

# IDEU8

## ユーザーズマニュアル

---

プログラム開発支援ソフトウェア

## ご注意

本資料の一部または全部をラピスセミコンダクタの許可なく、転載・複写することを堅くお断りします。

本資料の記載内容は改良などのため予告なく変更することがあります。

本資料に記載されている内容は製品のご紹介資料です。ご使用にあたりましては、別途仕様書を必ずご請求のうえ、ご確認ください。

本資料に記載されております応用回路例やその定数などの情報につきましては、本製品の標準的な動作や使い方を説明するものです。したがって、量産設計をされる場合には、外部諸条件を考慮していただきますようお願いいたします。

本資料に記載されております情報は、正確を期すため慎重に作成したのですが、万が一、当該情報の誤り・誤植に起因する損害がお客様に生じた場合においても、ラピスセミコンダクタはその責任を負うものではありません。

本資料に記載されております技術情報は、製品の代表的動作および応用回路例などを示したものであり、ラピスセミコンダクタまたは他社の知的財産権その他のあらゆる権利について明示的にも黙示的にも、その実施または利用を許諾するものではありません。上記技術情報の使用に起因して紛争が発生した場合、ラピスセミコンダクタはその責任を負うものではありません。

本資料に掲載されております製品は、一般的な電子機器（AV 機器、OA 機器、通信機器、家電製品、アミューズメント機器など）への使用を意図しています。

本資料に掲載されております製品は、「耐放射線設計」はなされていません。

ラピスセミコンダクタは常に品質・信頼性の向上に取り組んでおりますが、種々の要因で故障することもあり得ます。

ラピスセミコンダクタ製品が故障した際、その影響により人身事故、火災損害等が起こらないようご使用機器でのデバッグ、冗長設計、延焼防止、フェイルセーフ等の安全確保をお願いします。定格を超えたご使用や使用上の注意書が守られていない場合、いかなる責任もラピスセミコンダクタは負うものではありません。

極めて高度な信頼性が要求され、その製品の故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのある機器・装置・システム（医療機器、輸送機器、航空宇宙機、原子力制御、燃料制御、各種安全装置など）へのご使用を意図して設計・製造されたものではありません。上記特定用途に使用された場合、いかなる責任もラピスセミコンダクタは負うものではありません。上記特定用途への使用を検討される際は、事前にローム営業窓口までご相談願います。

本資料に記載されております製品および技術のうち「外国為替及び外国貿易法」に該当する製品または技術を輸出する場合、または国外に提供する場合には、同法に基づく許可が必要です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標です。また、その他の製品名や社名などは、一般に商標または登録商標です。

---

## ラピスセミコンダクタ株式会社

〒193-8550 東京都八王子市東浅川町 550 番地 1

<http://www.lapis-semi.com/jp/>

# 目 次

<b>1 章 イントロダクション</b>	<b>1</b>
1.1 製品について	1
1.1.1 概要	1
1.1.2 特徴	1
1.2 マニュアルの構成	2
1.3 表記法	3
<b>2 章 インストール</b>	<b>4</b>
2.1 システム構成	4
2.2 IDEU8 のインストール	4
<b>3 章 チュートリアル</b>	<b>5</b>
3.1 プロジェクトの作成	5
3.1.1 プロジェクトの準備	5
3.1.2 プロジェクトへの登録	6
3.2 オプション設定	7
3.2.1 コンパイル／アセンブルオプション	8
3.2.2 ターゲットオプション	8
3.3 ビルド作業	8
3.4 エラー発生箇所へのジャンプ	9
3.5 プログラムのデバッグ	9
<b>4 章 プロジェクト管理</b>	<b>10</b>
4.1 プロジェクトの作成／保存／読み込み	10
4.1.1 プロジェクトの作成	10
4.1.2 プロジェクトの保存	10
4.1.3 プロジェクトの読み込み	11
4.2 プロジェクトウィンドウ	11
4.2.1 プロジェクトのフォルダツリー表示	11
4.2.2 ポップアップメニュー	12
4.3 ファイル管理	14
4.3.1 プロジェクトへの登録	14

4.3.2	プロジェクトからの削除.....	15
4.3.3	ソースの設定 .....	15
4.3.4	リンクの順序 .....	16
<b>4.4</b>	<b>コンパイル／アセンブルオプション設定.....</b>	<b>17</b>
4.4.1	一般タブ .....	17
4.4.2	ディレクトリタブ .....	18
4.4.3	最適化タブ .....	20
4.4.4	コード生成タブ .....	21
4.4.5	警告タブ .....	22
4.4.6	リスティングタブ .....	23
4.4.7	メモリ設定タブ .....	25
4.4.8	マクロタブ .....	26
<b>4.5</b>	<b>ターゲットオプション設定 .....</b>	<b>27</b>
4.5.1	一般タブ .....	27
4.5.2	セグメントタブ .....	29
4.5.3	メモリ設定タブ .....	31
<b>4.6</b>	<b>オプションセットの定義.....</b>	<b>33</b>
<b>5</b>	<b>章 プログラムのビルド.....</b>	<b>35</b>
5.1	ビルド処理の概要 .....	35
5.2	ビルド .....	35
5.3	リビルド.....	35
5.4	選択ファイルのビルド.....	36
5.5	ビルドの中止.....	36
5.6	クリーン .....	36
5.7	デバッグ .....	36
5.7.1	プログラムの修正 .....	36
5.7.2	IDEU8 と DTU8 デバッグ間のリンク.....	36
<b>5.8</b>	<b>アウトプットウィンドウ .....</b>	<b>36</b>
5.8.1	メッセージのコピー .....	37
5.8.2	エラー発生ソース行へジャンプ .....	37
5.8.3	メッセージの保存 .....	37
5.8.4	メッセージのクリア .....	37
5.8.5	ドッキングビュー .....	37
<b>6</b>	<b>章 その他の機能.....</b>	<b>38</b>

