なお、本項目は、 [リンク・オプション] タブ の [ヘキサ出力] カテゴリの[Generate Additional Output] プロパティで[指定しない(オプション指定なし)]以外を選択した場合のみ表示します。
		ライブラリ形式	ビルド時にライブラリ・ファイルを生成します。

Target Processor	コンパイラ・ドライバの -cpu オプションを指定します。		
	デフォルト	RH850G3M(-cpu=rh850g3m)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	RH850G3M(-cpu=rh850g3m)	-cpu=rh850g3m オプションを指定します。
		RH850G3K(-cpu=rh850g3k)	-cpu=rh850g3k オプションを指定します。
		RH850G3KH(-cpu=rh850g3kh)	-cpu=rh850g3kh オプションを指定します。
		RH850G3MH(-cpu=rh850g3mh)	-cpu=rh850g3mh オプションを指定します。
		RH850G4MH(-cpu=rh850g4mh)	-cpu=rh850g4mh オプションを指定します。
指定しない (オプション指定なし)		-cpu オプションを指定しません。	
Object File Output Directory	中間ファイルを出力するフォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 空欄の場合は、プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -object_dir オプション, または -o オプションに相当します。		
	デフォルト	obj¥%ProjectName%¥%BuildModeName%	
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, フォルダの参照 ダイアログによる編集	
	指定可能値	247 文字までの文字列	

(3) [よく使うオプション (コンパイル)]

コンパイル時によく使うオプションに関する詳細情報の表示、および設定の変更を行います。

Optimization Strategy	コンパイラ・ドライバの -O オプションを指定します。		
	デフォルト	Optimize for Debuggability (-Odebug)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Optimize for Debuggability (-Odebug)	-Odebug オプションを指定します。
		Optimize for General Use (-O)	-O オプションを指定します。
		Optimize for Size (-Osize)	-Osize オプションを指定します。
		Optimize for Speed (-Ospeed)	-Ospeed オプションを指定します。
		No Optimizations (-Onone)	-Onone オプションを指定します。
指定しない (オプション指定なし)		Optimization Strategy を指定しません。	

Include Directories	コンパイラ・ドライバの-Iオプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字, 小文字の区別はありません。	
	デフォルト	Include Directories[定義数]
	変更方法	[...] ボタンをクリックし, パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	247 文字までの文字列 256 個まで指定可能です。
システム・インクルード・パス	コンパイル時に CS+ が設定するインクルード・パスの指定順を変更します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 システム・インクルード・パスは, 追加のインクルード・パスより低い優先度で検索します。 パスはプロジェクト・フォルダを基点とします。 コンパイラ・ドライバの-Iオプションに相当します。 インクルード・パスはサブプロパティとして表示します。	
	デフォルト	システム・インクルード・パス [定義数]
	変更方法	[...] ボタンをクリックし, システム・インクルード・パス順設定 ダイアログによる編集
	指定可能値	変更不可 (インクルード・パスの設定順の変更のみ可能)

Define Preprocessor Symbol	コンパイラ・ドライバの -D オプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。	
	デフォルト	Define Preprocessor Symbol[定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

(4) [よく使うオプション (アセンブル)]

アセンブル時によく使うオプションに関する詳細情報の表示、および設定の変更を行います。

Include Directories	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字、小文字の区別はありません。	
	デフォルト	Include Directories[定義数]
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	247 文字までの文字列 256 個まで指定可能です。

システム・インクルード・パス	アセンブル時に CS+ が設定するインクルード・パスの指定順を変更します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 システム・インクルード・パスは、追加のインクルード・パスより低い優先度で検索します。 パスはプロジェクト・フォルダを基点とします。 コンパイラ・ドライバの -I オプションに相当します。 インクルード・パスはサブプロパティとして表示します。	
	デフォルト	システム・インクルード・パス [定義数]
	変更方法	[...] ボタンをクリックし、 システム・インクルード・パス順設定 ダイアログ による編集
	指定可能値	変更不可（インクルード・パスの設定順の変更のみ可能）
Define Preprocessor Symbol	コンパイラ・ドライバの -D オプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。	
	デフォルト	Define Preprocessor Symbol [定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

(5) [よく使うオプション（リンク）]

リンク時によく使うオプションに関する詳細情報の表示、および設定の変更を行います。

なお、本カテゴリは、ライブラリ用のプロジェクトの場合は表示しません。

Libraries	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 ライブラリ・ファイル名はサブプロパティとして表示します。 <table border="1"> <tr> <td>デフォルト</td><td>Libraries[定義数]</td></tr> <tr> <td>変更方法</td><td>[...] ボタンをクリックし、パス編集 ダイアログによる編集 → [参照] ボタンをクリックし、使用するライブラリ・ファイルを指定 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能</td></tr> <tr> <td>指定可能値</td><td>259 文字までの文字列 65536 個まで指定可能です。</td></tr> </table>	デフォルト	Libraries[定義数]	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 → [参照] ボタンをクリックし、使用するライブラリ・ファイルを指定 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	指定可能値	259 文字までの文字列 65536 個まで指定可能です。
デフォルト	Libraries[定義数]						
変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 → [参照] ボタンをクリックし、使用するライブラリ・ファイルを指定 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能						
指定可能値	259 文字までの文字列 65536 個まで指定可能です。						
出力フォルダ	出力フォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 空欄の場合は、プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -o オプションに相当します。 <table border="1"> <tr> <td>デフォルト</td><td>%BuildModeName%</td></tr> <tr> <td>変更方法</td><td>テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集</td></tr> <tr> <td>指定可能値</td><td>247 文字までの文字列</td></tr> </table>	デフォルト	%BuildModeName%	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集	指定可能値	247 文字までの文字列
デフォルト	%BuildModeName%						
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集						
指定可能値	247 文字までの文字列						

出力ファイル名	出力ファイル名を指定します。 次のプレースホルダに対応しています。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %ProjectName% : プロジェクト名に置換します。 コンパイラ・ドライバの -o オプションに相当します。		
	デフォルト	%ProjectName%	
	変更方法	テキスト・ボックスによる直接入力	
	指定可能値	259 文字までの文字列	
Generate Additional Output	コンパイラ・ドライバの -srec オプション, または -hex オプションを指定します。		
	デフォルト	S-Record File(-srec)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	S-Record File(-srec)	-srec オプションを指定します。
		HEX386 File(-hex)	-hex オプションを指定します。
		指定しない (オプション指定なし)	Generate Additional Output を指定しません。

(6) [よく使うオプション (ライブラリ生成)]

ライブラリ生成時によく使うオプションに関する詳細情報の表示, および設定の変更を行います。
なお, 本カテゴリは, ライブラリ用のプロジェクトの場合のみ表示します。

出力フォルダ	出力フォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 空欄の場合は, プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -o オプションに相当します。		
	デフォルト	%BuildModeName%	
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, フォルダの参照 ダイアログによる編集	
	指定可能値	247 文字までの文字列	

出力ファイル名	出力ファイル名を指定します。 次のプレースホルダに対応しています。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %ProjectName% : プロジェクト名に置換します。 コンパイラ・ドライバの -o オプションに相当します。	
	デフォルト	%ProjectName%
	変更方法	テキスト・ボックスによる直接入力
	指定可能値	259 文字までの文字列

(7) [ビルド方法]

ビルド方法に関する詳細情報の表示、および設定の変更を行います。

インクルード・ファイルが見つからないソースの扱い	ソース・ファイルがインクルードしているファイルが標準のインクルード・パス、および [Include Directories] プロパティで見つからない場合、そのソース・ファイルを再コンパイル／アセンブルするかどうかを選択します。		
	デフォルト	再コンパイル／アセンブルする	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	再コンパイル／アセンブルする	インクルード・ファイルが見つからない場合、ソース・ファイルを再コンパイル／アセンブルします。
		再コンパイル／アセンブルしない	インクルード・ファイルが見つからない場合でも、ソース・ファイル再コンパイル／アセンブルしません。

(8) [ツールのパス]