6.1	各種ファイルの新規作成.....	38
6.2	エディタウィンドウ .....	39
6.2.1	編集メニュー .....	39
6.3	システムパスの設定.....	41
6.4	その他の環境設定.....	42
6.5	ファイルの印刷.....	44
6.6	オンラインヘルプ .....	44
6.7	バージョン情報の表示.....	45

# 1章 イントロダクション

---

## 1.1 製品について

### 1.1.1 概要

統合化開発環境 IDEU8（以下、IDEU8）は、弊社オリジナル 8 ビット RISC プロセッサ nX-U8 コアを搭載したマイクロコントローラ用プログラム開発のための作業環境を提供するためのソフトウェアツールです。

IDEU8 は、CCU8 コンパイラパッケージ、MACU8 アセンブラパッケージおよび DTU8 デバッガ等の開発支援ソフトウェアを 1 つに統合し、さらに、プロジェクト管理機能、プログラミングエディタなどの開発効率を向上させるための機能を持ちます。従来まで、手作業で行っていたメイクファイルやバッチファイルの作成も、IDEU8 では必要ありません。

IDEU8 の環境下では、エディタによるプログラム作成からコンパイル、リンクに至るプログラム開発工程や、デバッガを使用したプログラムデバッグ工程において、従来個別に使用してきた開発支援ソフトウェアツールを利用できます。<sup>1</sup>

### 1.1.2 特徴

#### ■ 32 ビット Windows 対応

IDEU8 を使用することにより、現代的な GUI（Graphical User Interface）環境でのプログラム開発作業が可能です。

32 ビット Windows アプリケーションとして作成され、快適な動作環境を提供します。

#### ■ プロジェクト管理

ソースファイル間の依存関係やコンパイラ、アセンブラのオプション設定等をプロジェクトとして一括管理します。

プロジェクトウィンドウでは、各種ソースファイルやライブラリファイルをプロジェクトへ追加／削除することができ、登録されたファイル群は、ツリー形式でグラフィカルに表示されます。

さらに、プロジェクトに、ドキュメントファイルを登録することにより、関連情報をプロジェクトに関連付けることができます。

#### ■ ダイアログボックスによる容易なオプション設定

コマンドラインでは複雑かつ面倒なオプション設定も、IDEU8 では分かりやすいダイアログボックスによってに設定することができます。

オプションは、複数のオプションセットを登録し、ビルド単位で割り付けることが可能です。

---

<sup>1</sup> MACU8 アセンブラパッケージの LIBU8 ライブラリアンを除きます。

## 1章 イントロダクション

### ■ 内蔵エディタによるプログラムの表示／編集

コンパイル、アセンブル時にエラーが発生した場合、対象となるソース行を IDEU8 に内蔵されているプログラミングエディタで編集することができます。さらに、ビルド時のエラー発生行へのタグジャンプにも対応しています。

### ■ ヘルプ機能

選択中のメニューの簡単な紹介がステータスラインに表示されるクイックヘルプ機能と、画面上で機能や使い方を説明するオンラインヘルプ機能があります。

## 1.2 マニュアルの構成

IDEU8 と共に使用する開発支援ソフトウェアの詳細については、以下のユーザーズマニュアルを参照してください。

### ■ CCU8 コンパイラパッケージ

CCU8 コンパイラ ユーザーズマニュアル

CCU8 コンパイラ プログラミングガイド

RTL8 ランタイムライブラリ ユーザーズマニュアル

### ■ MACU8 アセンブラパッケージ

MACU8 アセンブラパッケージ ユーザーズマニュアル

### ■ DTU8 デバッガ

DTU8 デバッガ ユーザーズマニュアル

このマニュアルの各章の概要は、以下のとおりです。

### ■ 1章 イントロダクション

この章です。

### ■ 2章 インストール

IDEU8 を使用するために必要な環境とインストール方法について説明しています。

### ■ 3章 チュートリアル

IDEU8 の起動から、プロジェクトの作成、コンパイラオプションの設定、プログラムのデバッグに至る一連の作業の流れを簡単に説明します。

### ■ 4章 プロジェクト管理

プロジェクトウィンドウの詳細と、ソースファイルの追加／削除方法、およびプロジェクトファイルの保存／読み込み等について説明しています。

### ■ 5章 プログラムのビルド

プロジェクトのビルド機能とビルド結果が表示されるアウトプットウィンドウについて説明しています。

### ■ 6章 その他の機能

エディタウィンドウや、コンパイラ、アセンブラ、リンカおよびデバッガのパス設定やオンラ

インヘルプなど、その他の機能の使い方を説明しています。

### ■ 索引

辞書順の索引です。特に、各種ビルドツールを指定する箇所を調べる場合に便利です。

## 1.3 表記法

キー操作は、キーの刻印をカギ括弧で囲み、同時に押すキーがあれば、プラス(+)で連結しています。例えば、文章中に[Shift]+[F1]と記述されているときには、Shift キーを押しながら、F1 キーを押すことを表わしています。

その他には、特に説明を要するアイコンや表記を使用していません。



## 2章 インストール

---

### 2.1 システム構成

IDEU8 を使用するために必要な環境は、以下の通りです。

#### ■ 動作環境

- ・ WindowsXP/Vista\*/7\*が動作しているパソコン
  - ・ \*32bit 版/64bit 版
- ・ 上記オペレーティングシステムが、快適に動作する環境
- ・ SVGA (800×600)以上のグラフィックアダプタとディスプレイ
- ・ 20MB 以上のハードディスクの空き容量
- ・ CD-ROM ドライブ
- ・ マウス

### 2.2 IDEU8 のインストール

IDEU8 のセットアッププログラムには、IDEU8 本体を構成するプログラムの他に、ビルドで使用する CCU8 コンパイラ等のソフトウェアやデバッグで使用する DTU8 デバッガが含まれています。従って、IDEU8 をインストールすれば IDEU8 に統合される各種ソフトウェア群も併せてインストールされます。

#### ■ セットアップ CD からのインストール

1. nX-U8 開発ツールのセットアップ CD を CD-ROM ドライブにセットします。
2. Windows のオートラン機構で、立ち上がるセットアッププログラムの指示に従ってください。もしセットアッププログラムが立ち上がらない場合には、CD のルートにある Setup.exe を起動してください。
3. 以後は、画面の指示に従ってください。

#### ■ アンインストール

コントロールパネル中の[アプリケーションの追加と削除]で、[nX-U8 開発ツール]を選択することにより、IDEU8 および統合されている各種ソフトウェアをアンインストールできます。

## 3章 チュートリアル

### 3.1 プロジェクトの作成

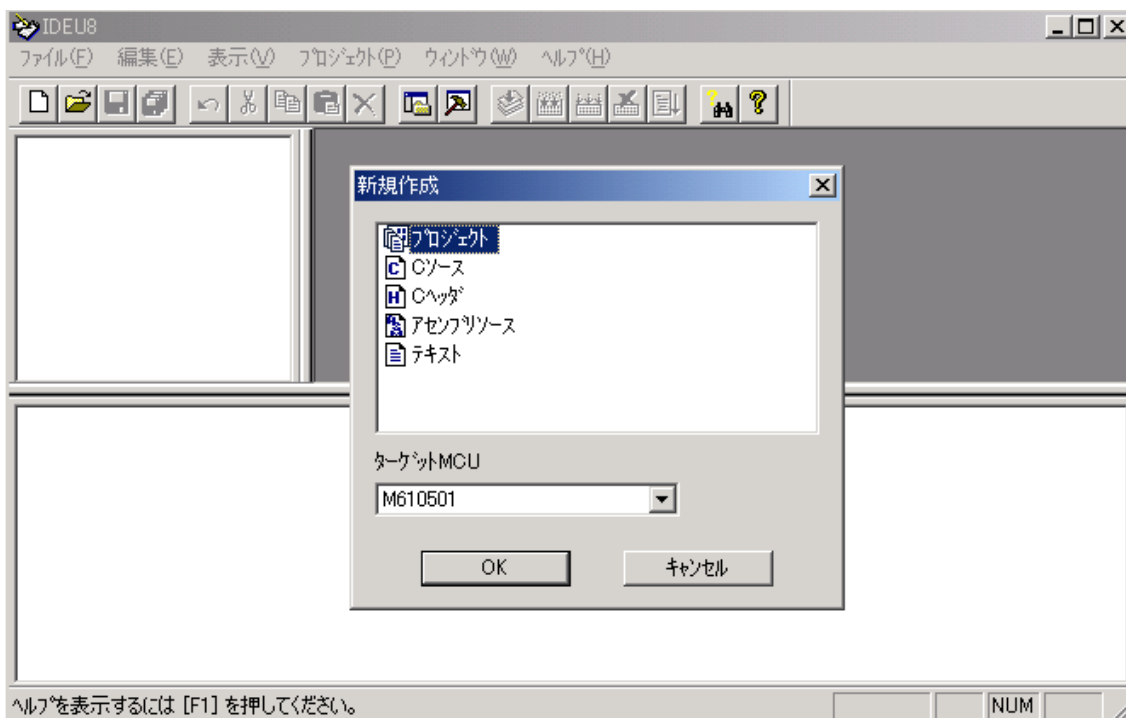
IDEU8 で作業を行うには、まず、IDEU8 のプロジェクトファイルを作成する必要があります。

このプロジェクトは、実行ファイルをビルド(コンパイル、アセンブル、リンクまでの一連の処理)するために必要な以下のような情報を保持しています。

- ・ 実行プログラムをビルドするために必要なソースファイル、ライブラリファイルのリスト
- ・ 実行プログラムを構築するために必要なビルドツールの起動オプション
- ・ ビルド処理時に生成するソースとインクルードファイルの依存関係のデータベース

#### 3.1.1 プロジェクトの準備

[ファイル]メニューの [新規作成...]を選択すると、次のようなダイアログボックスが表示されます。



ダイアログボックスの上部のリストから[プロジェクト]を選択し、[ターゲット MCU]から使用するマイコンの機種名を選択してください。

### 3章 チュートリアル

[OK]ボタンを押すと、[プロジェクトの新規作成]のファイルダイアログボックスが開き、プロジェクトファイル（.PID）を指定することができます。

ここで作成するプロジェクトファイルの名前のベース名は、最終ターゲット（ビルド処理を行って構築する実行プログラム）ファイルのベース名になります。例えば、Skeleton.pid という名前のプロジェクトファイルを指定すると、最終ターゲットのファイル名は、Skeleton.abs や Skeleton.hex などとなります。