ツールのパスに関する詳細情報の表示、および設定の変更を行います。

コンパイラ・パッケージのフォルダ	使用するコンパイラ・パッケージのフォルダを指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	インストール環境に応じた値
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, フォルダの参照 ダイアログによる編集
	指定可能値	247 文字までの文字列
コンパイラ実行ファイル名	コンパイラの実行ファイル名を指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	ccrh850.exe
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, 文字列入力 ダイアログによる編集
	指定可能値	259 文字までの文字列
アセンブラ実行ファイル名	アセンブラの実行ファイル名を指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	ccrh850.exe
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, 文字列入力 ダイアログによる編集
	指定可能値	259 文字までの文字列

リンカ実行ファイル名	リンカの実行ファイル名を指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	ccrh850.exe
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, 文字列入力 ダイアログによる編集
	指定可能値	259 文字までの文字列

(9) [記録]

記録に関する詳細情報の表示, および設定の変更を行います。

メモ	このビルド・ツールにメモを追加します。 1 行に 1 項目ずつ指定します。 この設定はすべてのビルド・モードで共通です。 追加したメモはサブプロパティとして表示します。	
	デフォルト	メモ [項目数]
	変更方法	[...] ボタンをクリックし, テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

(10) [その他]

ビルド・ツールに関するその他の詳細情報の表示, および設定の変更を行います。

出力メッセージ・フォーマット	ビルド中のメッセージのフォーマットを指定します。 対象となるのは, 使用するビルド・ツール, およびプラグインによって追加されたコマンドの出力メッセージです。 [ビルド前に実行するコマンド] プロパティ, および [ビルド後に実行するコマンド] プロパティなどで指定したコマンドの出力メッセージは対象外です。 次のプレースホルダに対応しています。 %Program% : 実行中のプログラム名に置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %TargetFiles% : コンパイル／アセンブル中のファイル名, またはリンク後の出力ファイル名に置換します。 空欄の場合は, %Program% %Options% を自動的に設定します。		
	デフォルト	%TargetFiles%	
	変更方法	テキスト・ボックスによる直接入力 (256 文字までの文字列), またはドロップダウン・リストによる選択	
	指定可能値	%TargetFiles%	出力メッセージにファイル名を表示します。
		%TargetFiles%: %Options%	出力メッセージにファイル名とコマンド・ライン・オプションを表示します。
		%Program% %Options%	出力メッセージにプログラム名とコマンド・ライン・オプションを表示します。

ビルド・オプション一覧表示フォーマット	ビルド・オプション一覧の表示フォーマットを指定します。 対象となるのは、使用するビルド・ツール、およびプラグインによって追加されたコマンドのオプションです。 [ビルド前に実行するコマンド] プロパティ、および [ビルド後に実行するコマンド] プロパティなどで指定したコマンドのオプションは対象外です。 次のプレースホルダに対応しています。 %Program% : 実行中のプログラム名に置換します。 %Options% : ビルド時のコマンド・ライン・オプションに置換します。 %TargetFiles% : コンパイル／アセンブル中のファイル名、またはリンク後の出力ファイル名に置換します。 空欄の場合は、%TargetFiles% : %Program% %Options% を自動的に設定します。 <table border="1"> <tr> <td>デフォルト</td><td>%TargetFiles% : %Program% %Options%</td></tr> <tr> <td>変更方法</td><td>テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力ダイアログによる編集</td></tr> <tr> <td>指定可能値</td><td>256 文字までの文字列</td></tr> </table>	デフォルト	%TargetFiles% : %Program% %Options%	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力ダイアログによる編集	指定可能値	256 文字までの文字列
デフォルト	%TargetFiles% : %Program% %Options%						
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力ダイアログによる編集						
指定可能値	256 文字までの文字列						
ビルド前に実行するコマンド	ビルド処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、ビルド処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。 <table border="1"> <tr> <td>デフォルト</td><td>ビルド前に実行するコマンド [定義数]</td></tr> <tr> <td>変更方法</td><td>[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能</td></tr> <tr> <td>指定可能値</td><td>1023 文字までの文字列 64 個まで指定可能です。</td></tr> </table>	デフォルト	ビルド前に実行するコマンド [定義数]	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	指定可能値	1023 文字までの文字列 64 個まで指定可能です。
デフォルト	ビルド前に実行するコマンド [定義数]						
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能						
指定可能値	1023 文字までの文字列 64 個まで指定可能です。						

ビルド後に実行するコマンド	ビルド処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、ビルド処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。	
	デフォルト	ビルド後に実行するコマンド [定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	1023 文字までの文字列 64 個まで指定可能です。
その他の追加オプション	その他に追加するオプションを入力します。 なお、ここで設定したオプションは、コンパイラ・ドライバのオプション群の最後に付加します。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集
	指定可能値	32767 文字までの文字列

[コンパイル・オプション] タブ

本タブでは、コンパイル・フェーズに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [デバッグ情報]
- (2) [最適化]
- (3) [プリプロセス]
- (4) [出力コード]
- (5) [その他]

備考 コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 本タブはプロジェクトの種類に [空のアプリケーション (GHS CCRH850)]、または [ライブラリ (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

[各カテゴリの説明]

- (1) [デバッグ情報]
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	MULTI(-G)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。
Generate MULTI and Native Information	コンパイラ・ドライバの -dwarf2 オプションを指定します。		
	デフォルト	On(-dwarf2)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	On(-dwarf2)	-dwarf2 オプションを指定します。
		指定しない (オプション指定なし)	-dwarf2 オプションを指定しません。

- (2) [最適化]
最適化に関する詳細情報の表示、および設定の変更を行います。

Optimization Strategy	コンパイラ・ドライバの -O オプションを指定します。		
	デフォルト	Optimize for Debuggability (-Odebug)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Optimize for Debuggability (-Odebug)	-Odebug オプションを指定します。
		Optimize for General Use (-O)	-O オプションを指定します。
		Optimize for Size (-Osize)	-Osize オプションを指定します。
		Optimize for Speed (-Ospeed)	-Ospeed オプションを指定します。
		No Optimizations (-Onone)	-Onone オプションを指定します。
		指定しない (オプション指定なし)	Optimization Strategy を指定しません。

(3) [プリプロセス]

プリプロセスに関する詳細情報の表示、および設定の変更を行います。

Include Directories	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字、小文字の区別はありません。	
	デフォルト	Include Directories[定義数]
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	247 文字までの文字列 256 個まで指定可能です。

システム・インクルード・パス	コンパイル時に CS+ が設定するインクルード・パスの指定順を変更します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 システム・インクルード・パスは、追加のインクルード・パスより低い優先度で検索します。 パスはプロジェクト・フォルダを基点とします。 コンパイラ・ドライバの -I オプションに相当します。 インクルード・パスはサブプロパティとして表示します。	
	デフォルト	システム・インクルード・パス [定義数]
	変更方法	[...] ボタンをクリックし、システム・インクルード・パス順設定 ダイアログによる編集
	指定可能値	変更不可（インクルード・パスの設定順の変更のみ可能）
Define Preprocessor Symbol	コンパイラ・ドライバの -D オプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。	
	デフォルト	Define Preprocessor Symbol [定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

(4) [出力コード]

出力コードに関する詳細情報の表示、および設定の変更を行います。

Special Data Area	コンパイラ・ドライバの -sda オプションと -zda オプションを設定します。		
	デフォルト	指定しない (オプション指定なし)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Small Data Area(-sda)	-sda オプションを指定します。
		Small Data Area with Threshold(-sda=size)	-sda=size オプションを指定します。
		Zero Data Area with Threshold(-zda=size)	-zda=size オプションを指定します。
		指定しない (オプション指定なし)	Special Data Area を指定しません。

Threshold Value	コンパイラ・ドライバの -sda オプション, または -zda オプションの Threshold Value を指定します。 なお, 本プロパティは, [Special Data Area] プロパティで [Small Data Area with Threshold(-sda=size)], または [Zero Data Area with Threshold(-zda=size)] を選択した場合のみ表示します。		
	デフォルト	空欄	
	変更方法	テキスト・ボックスによる直接入力	
	指定可能値	10 進数, または空欄	
23-bit SDA	コンパイラ・ドライバの -large_sda オプションを設定します。		
	デフォルト	指定しない (オプション指定なし)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Generate 23-bit SDA relocations for load/store instructions(-large_sda)	-large_sda オプションを指定します。
		指定しない (オプション指定なし)	-large_sda オプションを指定しません。