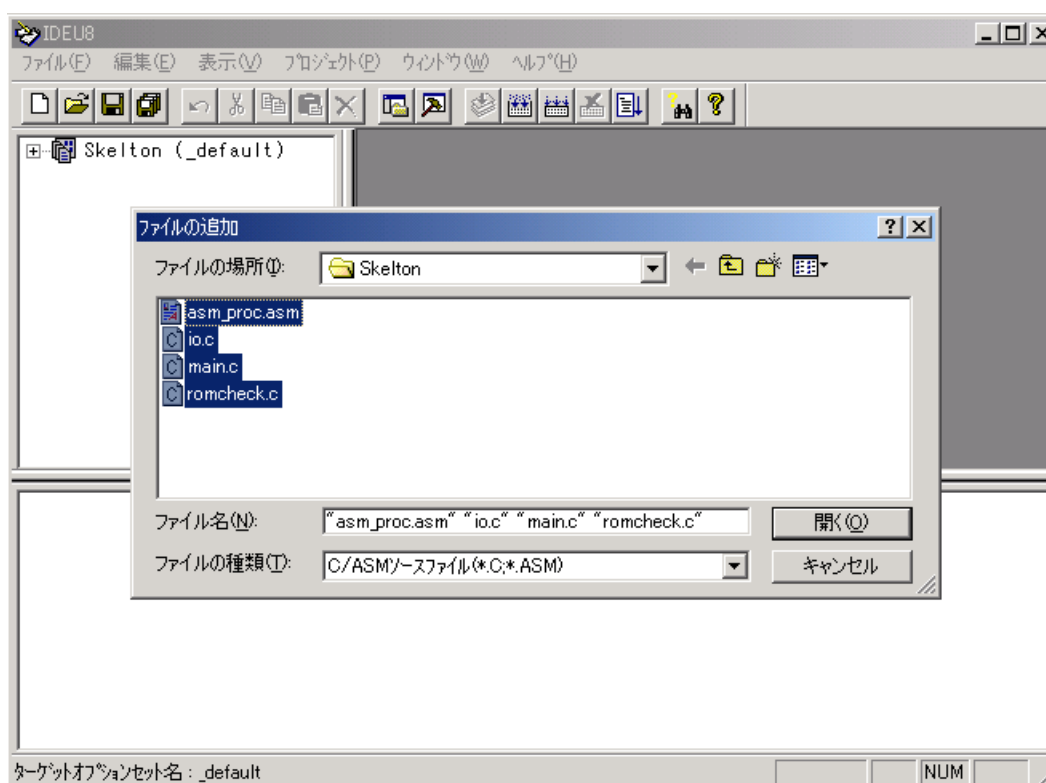
なお、既存の IDEU8 のプロジェクトを開くには、[ファイル]メニューの[開く]を指定してください。

#### 3.1.2 プロジェクトへの登録

IDEU8 の内蔵エディタを使用してソースファイルを新規に作成するには、[ファイル]メニューの[新規作成...]選択し、作成したいファイルの種類を選んでください。[C ソース]を選択すると、拡張子が.cである新規ファイルを作成するためのエディタウィンドウを開きます。

IDEU8 の内蔵エディタで作成したソースファイルや既存のソースファイルを、IDEU8 でビルドするためには、プロジェクトに登録する必要があります。

[プロジェクト]メニューの [ファイルを追加...]を選択すると、次のようなダイアログボックスが表示されます。



IDEU8 に登録する必要があるファイルは、ビルドに必要なソースファイルとライブラリファイルで、拡張子で判断されます。

- ・ .C        コンパイルの対象になる C ソースファイル
- ・ .ASM      アセンブルの対象になるアセンブリ言語ソースファイル
- ・ .OBJ      リnkの対象になるオブジェクトファイル
- ・ .LIB      リnkの対象になるライブラリオブジェクトファイル

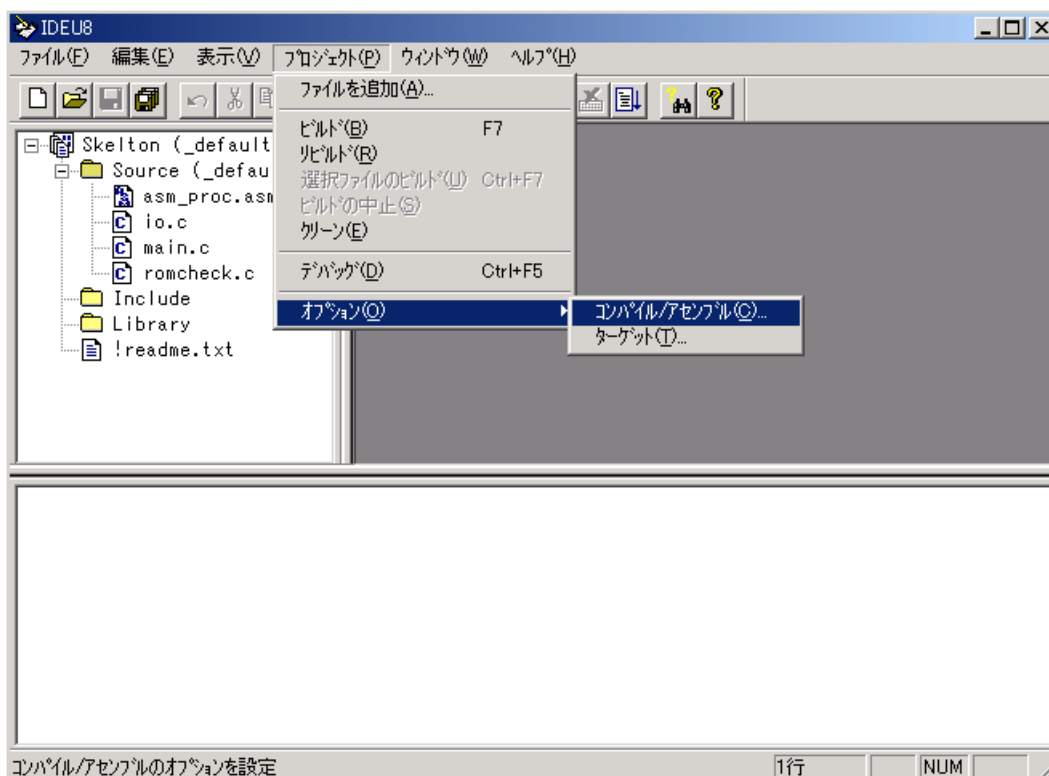
.C と .ASM のソースファイルは、プロジェクトウィンドウの Source フォルダに登録されます。  
 .OBJ と .LIB のオブジェクトファイルは、プロジェクトウィンドウの Library フォルダに登録されます。

プロジェクトウィンドウの Include フォルダは、IDEU8 が自動的にメンテナンスします。IDEU8 は、ビルド処理中に、コンパイル、アセンブルしたソースファイルがインクルードしているファイルを検出し、Include フォルダに登録します。

なお、.C、.ASM、.OBJ、.LIB 以外の拡張子を持つファイルは、プロジェクトウィンドウのトップにあるフォルダ(プロジェクトフォルダ)に登録されます。プロジェクトに関するドキュメントのテキストファイルや、その他のアプリケーションのデータファイルを登録しておくことができます。

## 3.2 オプション設定

CCU8 コンパイラや RASU8 アセンブラ、RLU8 リンカなどのビルドツールには、実行プログラムを構築するためのさまざまなオプションが用意されています。これらのオプションは、IDEU8 を使用することにより、ダイアログボックスで容易に設定できます。



### 3.2.1 コンパイル／アセンブルオプション

[プロジェクト]メニューの[オプション]→[コンパイル／アセンブル...]で開くダイアログボックスで、各種のビルドツールのオプションのうち、中間オブジェクトファイルの生成に関する設定、すなわち CCU8 コンパイラおよび RASU8 アセンブラのオプション設定を行います。

オプション設定の詳細については、「4.5 コンパイル/アセンブルオプション設定」を参照してください。

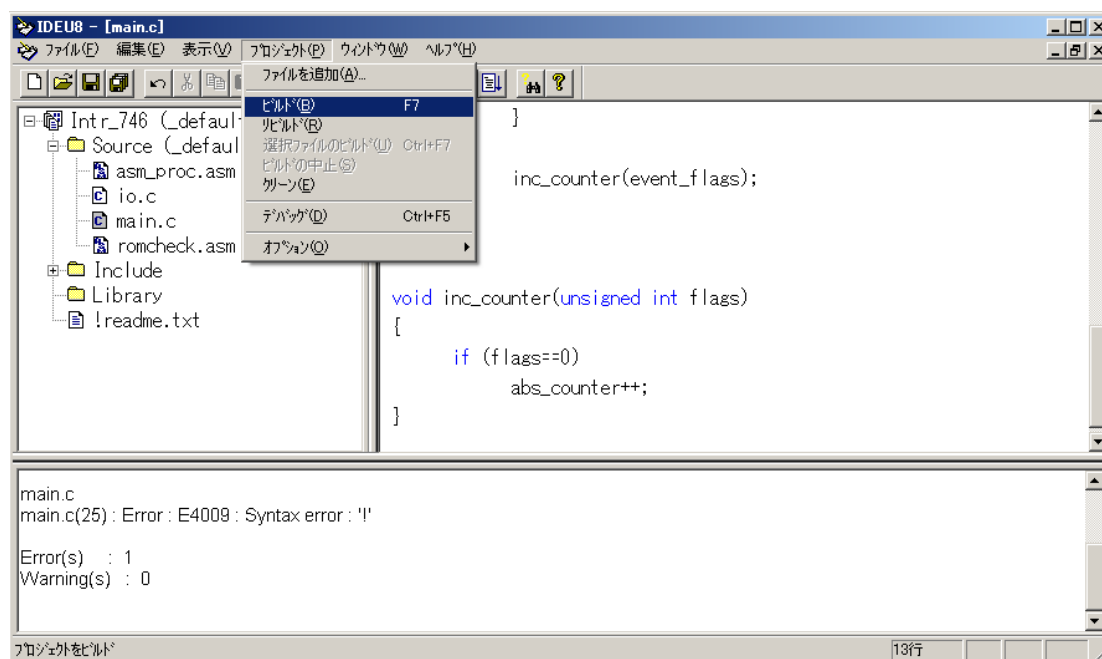
### 3.2.2 ターゲットオプション

[プロジェクト]メニューの[オプション]→[ターゲット...]で開くダイアログボックスで、各種のビルドツールのオプションのうちターゲットファイルの生成に関する設定、すなわち RLU8 リンカおよび OHU8 オブジェクトコンバータのオプション設定を行います。