(5) [その他]

コンパイルに関するその他の詳細情報の表示, および設定の変更を行います。

Intermediate Forms of Output	コンパイラ・ドライバの -c オプションを設定します。		
	デフォルト	Object File(-c)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Object File(-c)	-c オプションを指定します。
		指定しない (オプション指定なし)	-c オプションを指定しません。

コンパイル前に実行するコマンド	コンパイル処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %CompiledFile% : コンパイル時の出力ファイルの絶対パスに置換します。 %InputFile% : コンパイル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、コンパイル処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。
デフォルト	コンパイル前に実行するコマンド [定義数]
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
指定可能値	1023 文字までの文字列 64 個まで指定可能です。

コンパイル後に実行するコマンド	コンパイル処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %CompiledFile% : コンパイル時の出力ファイルの絶対パスに置換します。 %InputFile% : コンパイル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、コンパイル処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。		
	<table border="1"> <tr> <td>デフォルト</td><td>コンパイル後に実行するコマンド [定義数]</td></tr> </table>	デフォルト	コンパイル後に実行するコマンド [定義数]
デフォルト	コンパイル後に実行するコマンド [定義数]		
	<table border="1"> <tr> <td>変更方法</td><td>[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能</td></tr> </table>	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能		
	<table border="1"> <tr> <td>指定可能値</td><td>1023 文字までの文字列 64 個まで指定可能です。</td></tr> </table>	指定可能値	1023 文字までの文字列 64 個まで指定可能です。
指定可能値	1023 文字までの文字列 64 個まで指定可能です。		
その他の追加オプション	その他に追加するコンパイル・オプションを入力します。 ここで設定したオプションは、コンパイル・オプション群の最後に付加します。		
	<table border="1"> <tr> <td>デフォルト</td><td>空欄</td></tr> </table>	デフォルト	空欄
デフォルト	空欄		
	<table border="1"> <tr> <td>変更方法</td><td>テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集</td></tr> </table>	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集		
	<table border="1"> <tr> <td>指定可能値</td><td>32767 文字までの文字列</td></tr> </table>	指定可能値	32767 文字までの文字列
指定可能値	32767 文字までの文字列		

[アセンブル・オプション] タブ

本タブでは、アセンブル・フェーズに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [デバッグ情報]
- (2) [プリプロセス]
- (3) [その他]

備考 コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 本タブはプロジェクトの種類に [空のアプリケーション (GHS CCRH850)]、または [ライブラリ (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

[各カテゴリの説明]

- (1) [デバッグ情報]
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	MULTI(-G)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。

- (2) [プリプロセス]
プリプロセスに関する詳細情報の表示、および設定の変更を行います。

Include Directories	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字、小文字の区別はありません。		
	デフォルト	Include Directories[定義数]	
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	
	指定可能値	247 文字までの文字列 256 個まで指定可能です。	

システム・インクルード・パス	アセンブル時に CS+ が設定するインクルード・パスの指定順を変更します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 システム・インクルード・パスは、追加のインクルード・パスより低い優先度で検索します。 パスはプロジェクト・フォルダを基点とします。 コンパイラ・ドライバの -I オプションに相当します。 インクルード・パスはサブプロパティとして表示します。	
	デフォルト	システム・インクルード・パス [定義数]
	変更方法	[...] ボタンをクリックし、システム・インクルード・パス順設定 ダイアログによる編集
	指定可能値	変更不可（インクルード・パスの設定順の変更のみ可能）
Define Preprocessor Symbol	コンパイラ・ドライバの -D オプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。	
	デフォルト	Define Preprocessor Symbol [定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

(3) [その他]

アセンブルに関するその他の詳細情報の表示、および設定の変更を行います。

Intermediate Forms of Output	コンパイラ・ドライバの -c オプションを設定します。				
	デフォルト	Object File(-c)			
	変更方法	ドロップダウン・リストによる選択			
	指定可能値	<table><tr><td>Object File(-c)</td><td>-c オプションを指定します。</td></tr><tr><td>指定しない (オプション指定なし)</td><td>-c オプションを指定しません。</td></tr></table>	Object File(-c)	-c オプションを指定します。	指定しない (オプション指定なし)
Object File(-c)	-c オプションを指定します。				
指定しない (オプション指定なし)	-c オプションを指定しません。				

アセンブル前に実行するコマンド	アセンブル処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %AssembledFile% : アセンブル時の出力ファイルの絶対パスに置換します。 %BuildModeName% : ビルド・モード名に置換します。 %InputFile% : アセンブル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、アセンブル処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。
デフォルト	アセンブル前に実行するコマンド [定義数]
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
指定可能値	1023 文字までの文字列 64 個まで指定可能です。

アセンブル後に実行するコマンド	アセンブル処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %AssembledFile% : アセンブル時の出力ファイルの絶対パスに置換します。 %BuildModeName% : ビルド・モード名に置換します。 %InputFile% : アセンブル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、アセンブル処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。		
	<table border="1"> <tr> <td>デフォルト</td><td>アセンブル後に実行するコマンド [定義数]</td></tr> </table>	デフォルト	アセンブル後に実行するコマンド [定義数]
デフォルト	アセンブル後に実行するコマンド [定義数]		
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能		
指定可能値	1023 文字までの文字列 64 個まで指定可能です。		
その他の追加オプション	その他に追加するアセンブル・オプションを入力します。 必要に応じて -asm= を付加してください。 ここで設定したオプションは、アセンブル・オプション群の最後に付加します。		
	<table border="1"> <tr> <td>デフォルト</td><td>空欄</td></tr> </table>	デフォルト	空欄
デフォルト	空欄		
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集		
指定可能値	32767 文字までの文字列		

[リンク・オプション] タブ

本タブでは、リンク・フェーズに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [デバッグ情報]
- (2) [出力ファイル]
- (3) [ライブラリ]
- (4) [ヘキサ出力]
- (5) [リスト]
- (6) [その他]

備考 コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 1. 本タブは、ライブラリ用のプロジェクトの場合は表示しません。

注意 2. 本タブはプロジェクトの種類に [空のアプリケーション (GHS CCRH850)]、または [ライブラリ (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

[各カテゴリの説明]

- (1) [デバッグ情報]
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	MULTI(-G)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。

- (2) [出力ファイル]
出力ファイルに関する詳細情報の表示、および設定の変更を行います。

出力フォルダ	出力フォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 空欄の場合は、プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -o オプションに相当します。		
	デフォルト	%BuildModeName%	
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集	
	指定可能値	247 文字までの文字列	

出力ファイル名	出力ファイル名を指定します。 次のプレースホルダに対応しています。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %ProjectName% : プロジェクト名に置換します。 コンパイラ・ドライバの -o オプションに相当します。	
	デフォルト	%ProjectName%
	変更方法	テキスト・ボックスによる直接入力
	指定可能値	259 文字までの文字列

(3) [ライブラリ]

ライブラリに関する詳細情報の表示、および設定の変更を行います。

Alternate Library Directory	コンパイラ・ドライバの -YL オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 ライブラリ・ディレクトリ名に大文字、小文字の区別はありません。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集
	指定可能値	247 文字までの文字列

Libraries	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 ライブラリ・ファイル名はサブプロパティとして表示します。	
	デフォルト	Libraries[定義数]
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 → [参照] ボタンをクリックし、使用するライブラリ・ファイルを指定 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	259 文字までの文字列 65536 個まで指定可能です。
システム・ライブラリ・ファイル	リンク時に CS+ が設定するライブラリ・ファイルを表示します。 コンパイラ・ドライバの -I オプションに相当します。 システム・ライブラリ・ファイル名はサブプロパティとして表示します。	
	デフォルト	システム・ライブラリ・ファイル[定義数]
	指定可能値	変更不可

(4) [ヘキサ出力]

ヘキサ出力に関する詳細情報の表示、および設定の変更を行います。

Generate Additional Output	コンパイラ・ドライバの -srec オプション、または -hex オプションを指定します。	
	デフォルト	S-Record File(-srec)
	変更方法	ドロップダウン・リストによる選択
	指定可能値	S-Record File(-srec) -srec オプションを指定します。
		HEX386 File(-hex) -hex オプションを指定します。
		指定しない (オプション指定なし) Generate Additional Output を指定しません。