オプション設定の詳細については、「4.6 ターゲットオプション設定」を参照してください。

## 3.3 ビルド作業

[プロジェクト]メニューの[ビルド]を指定すると、プロジェクトに登録したCソースファイル、アセンブリ言語ソースファイル、各種ライブラリファイルのそれぞれのファイルに必要な処理を行い、最終的に実行ファイルを作成することを「ビルド」と呼んでいます。



ビルド処理の基本は、[プロジェクト]メニューの[ビルド]を指定することです。この指定は、初めてのビルド処理であれば、全てのソースファイルをコンパイルおよびアセンブルし、そのオブジェクトファイルをリンクして、最終ターゲットファイルを作成します。2回目以降のビルド処理は、ファイルのタイムスタンプをチェックして、前回のビルド時からの変更されたファイルを検出して、最終ターゲットファイルを作成するために必要な最小限の処理を行います。前回の

ビルド処理以降に修正されたソースファイルまたは修正されたヘッダファイルをインクルードしているソースファイルだけを、コンパイル、アセンブルして、そのオブジェクトファイルをリンクします。

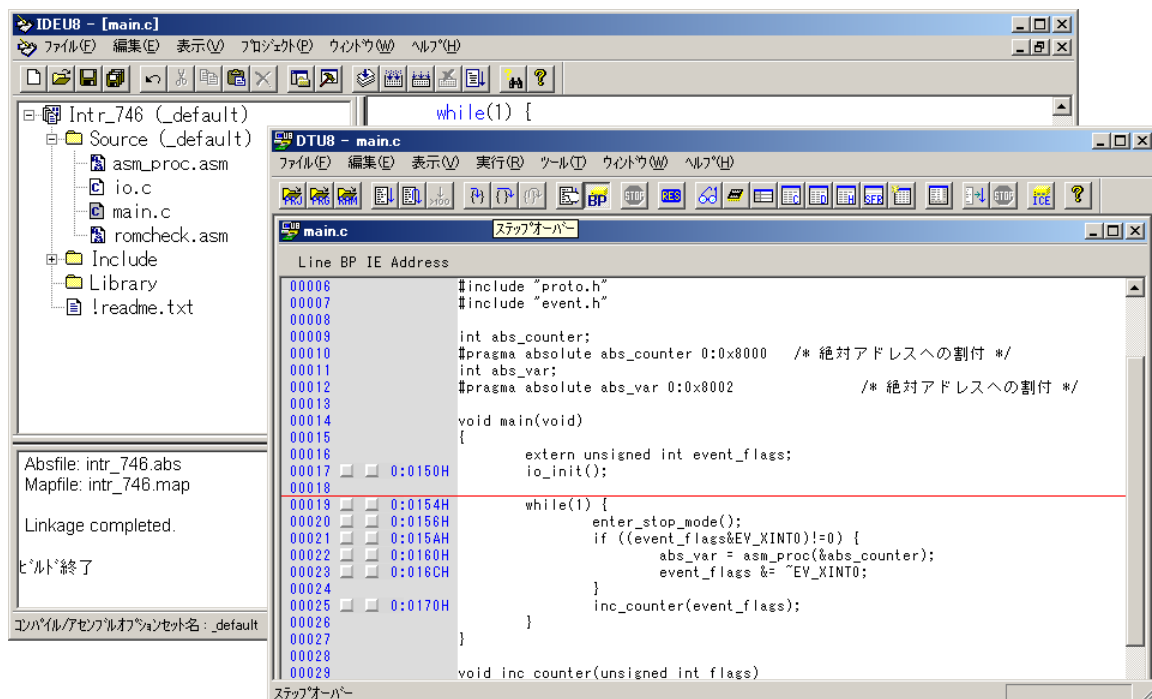
### 3.4 エラー発生箇所へのジャンプ

アウトプットウィンドウには、ビルド処理の経過が表示されます。

ビルド処理で行われたコンパイルやアセンブルによって、エラーや警告が発生した場合は、アウトプットウィンドウ上のエラーや警告メッセージにカーソルを移動し、マウスでダブルクリックするか、[Enter]キーを押してください。すると、エディタウィンドウにエラー発生ソースを開き、エラー発生箇所カーソルを移動します。

### 3.5 プログラムのデバッグ

[プロジェクト]メニューの[デバッグ]を指定すると、IDEU8 から DTU8 デバッガを起動することができます。IDEU8 から DTU8 デバッガを起動する場合、まず、ビルド処理を行い、その結果生成されたアブソリュートオブジェクトファイル(.ABS)を DTU8 デバッガに引き渡します。IDEU8 から起動された DTU8 デバッガは、エミュレータに接続して、IDEU8 から引き渡されたプログラムコードをエミュレータにダウンロードして、デバッグ作業開始までの一連の準備を自動的に行います。



デバッグ中にプログラムを修正する場合は、DTU8 デバッガ起動したまま、IDEU8 へ画面を切り替え、IDEU8 上でプログラムを修正しビルドします。再度 DTU8 デバッガに画面を戻すと、DTU8 デバッガは、ダウンロードされているオブジェクトが更新されていることを検知します。