出力フォルダ	ヘキサ・ファイルの出力フォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 空欄の場合は、プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -srec オプション、または -hex オプションに相当します。 なお、本プロパティは、[Generate Additional Output] プロパティで [指定しない(オプション指定なし)] 以外を選択した場合のみ表示します。	
	デフォルト	%BuildModeName%
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集
	指定可能値	247 文字までの文字列
	出力ファイル名	ヘキサ・ファイル名を指定します。 拡張子を省略した場合は、".run" を自動的に付加します。 次のプレースホルダに対応しています。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %ProjectName% : プロジェクト名に置換します。 コンパイラ・ドライバの -srec オプション、または -hex オプションに相当します。 なお、本プロパティは、[Generate Additional Output] プロパティで [指定しない(オプション指定なし)] 以外を選択した場合のみ表示します。
デフォルト		%ProjectName%.run
変更方法		テキスト・ボックスによる直接入力
指定可能値		259 文字までの文字列

(5) [リスト]

リストに関する詳細情報の表示、および設定の変更を行います。

Map File Generation	コンパイラ・ドライバの -map オプションを指定します。		
	デフォルト	Generate Default Map File(-map)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Generate Default Map File(-map)	-map オプションを指定します。
		指定しない (オプション指定なし)	-map オプションを指定しません。

(6) [その他]

リンクに関するその他の詳細情報の表示、および設定の変更を行います。

Start Files	コンパイラ・ドライバの -nostartfiles オプションを指定します。		
	デフォルト	指定しない (オプション指定なし)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Do Not Use Start Files(-nostartfiles)	-nostartfiles オプションを指定します。
リンク前に実行するコマンド		指定しない (オプション指定なし)	-nostartfiles オプションを指定しません。
	リンク処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください (例 : call a.bat)。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %LinkedFile% : リンク処理時の出力ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に "#!python" と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、リンク処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。		
	デフォルト	リンク前に実行するコマンド [定義数]	
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	
	指定可能値	1023 文字までの文字列 64 個まで指定可能です。	

リンク後に実行するコマンド	リンク処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %LinkedFile% : リンク処理時の出力ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、リンク処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。	
	デフォルト	リンク後に実行するコマンド [定義数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	1023 文字までの文字列 64 個まで指定可能です。
その他の追加オプション	その他に追加するリンク・オプションを入力します。 なお、ここで設定したオプションは、リンク・オプション群の最後に付加されます。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集
	指定可能値	32767 文字までの文字列

「ライブラリ生成オプション」タブ

本タブでは、ライブラリ生成フェーズに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [デバッグ情報]
- (2) [出力ファイル]
- (3) [ライブラリ]
- (4) [リスト]
- (5) [その他]

備考 コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 1. 本タブは、ライブラリ用のプロジェクトの場合は表示しません。

注意 2. 本タブはプロジェクトの種類に [空のアプリケーション (GHS CCRH850)]、または [ライブラリ (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

「各カテゴリの説明」

- (1) [デバッグ情報]
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	MULTI(-G)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。

- (2) [出力ファイル]
出力ファイルに関する詳細情報の表示、および設定の変更を行います。

出力フォルダ	出力フォルダを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 空欄の場合は、プロジェクト・フォルダを指定したものとみなします。 コンパイラ・ドライバの -o オプションに相当します。		
	デフォルト	%BuildModeName%	
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集	
	指定可能値	247 文字までの文字列	

出力ファイル名	出力ファイル名を指定します。 次のプレースホルダに対応しています。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %ProjectName% : プロジェクト名に置換します。 コンパイラ・ドライバの -o オプションに相当します。	
	デフォルト	lib%ProjectName%.a
	変更方法	テキスト・ボックスによる直接入力
	指定可能値	259 文字までの文字列

(3) [ライブラリ]

ライブラリに関する詳細情報の表示、および設定の変更を行います。

Alternate Library Directory	コンパイラ・ドライバの -YL オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 ライブラリ・ディレクトリ名に大文字、小文字の区別はありません。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、フォルダの参照 ダイアログによる編集
	指定可能値	247 文字までの文字列

Libraries	コンパイラ・ドライバの -l オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 ライブラリ・ファイル名はサブプロパティとして表示します。	
	デフォルト	Libraries[定義数]
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 → [参照] ボタンをクリックし、使用するライブラリ・ファイルを指定 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	259 文字までの文字列 65536 個まで指定可能です。
	システム・ライブラリ・ファイル	リンク時に CS+ が設定するライブラリ・ファイルを表示します。 コンパイラ・ドライバの -l オプションに相当します。 システム・ライブラリ・ファイル名はサブプロパティとして表示します。
デフォルト		システム・ライブラリ・ファイル[定義数]
指定可能値		変更不可

(4) [リスト]

リストに関する詳細情報の表示、および設定の変更を行います。

Map File Generation	コンパイラ・ドライバの -map オプションを指定します。		
	デフォルト	指定しない (オプション指定なし)	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Generate Default Map File(-map)	-map オプションを指定します。
		指定しない (オプション指定なし)	-map オプションを指定しません。

(5) [その他]

ライブラリ生成に関するその他の詳細情報の表示、および設定の変更を行います。

ライブラリ生成前に実行するコマンド	ライブラリ生成処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %LibraryFile% : ライブラリ生成処理時の出力ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、ライブラリ生成処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。
デフォルト	ライブラリ生成前に実行するコマンド [定義数]
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
指定可能値	1023 文字までの文字列 64 個まで指定可能です。

ライブラリ生成後に実行するコマンド	ライブラリ生成処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %LibraryFile% : ライブラリ生成処理時の出力ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、ライブラリ生成処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。
デフォルト	ライブラリ生成後に実行するコマンド [定義数]
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
指定可能値	1023 文字までの文字列 64 個まで指定可能です。
その他の追加オプション	その他に追加するライブラリ生成オプションを入力します。 なお、ここで設定したオプションは、ライブラリ生成オプション群の最後に付加されます。
デフォルト	空欄
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集
指定可能値	32767 文字までの文字列

[I/O ヘッダ・ファイル生成オプション] タブ

本タブでは、I/O ヘッダ・ファイル生成ツールに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [I/O ヘッダ・ファイル]
- (2) [その他]

[各カテゴリの説明]

- (1) [I/O ヘッダ・ファイル]

I/O ヘッダ・ファイルに関する詳細情報の表示、および設定の変更を行います。

ビルド時に I/O ヘッダ・ファイルを更新する	ビルド時に I/O ヘッダ・ファイルを更新するかどうかを選択します。 デバイス・ファイルが I/O ヘッダ・ファイル生成時より新しい場合や I/O ヘッダ・ファイル生成関連のプロパティが更新されている場合、I/O ヘッダ・ファイルを更新します。更新は自動的に上書きで行い、拡張子 bak でバックアップ・ファイルを作成します。 この内容はすべてのビルド・モードで共通です。			
	デフォルト	いいえ		
	変更方法	ドロップダウン・リストによる選択		
	指定可能値	はい (デバイス・ファイルのみチェック)	ビルド時にデバイス・ファイルが更新されている場合、I/O ヘッダ・ファイルを更新します。	
		はい (プロパティのみチェック)	ビルド時にプロパティが更新されている場合、I/O ヘッダ・ファイルを更新します。	
はい (デバイス・ファイルとプロパティをチェック)		ビルド時にデバイス・ファイル、またはプロパティが更新されている場合、I/O ヘッダ・ファイルを更新します。		
いいえ		ビルド時に I/O ヘッダ・ファイルを更新しません。		
I/O ヘッダ・ファイル生成時のデバイス・ファイル	I/O ヘッダ・ファイルを生成した時のデバイス・ファイルのファイル名とバージョンを表示します。 なお、本プロパティは、[ビルド時に I/O ヘッダ・ファイルを更新する] プロパティで [いいえ] 以外を選択した場合のみ表示します。			
	デフォルト	I/O ヘッダ・ファイル生成時のデバイス・ファイルとバージョン		
	変更方法	変更不可		
現在のデバイス・ファイル	実行中の CS+ 環境にインストールされているデバイス・ファイルのファイル名とバージョンを表示します。 なお、本プロパティは、[ビルド時に I/O ヘッダ・ファイルを更新する] プロパティで [いいえ] 以外を選択した場合のみ表示します。			
	デフォルト	現在のデバイス・ファイル		
	変更方法	変更不可		

ファイルに出力するモジュールを選択する	I/O ヘッダ・ファイルに出力するモジュールを選択するかどうかを選択します。			
	デフォルト	いいえ		
	変更方法	ドロップダウン・リストによる選択		
	指定可能値	はい	I/O ヘッダ・ファイルに出力するモジュールを選択するために、[デバイス・ファイルで定義されているモジュール一覧] プロパティを表示します。 選択したモジュール内の IOR のみを I/O ヘッダ・ファイルに出力します。	
		いいえ	I/O ヘッダ・ファイルにすべてのモジュールを出力します。	
デバイス・ファイルで定義されているモジュール一覧	デバイス・ファイルで定義されているモジュールの一覧を表示します。 サブプロパティには、以下の項目を表示します。 モジュール名 : デバイス・ファイルで定義されているモジュール名 ファイル名 : 出力先の I/O ヘッダ・ファイル名 出力状況 : I/O ヘッダ・ファイルに出力する／しないの設定状況 なお、本プロパティは、[ファイルに出力するモジュールを選択する] プロパティで [はい] を選択した場合のみ表示します。			
	デフォルト	[デバイス・ファイルで定義されているモジュールの総数]		
	変更方法	[...] ボタンをクリックし、 ファイルに出力するモジュールの選択ダイアログ による編集 サブプロパティの直接入力による編集は不可		
	指定可能値	259 文字までの文字列		
μITRON を考慮した定義を出力する	μITRON を考慮した定義を出力するかどうかを選択します。			
	デフォルト	いいえ		
	変更方法	ドロップダウン・リストによる選択		
	指定可能値	はい (-uitron=on)	μITRON を考慮した定義を出力します。	
		いいえ	μITRON を考慮した定義を出力しません。	
MISRA-C オプションを有効にする	MISRA-C ルールに適合した I/O ヘッダ・ファイルを出力するかどうかを選択します。			
	デフォルト	いいえ		
	変更方法	ドロップダウン・リストによる選択		
	指定可能値	はい (-misra_c=on2)	MISRA-C ルールに適合した I/O ヘッダ・ファイルを出力します。 共用体を出力しないため、同一アドレスに配置されている IOR へのアクセスは、最大アクセス・サイズのものに限られます。	
		いいえ	MISRA-C ルールを考慮しません。	

モジュール配列化オプションを有効にする	0 から始まる番号を持つモジュールに対して、配列でアクセス可能となる定義をヘッダ・ファイルに出力するかどうかを選択します。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい (-modulearray=on)	モジュール配列化オプションを有効にします。
		いいえ	モジュール配列化オプションを有効にしません。
モジュール中にブロックを定義する	モジュール中にブロックを定義するかどうかを選択します。 なお、本プロパティは、ブロック定義用の情報ファイルが存在する場合のみ表示します。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	モジュール中にブロックを定義して、ブロック単位での配列化と IOR の配列化を実施します。
		いいえ	モジュール中にブロックを定義しません。
IOR 配列化オプションを有効にする	0 から始まる番号を持つ IOR に対して、配列でアクセス可能となる定義を出力するかどうかを選択します。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい (-iorarray=on)	IOR 配列化オプションを有効にします。
		いいえ	IOR 配列化オプションを有効にしません。
構造体定義を共有する	構造体定義を共有するかどうかを選択します。		
	デフォルト	はい	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	構造体定義を共有します。
		いいえ (-share_structure=off)	構造体定義を共有しません。

(2) [その他]

I/O ヘッダ・ファイルに関するその他の詳細情報の表示、および設定の変更を行います。

その他の追加オプション	その他の追加する I/O ヘッダ・ファイル生成オプションを入力します。 なお、ここで設定したオプションは、I/O ヘッダ・ファイル生成オプション群の最後に付加されます。		
	デフォルト	空欄	
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力ダイアログによる編集	
	指定可能値	259 文字までの文字列	

[ビルド設定] タブ

本タブでは、各 C ソース・ファイル、アセンブリ・ソース・ファイル、リンク・ディレクティブ・ファイル、オブジェクト・ファイル、ライブラリ・ファイルに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

(1) [ビルド]

[各カテゴリの説明]

(1) [ビルド]

ビルドに関する詳細情報の表示、および設定の変更を行います。

ビルドの対象とする	選択しているファイルをビルド対象とするかどうかを選択します。		
	デフォルト	はい	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	選択しているファイルをビルド対象とします。
		いいえ	選択しているファイルをビルド対象としません。
個別コンパイル・オプションを設定する	選択している C ソース・ファイルにプロジェクトの設定とは異なるコンパイル・オプションを設定するかどうかを選択します。 なお、本プロパティは、プロジェクト・ツリーで C ソース・ファイルを選択し、[ビルドの対象とする] プロパティで [はい] を選択した場合のみ表示します。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	選択している C ソース・ファイルにプロジェクトの設定とは異なるオプションを設定します。
		いいえ	選択している C ソース・ファイルにプロジェクトの設定とは異なるオプションを設定しません。
個別アセンブル・オプションを設定する	選択しているアセンブリ・ソース・ファイルにプロジェクトの設定とは異なるアセンブル・オプションを設定するかどうかを選択します。 なお、本プロパティは、プロジェクト・ツリーでアセンブリ・ソース・ファイルを選択し、[ビルドの対象とする] プロパティで [はい] を選択した場合のみ表示します。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	選択しているアセンブリ・ソース・ファイルにプロジェクトの設定とは異なるオプションを設定します。
		いいえ	選択しているアセンブリ・ソース・ファイルにプロジェクトの設定とは異なるオプションを設定しません。
ファイル形式	選択しているファイルの形式を表示します。		
	デフォルト	C ソース・ファイル (C ソース・ファイルを選択している場合) アセンブリ・ソース・ファイル (アセンブリ・ソース・ファイルを選択している場合) リンク・ディレクティブ (リンク・ディレクティブ・ファイルを選択している場合) オブジェクト・ファイル (オブジェクト・ファイルを選択している場合) ライブラリ・ファイル (ライブラリ・ファイルを選択している場合)	
	変更方法	変更不可	

〔個別コンパイル・オプション〕 タブ

本タブでは、1つのCソース・ファイルに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

なお、本タブは、[〔共通オプション〕タブ](#)、および[〔コンパイル・オプション〕タブ](#)の設定内容を継承します。

これらのタブと異なる値を設定した場合は、プロパティが太字表示となります。

- (1) [〔デバッグ情報〕](#)
- (2) [〔最適化〕](#)
- (3) [〔プリプロセス〕](#)
- (4) [〔出力コード〕](#)
- (5) [〔出力ファイル〕](#)
- (6) [〔その他〕](#)

備考 1. 本タブは、[〔ビルド設定〕タブ](#)の[〔ビルド〕](#)カテゴリの[〔個別コンパイル・オプションを設定する〕](#)プロパティで「はい」を選択した場合のみ表示します。

備考 2. コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 本タブはプロジェクトの種類に「空のアプリケーション (GHS CCRH850)」, または「ライブラリ (GHS CCRH850)」を指定して作成したプロジェクトの場合のみ表示します。

〔各カテゴリの説明〕

- (1) [〔デバッグ情報〕](#)
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。
Generate MULTI and Native Information	コンパイラ・ドライバの -dwarf2 オプションを指定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	On(-dwarf2)	-dwarf2 オプションを指定します。
		指定しない (オプション指定なし)	-dwarf2 オプションを指定しません。