## 4章 プロジェクト管理

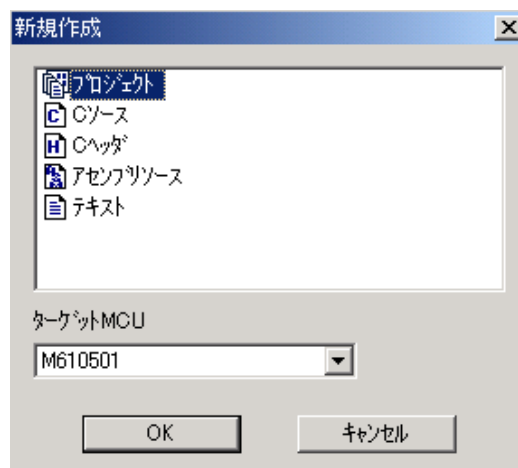
---

### 4.1 プロジェクトの作成／保存／読み込み

IDEU8 の起動直後は、プロジェクトを開いていない状態で立ち上がります。

#### 4.1.1 プロジェクトの作成

[ファイル]メニューの [新規作成...] を選択すると、次のようなダイアログボックスが表示されます。



リストから[プロジェクト]を選択し、ターゲット MCU を設定して[OK]ボタンを押すと、新規にプロジェクトファイルを作成することができます。作成したプロジェクトの名前が、アブソリュートオブジェクトファイルやマップファイル等の名称になります。

ターゲット MCU で設定した名称は、CCU8 コンパイラの/T オプションに指定されます。

プロジェクトの新規作成が終了すると、[プロジェクト]メニューの[ファイルの追加...]とプロジェクトウィンドウのポップアップメニューの[プロジェクトに追加...]メニューが有効となり、ソースファイルを登録できるようになります。

#### 4.1.2 プロジェクトの保存

IDEU8 は、作成したプロジェクトをプロジェクトファイル(.PID)として保存することができます。プロジェクトを保存する場合は、以下の操作を行います。

- ・ [ファイル]メニューの[プロジェクトの保存]を選択。
- ・ プロジェクトウィンドウのプロジェクトフォルダ(最上位のフォルダ)を選択して、[ファイル]メニューの[上書き保存]を選択。
- ・ プロジェクトウィンドウのプロジェクトフォルダ(最上位のフォルダ)を選択して、[ファイ

ル]メニューの[名前を付けて保存...]を選択。この場合、プロジェクトを別名で保存できます。<sup>2</sup>

### 4.1.3 プロジェクトの読み込み

IDEU8 は、保存されているプロジェクトファイルを読み込むことで、プロジェクトを開くことができます。

保存されているプロジェクトファイルを IDEU8 に読み込む場合は、[ファイル]メニューの[開く...]を選択し、表示されるダイアログボックスの[ファイルの種類]をプロジェクトファイル (\*.PID) に設定します。

IDEU8 は、一度に、ひとつのプロジェクトしか開けません、プロジェクトを開こうとしたときに、既に開かれているプロジェクトファイルに変更があれば、次のようなメッセージが表示され、閉じようとするプロジェクトの保存の確認を行います。



## 4.2 プロジェクトウィンドウ

プロジェクトウィンドウは、[表示]メニューの[プロジェクト]を指定するか、[Alt]+[0(ゼロ)]キーで表示します。

IDEU8 がプロジェクトを開いていない場合には、プロジェクトウィンドウは、空白になっています。

プロジェクトを新規に作成するか、プロジェクトファイルを読み込むと、プロジェクトウィンドウにフォルダツリー形式が現れます。

### 4.2.1 プロジェクトのフォルダツリー表示

プロジェクトウィンドウは、ツリー構造のフォルダとして、表現されています。最上位にプロジェクトフォルダ、その下の階層に”Source”、”Include”、”Library”の 3 つのフォルダがあります。

フォルダ名	拡張子	説明
プロジェクトフォルダ (ツリーの最上位)	(表示しません)	プロジェクトとターゲットの名称を示します。
Source フォルダ	.C と .ASM	ビルド時に、コンパイル、または、アセンブルの対象になるファイルです。
Include フォルダ	(任意)	ビルド時にソースファイルが依存しているファイルで、IDEU8 が登録します。
Library フォルダ	.OBJ と .LIB	ビルド時に、リンクされるファイルです。