- (2) [〔最適化〕](#)
最適化に関する詳細情報の表示、および設定の変更を行います。

Optimization Strategy	コンパイラ・ドライバの -O オプションを指定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Optimize for Debuggability (-Odebug)	-Odebug オプションを指定します。
		Optimize for General Use (-O)	-O オプションを指定します。
		Optimize for Size (-Osize)	-Osize オプションを指定します。
		Optimize for Speed (-Ospeed)	-Ospeed オプションを指定します。
		No Optimizations (-Onone)	-Onone オプションを指定します。
		指定しない (オプション指定なし)	Optimization Strategy を指定しません。

(3) [プリプロセス]

プリプロセスに関する詳細情報の表示、および設定の変更を行います。

Include Directories	コンパイラ・ドライバの -I オプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字、小文字の区別はありません。		
	デフォルト	Include Directories[定義数]	
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	
	指定可能値	247 文字までの文字列	
		256 個まで指定可能です。	

ビルド・ツールに指定した全体インクルード・パスも使用する	使用するビルド・ツールの 【コンパイル・オプション】 タブの 【プリプロセス】 カテゴリの 【Include Directories】 プロパティで指定したインクルード・パスも使用してコンパイルするかどうかを選択します。 インクルード・パスは、以下の順で追加します。 - 本タブの 【Include Directories】 プロパティで指定したパス - 【コンパイル・オプション】 タブの 【Include Directories】 プロパティで指定したパス - 【コンパイル・オプション】 タブの 【システム・インクルード・パス】 プロパティに表示しているパス コンパイラ・ドライバの -I オプションに相当します。		
	デフォルト	はい	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	使用するビルド・ツールのプロパティで指定したインクルード・パスも使用してコンパイルします。
		いいえ	使用するビルド・ツールのプロパティで指定したインクルード・パスを使用しません。
Define Preprocessor Symbol	コンパイラ・ドライバの -D オプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。		
	デフォルト	Define Preprocessor Symbol[定義数]	
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	
	指定可能値	256 文字までの文字列 256 個まで指定可能です。	

(4) 【出力コード】

出力コードに関する詳細情報の表示、および設定の変更を行います。

Special Data Area	コンパイラ・ドライバの -sda オプションと -zda オプションを設定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Small Data Area(-sda)	-sda オプションを指定します。
		Small Data Area with Threshold(-sda=size)	-sda=size オプションを指定します。
		Zero Data Area with Threshold(-zda=size)	-zda=size オプションを指定します。
		指定しない (オプション指定なし)	Special Data Area を指定しません。
Threshold Value	コンパイラ・ドライバの -sda オプション、または -zda オプションの Threshold Value を指定します。 なお、本プロパティは、【Special Data Area】 プロパティで 【Small Data Area with Threshold(-sda=size)】、または 【Zero Data Area with Threshold(-zda=size)】 を選択した場合のみ表示します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	テキスト・ボックスによる直接入力	
	指定可能値	10 進数、または空欄	

23-bit SDA	コンパイラ・ドライバの -large_sda オプションを設定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Generate 23-bit SDA relocations for load/store instructions(-large_sda)	-large_sda オプションを指定します。
		指定しない (オプション指定なし)	-large_sda オプションを指定しません。

(5) [出力ファイル]

出力ファイルに関する詳細情報の表示、および設定の変更を行います。

オブジェクト・ファイル名	コンパイル後に生成するオブジェクト・ファイルの名前を指定します。 “.o” 以外の拡張子を指定することはできません。 拡張子を省略した場合は, “.o” を自動的に付加します。 空欄の場合は, ソース・ファイル名の拡張子を “.o” に置き換えたものとなります。 コンパイラ・ドライバの -o オプションに相当します。		
	デフォルト	空欄	
	変更方法	テキスト・ボックスによる直接入力	
	指定可能値	259 文字までの文字列	

(6) [その他]

コンパイルに関するその他の詳細情報の表示、および設定の変更を行います。

Intermediate Forms of Output	コンパイラ・ドライバの -c オプションを設定します。		
	デフォルト	コンパイル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	Object File(-c)	-c オプションを指定します。
		指定しない (オプション指定なし)	-c オプションを指定しません。

コンパイル前に実行するコマンド	コンパイル処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %CompiledFile% : コンパイル時の出力ファイルの絶対パスに置換します。 %InputFile% : コンパイル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2行目から最終行までの内容を Python コンソールのスクリプトと判断し、コンパイル処理前に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。
デフォルト	コンパイル・オプションの設定値
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
指定可能値	1023 文字までの文字列 64 個まで指定可能です。

コンパイル後に実行するコマンド	コンパイル処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %CompiledFile% : コンパイル時の出力ファイルの絶対パスに置換します。 %InputFile% : コンパイル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、コンパイル処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。		
	<table border="1"> <tr> <td>デフォルト</td><td>コンパイル・オプションの設定値</td></tr> </table>	デフォルト	コンパイル・オプションの設定値
デフォルト	コンパイル・オプションの設定値		
変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能		
指定可能値	1023 文字までの文字列 64 個まで指定可能です。		
その他の追加オプション	その他に追加するコンパイル・オプションを入力します。 ここで設定したオプションは、コンパイル・オプション群の最後に付加します。		
	<table border="1"> <tr> <td>デフォルト</td><td>コンパイル・オプションの設定値</td></tr> </table>	デフォルト	コンパイル・オプションの設定値
デフォルト	コンパイル・オプションの設定値		
変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集		
指定可能値	32767 文字までの文字列		

〔個別アセンブル・オプション〕 タブ

本タブでは、1つのアセンブリ・ソース・ファイルに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

なお、本タブは、[〔共通オプション〕タブ](#)、[〔コンパイル・オプション〕タブ](#)、および[〔アセンブル・オプション〕タブ](#)の設定内容を継承します。

これらのタブと異なる値を設定した場合は、プロパティが太字表示となります。

- (1) [〔デバッグ情報〕](#)
- (2) [〔プリプロセス〕](#)
- (3) [〔出力ファイル〕](#)
- (4) [〔その他〕](#)

備考 1. 本タブは、[〔ビルド設定〕タブ](#)の[〔ビルド〕](#)カテゴリの[〔個別アセンブル・オプションを設定する〕](#)プロパティで「はい」を選択した場合のみ表示します。

備考 2. コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 本タブはプロジェクトの種類に「空のアプリケーション (GHS CCRH850)」, または「ライブラリ (GHS CCRH850)」を指定して作成したプロジェクトの場合のみ表示します。

〔各カテゴリの説明〕

- (1) [〔デバッグ情報〕](#)
デバッグ情報に関する詳細情報の表示、および設定の変更を行います。

Debugging Level	コンパイラ・ドライバの -G オプションを指定します。		
	デフォルト	アセンブル・オプションの設定値	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	MULTI(-G)	-G オプションを指定します。
		指定しない (オプション指定なし)	-G オプションを指定しません。

- (2) [〔プリプロセス〕](#)
プリプロセスに関する詳細情報の表示、および設定の変更を行います。

Include Directories	コンパイラ・ドライバの-Iオプションを指定します。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 パスはプロジェクト・フォルダを基点とします。 指定したインクルード・パスはサブプロパティとして表示します。 インクルード・パスに大文字、小文字の区別はありません。				
	デフォルト	Include Directories[定義数]			
	変更方法	[...] ボタンをクリックし、パス編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能			
	指定可能値	247 文字までの文字列 256 個まで指定可能です。			
ビルド・ツールに指定した全体インクルード・パスも使用する	使用するビルド・ツールの 【アセンブル・オプション】 タブの 【プリプロセス】 カテゴリの [Include Directories] プロパティで指定したインクルード・パスも使用してアセンブルするかどうかを選択します。 インクルード・パスは、以下の順で追加します。 - 本タブの [Include Directories] プロパティで指定したパス - 【アセンブル・オプション】 タブの [Include Directories] プロパティで指定したパス - 【アセンブル・オプション】 タブの [システム・インクルード・パス] プロパティに表示しているパス コンパイラ・ドライバの-Iオプションに相当します。				
	デフォルト	はい			
	変更方法	ドロップダウン・リストによる選択			
	指定可能値	<table><tr><td>はい</td><td>使用するビルド・ツールのプロパティで指定したインクルード・パスも使用してアセンブルします。</td></tr><tr><td>いいえ</td><td>使用するビルド・ツールのプロパティで指定したインクルード・パスを使用しません。</td></tr></table>	はい	使用するビルド・ツールのプロパティで指定したインクルード・パスも使用してアセンブルします。	いいえ
はい	使用するビルド・ツールのプロパティで指定したインクルード・パスも使用してアセンブルします。				
いいえ	使用するビルド・ツールのプロパティで指定したインクルード・パスを使用しません。				
Define Preprocessor Symbol	コンパイラ・ドライバの-Dオプションを指定します。 「マクロ名 = 定義値」の形式で 1 行に 1 つずつ指定します。 指定したマクロはサブプロパティとして表示します。				
	デフォルト	Define Preprocessor Symbol[定義数]			
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能			
	指定可能値	256 文字までの文字列 256 個まで指定可能です。			

- (3) **【出力ファイル】**
出力ファイルに関する詳細情報の表示、および設定の変更を行います。

オブジェクト・ファイル名	アセンブル後に生成するオブジェクト・ファイルの名前を指定します。 “.o”以外の拡張子を指定することはできません。 拡張子を省略した場合は, “.o”を自動的に付加します。 空欄の場合は, ソース・ファイル名の拡張子を “.o”に置き換えたものとなります。 コンパイラ・ドライバの -o オプションに相当します。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力
	指定可能値	259 文字までの文字列

(4) [その他]

アセンブルに関するその他の詳細情報の表示, および設定の変更を行います。

Intermediate Forms of Output	コンパイラ・ドライバの -c オプションを設定します。	
	デフォルト	アセンブル・オプションの設定値
	変更方法	ドロップダウン・リストによる選択
	指定可能値	Object File(-c) -c オプションを指定します。 指定しない(オプション指定なし) -c オプションを指定しません。
アセンブル前に実行するコマンド	アセンブル処理前に実行するコマンドを指定します。 バッチファイルを指定する場合は, call 命令を使用してください (例: call a.bat)。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %AssembledFile% : アセンブル時の出力ファイルの絶対パスに置換します。 %BuildModeName% : ビルド・モード名に置換します。 %InputFile% : アセンブル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に “#!python” と記述すると, 2 行目から最終行までの内容を Python コンソールのスクリプトと判断し, アセンブル処理前に Python コンソールで実行します。 なお, スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。	
	デフォルト	アセンブル・オプションの設定値
	変更方法	[...] ボタンをクリックし, テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	1023 文字までの文字列 64 個まで指定可能です。

アセンブル後に実行するコマンド	アセンブル処理後に実行するコマンドを指定します。 バッチファイルを指定する場合は、call 命令を使用してください（例：call a.bat）。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %AssembledFile% : アセンブル時の出力ファイルの絶対パスに置換します。 %BuildModeName% : ビルド・モード名に置換します。 %InputFile% : アセンブル対象ファイルの絶対パスに置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %Options% : ビルド実行時のコマンド・ライン・オプションに置換します。 %OutputDir% : 出力フォルダの絶対パスに置換します。 %OutputFile% : 出力ファイルの絶対パスに置換します。 %Program% : 実行中のプログラム名に置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 先頭行に“#!python”と記述すると、2 行目から最終行までの内容を Python コンソールのスクリプトと判断し、アセンブル処理後に Python コンソールで実行します。 なお、スクリプト中にはプレースホルダの記述も可能です。 指定したコマンドはサブプロパティとして表示します。		
	デフォルト	アセンブル・オプションの設定値	
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能	
	指定可能値	1023 文字までの文字列 64 個まで指定可能です。	
	その他の追加オプション	その他に追加するアセンブル・オプションを入力します。 ここで設定したオプションは、アセンブル・オプション群の最後に付加します。	
		デフォルト	アセンブル・オプションの設定値
		変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、文字列入力 ダイアログによる編集
		指定可能値	32767 文字までの文字列

[ビルド・オプション] タブ

本タブでは、ビルド・ツールに対して、次に示すカテゴリごとに詳細情報の表示、および設定の変更を行います。

- (1) [ビルド・モード]
- (2) [ビルド]
- (3) [ビルド方法]
- (4) [ツールのパス]
- (5) [記録]

備考 コンパイラ・ドライバの各オプションについては、Green Hills Software, LLC の「V850/RH850 アプリケーション構築ガイド」を参照してください。CS+ の GHS CCRH850 ビルド・ツール・プラグインに関して、Green Hills Software, LLC に問い合わせないでください。

注意 1. 本タブはプロジェクトの種類に [既存 GHS プロジェクト・ファイル使用 (GHS CCRH850)] を指定して作成したプロジェクトの場合のみ表示します。

注意 2. GHS コンパイラをバージョンアップした場合は、[ツールのパス] カテゴリの [コンパイラ・パッケージのフォルダ] プロパティの設定を確認してください。

[各カテゴリの説明]

- (1) [ビルド・モード]

ビルド・モードに関する詳細情報の表示、および設定の変更を行います。

ビルド・モード	ビルド時に使用するビルド・モードを選択します。 なお、本プロパティには、コンテキスト・メニュー→ [すべてデフォルトに戻す] は適用されません。		
	デフォルト	DefaultBuild	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	DefaultBuild	プロジェクトの新規作成時にデフォルトで設定するビルド・モードでビルドを行います。
		プロジェクトに登録しているビルド・モード	プロジェクトに登録しているビルド・モード（DefaultBuild 以外）でビルドを行います。
すべてのビルド・モードのプロパティを一括して変更する	プロパティに値を設定する際、設定した値をすべてのビルド・モードに反映するかどうかを選択します。 設定した値が他のビルド・モードの値として適していない可能性もあるため、充分ご注意ください。		
	デフォルト	いいえ	
	変更方法	ドロップダウン・リストによる選択	
	指定可能値	はい	プロパティに値を設定する際、設定した値をすべてのビルド・モードに反映します。
		いいえ	プロパティに値を設定する際、設定した値を他のビルド・モードに反映しません。

- (2) [ビルド]

ビルドに関する詳細情報の表示、および設定の変更を行います。

ビルドする MULTI のプロジェクト・ファイル	プロジェクトのビルド時に参照する MULTI のプロジェクト・ファイルを指定します。 この内容はすべてのビルド・モードで共通です。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 本プロパティ、または [ビルドする MULTI のトップ・プロジェクト・ファイル] プロパティのうち、少なくともいずれか 1 つを指定してください。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、MULTI のプロジェクト・ファイルの指定 ダイアログによる編集
	指定可能値	259 文字までの文字列
ビルドする MULTI のトップ・プロジェクト・ファイル	プロジェクトのビルド時に参照する MULTI のトップ・プロジェクト・ファイルを指定します。 この内容はすべてのビルド・モードで共通です。 次のプレースホルダに対応しています。 %ActiveProjectDir% : アクティブ・プロジェクト・フォルダの絶対パスに置換します。 %ActiveProjectName% : アクティブ・プロジェクト名に置換します。 %BuildModeName% : ビルド・モード名に置換します。 %MainProjectDir% : メイン・プロジェクト・フォルダの絶対パスに置換します。 %MainProjectName% : メイン・プロジェクト名に置換します。 %MicomToolPath% : 本製品のインストール・フォルダの絶対パスに置換します。 %ProjectDir% : プロジェクト・フォルダの絶対パスに置換します。 %ProjectName% : プロジェクト名に置換します。 %TempDir% : テンポラリ・フォルダの絶対パスに置換します。 %WinDir% : Windows システム・フォルダの絶対パスに置換します。 本プロパティ、または [ビルドする MULTI のプロジェクト・ファイル] プロパティのうち、少なくともいずれか 1 つを指定してください。 gbuild コマンドの -top オプションに相当します。	
	デフォルト	空欄
	変更方法	テキスト・ボックスによる直接入力、または [...] ボタンをクリックし、MULTI のプロジェクト・ファイルの指定 ダイアログによる編集
	指定可能値	259 文字までの文字列

クリーン時に付加するオプション	クリーン時にコマンドラインに付加するオプションを指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	-clean
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, 文字列入力 ダイアログによる編集
	指定可能値	259 文字までの文字列

(3) [ビルド方法]

ビルド方法に関する詳細情報の表示, および設定の変更を行います。

パラレル・ビルドを行う	パラレル・ビルド機能を有効にするかどうかを選択します。 パラレル・ビルド機能により, CS+ はコンピュータが搭載しているすべてのプロセッサを使用してパラレルに複数のソース・ファイルをコンパイル／アセンブルします。これにより, より速くコンパイル／アセンブルすることができます。 さらに, プロジェクト間のパラレル・ビルドは, [ツール] メニュー→[オプション] を選択し, オプション ダイアログの [全般 - ビルド] カテゴリにて設定することができます。 gbuild コマンドの -parallel オプションに相当します。	
	デフォルト	いいえ
	変更方法	ドロップダウン・リストによる選択
	指定可能値	はい パラレル・ビルドを有効にします。
		いいえ パラレル・ビルドを無効にします。

(4) [ツールのパス]

ツールのパスに関する詳細情報の表示, および設定の変更を行います。

コンパイラ・パッケージのフォルダ	使用するコンパイラ・パッケージのフォルダを指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	インストール環境に応じた値
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, フォルダの参照 ダイアログによる編集
	指定可能値	247 文字までの文字列
ビルド用実行ファイル名	ビルド用実行ファイル名を指定します。 この内容はすべてのビルド・モードで共通です。	
	デフォルト	gbuild.exe
	変更方法	テキスト・ボックスによる直接入力, または [...] ボタンをクリックし, 文字列入力 ダイアログによる編集
	指定可能値	259 文字までの文字列

(5) [記録]

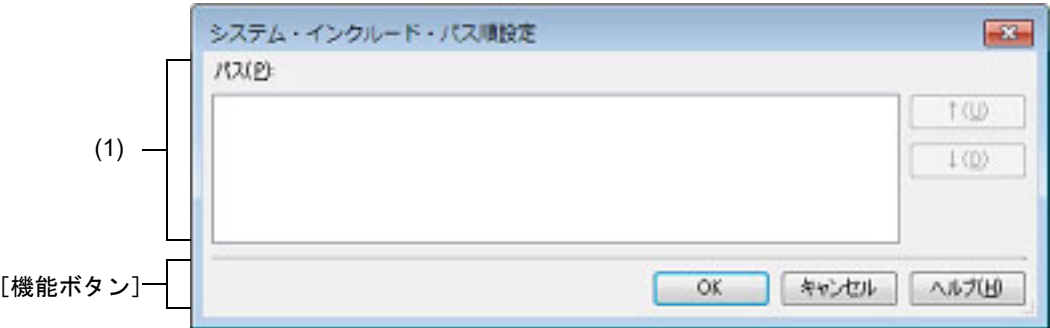
記録に関する詳細情報の表示, および設定の変更を行います。

メモ	このビルド・ツールにメモを追加します。 1 行に 1 項目ずつ指定します。 この設定はすべてのビルド・モードで共通です。 追加したメモはサブプロパティとして表示します。	
	デフォルト	メモ [項目数]
	変更方法	[...] ボタンをクリックし、テキスト編集 ダイアログによる編集 サブプロパティはテキスト・ボックスによる直接入力も可能
	指定可能値	256 文字までの文字列 256 個まで指定可能です。

システム・インクルード・パス順設定 ダイアログ

コンパイラに対して指定するシステム・インクルード・パスの参照，および指定順の設定を行います。

図 A.2 システム・インクルード・パス順設定 ダイアログ



ここでは，次の項目について説明します。

- [\[オープン方法\]](#)
- [\[各エリアの説明\]](#)
- [\[機能ボタン\]](#)

[オープン方法]

- [プロパティ パネル](#)において，以下のプロパティを選択したのち，[...] ボタンをクリック
 - [\[共通オプション\] タブ](#)の [\[よく使うオプション \(コンパイル\)\]](#) カテゴリの [\[システム・インクルード・パス\]](#)，[\[よく使うオプション \(アセンブル\)\]](#) カテゴリの [\[システム・インクルード・パス\]](#)
 - [\[コンパイル・オプション\] タブ](#)の [\[プリプロセス\]](#) カテゴリの [\[システム・インクルード・パス\]](#)
 - [\[アセンブル・オプション\] タブ](#)の [\[プリプロセス\]](#) カテゴリの [\[システム・インクルード・パス\]](#)

[各エリアの説明]

- (1) [パス一覧表示エリア](#)
コンパイラに対して指定するシステム・インクルード・パスの一覧を表示します。
 - (a) [\[パス\]](#)
システム・インクルード・パス名の一覧を，コンパイラへの指定順に表示します。
デフォルトでは，プロジェクトに登録している順番となります。
パスの表示順を変更することにより，コンパイラへの指定順を設定することができます。
表示順の変更は，[↑]，および [↓] ボタン，またはパス名のドラッグ・アンド・ドロップにより行います。

備考 1. パス名にマウス・カーソルをあわせると，そのパスを絶対パスでポップアップ表示します。
備考 2. 新規に追加したシステム・インクルード・パスは，一覧の最後のパスの次に追加します。
備考 3. パス名をドラッグ・アンド・ドロップする際，連続して並んでいるパス名のみ複数選択することができます。

(b) ボタン

↑	選択しているパスを上へ移動します。
↓	選択しているパスを下へ移動します。

備考 上記のボタンは，パスを選択していない場合は無効となります。

[機能ボタン]

ボタン	機能
OK	コンパイラへのパスの指定順を パス一覧表示エリア の表示順に設定し、本ダイアログをクローズします。
キャンセル	パスの指定順の設定をキャンセルし、本ダイアログをクローズします。
ヘルプ	本ダイアログのヘルプを表示します。

ファイルに出力するモジュールの選択 ダイアログ

I/O ヘッダ・ファイルに出力するモジュールの設定を行います。

図 A.3 ファイルに出力するモジュールの選択 ダイアログ



ここでは、次の項目について説明します。

- [\[オープン方法\]](#)
- [\[各エリアの説明\]](#)
- [\[機能ボタン\]](#)

[オープン方法]

- プロパティ パネルにおいて、以下のプロパティを選択したのち、[...] ボタンをクリック
 - [\[I/O ヘッダ・ファイル生成オプション\]](#) タブの [\[I/O ヘッダ・ファイル\]](#) カテゴリの [\[デバイス・ファイルで定義されているモジュール一覧\]](#)

[各エリアの説明]

- (1) モジュール名／ファイル名指定エリア
デバイス・ファイルで定義されているモジュール名と出力先の I/O ヘッダ・ファイル名の一覧を表示します。
 - (a) [モジュール名]
デバイス・ファイルで定義されているモジュール名を表示します。
チェック・ボックスを選択すると、[ファイル名] に表示している I/O ヘッダ・ファイルに出力します。
デフォルトではチェック状態です。
 - (b) [ファイル名]
出力先の I/O ヘッダ・ファイル名を表示します。
ファイル名を直接入力することにより、出力先の I/O ヘッダ・ファイルを変更します。
デフォルトのファイル名は、"iodefine.h" です。
 - (c) [全選択／全解除]
[モジュール名] のチェック・ボックスを全選択／全解除します。

チェック・ボックスを選択すると、[モジュール名] のチェック・ボックスをすべて選択します。
チェック・ボックスのチェックを外すと、[モジュール名] のチェック・ボックスのチェックをすべて外します。

備考 複数のモジュールで同じファイル名を指定した場合は、1つのファイルに複数モジュールが出力されます。その際、ファイル名の大文字、小文字は区別されません。

[機能ボタン]

ボタン	機能
OK	本ダイアログをクローズし、設定内容を読み出し元のプロパティに反映します。
設定を初期化	すべての [モジュール名] のチェック・ボックスと [ファイル名] をデフォルト値に設定します。
キャンセル	設定を無効とし、本ダイアログをクローズします。
ヘルプ	本ダイアログのヘルプを表示します。

改訂記録

Rev.	発行日	改定内容	
		ページ	ポイント
1.00	2020.11.01	-	初版発行

CS+ V8.05.00 ユーザーズマニュアル
GHS CCRH850 ビルド・ツール操作編

発行年月日 2020年 11月 1日 Rev.1.00

発行 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）

CS+ V8.05.00



ルネサスエレクトロニクス株式会社

R20UT4853JJ0100