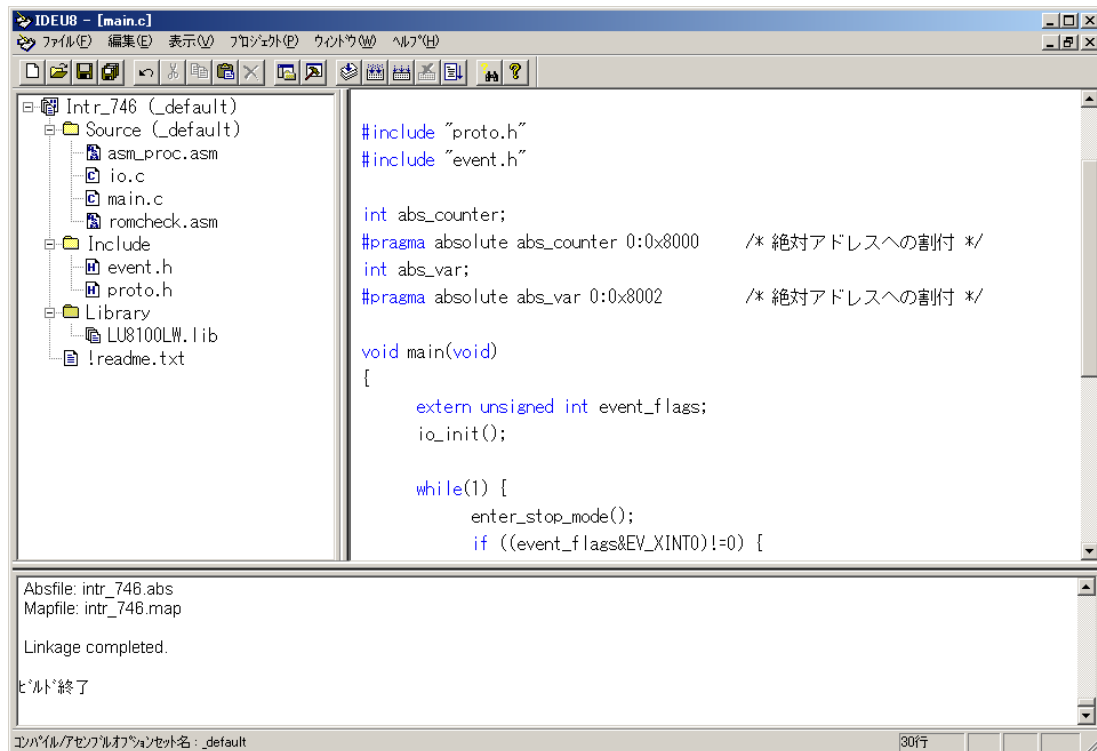
<sup>2</sup> プロジェクトの名前を変更すると、ターゲットファイルの名前も変わります。このため、次のビルドは、少なくとも、変更した名前になるようなリンク処理が行われます。



## 4章 プロジェクト管理

プロジェクトへの追加時に、拡張子が、.C、.ASM、.OBJ、.LIB 以外のファイル名を登録すると、プロジェクトフォルダのすぐ下に登録されます。

各フォルダの左側の+ボタンをマウスでクリックすると、フォルダ内のファイル群を展開して表示します。このとき、フォルダの左側の+ボタンは、-ボタンに変化します。



Source フォルダに登録されているファイルは、登録順序に関係なく、辞書順にソートされ、Source フォルダに表示されている順序で、ビルド処理が行われます。

Include フォルダには、ソースファイルをビルドしたときに検出する依存ファイル(インクルードファイル)を、辞書順にソートして、表示しています。なお、システムインクルードパス([ファイル]メニューの[環境設定→システムパス]で開くダイアログボックスの[システムインクルードパス]で指定するパス)にあるファイルの表示/非表示を切り替えることができます。設定方法は、「6.4 その他の機能」を参照してください。

Library フォルダの表示については、「4.3.4 リンクの順序」を参照してください。

### 4.2.2 ポップアップメニュー

プロジェクトウィンドウは、ポップアップメニューを持っています。

プロジェクトウィンドウのポップアップメニューを開くには、以下の操作を行います。

- ・ プロジェクトウィンドウ内の対象のファイルやフォルダをマウスで右クリック
- ・ プロジェクトウィンドウ内の対象のファイルやフォルダを選択して[Shift]+[F10]

## ■ ビルド

選択中のファイルをビルドします。Source フォルダ内のソースファイルであれば、指定したソースのみのビルドになり、プロジェクトフォルダであれば、最終ターゲットファイルを作成するまでのビルド処理を行います。

## ■ 開く

Source フォルダ、Include フォルダの中のファイルであれば、エディタウィンドウに表示します。

プロジェクトフォルダの直下にあるファイルの場合、拡張子に関連付けられたアプリケーションがあれば、そのアプリケーションを起動します。<sup>3</sup>

## ■ プロジェクトに追加...

プロジェクトにファイルを追加します。[プロジェクト]メニューの[ファイルの追加]と同じです。

## ■ プロジェクトから削除

選択したファイルをプロジェクトから削除します。

## ■ 設定

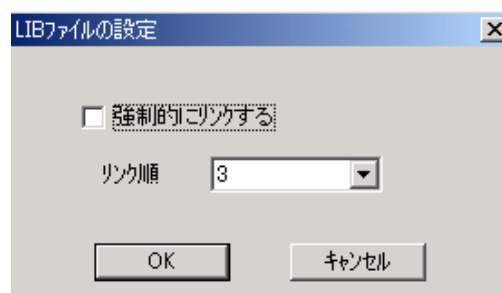
Source フォルダが選択されている場合、ソースファイルすべてに割り当てるデフォルトのコンパイル/アセンブルオプションのオプションセットを選択します。

Source フォルダのファイルが選択されている場合、ソースファイル個々に割り当てるコンパイル/アセンブルオプションのオプションセットを選択します。

プロジェクトフォルダが選択されている場合、ターゲットオプションのオプションセットを選択します。

オプションセットについては、「4.6 オプションセット」で説明します。

Library フォルダの中の.LIB ファイルを選択していたら、ライブラリファイルのリンク制御のためのダイアログボックスが表示されます。このダイアログボックスの使用方法については、「4.3.4 リンクの順序」を参照してください。



## ■ ドッキングビュー

[ドッキングビュー]がチェックされているときには、プロジェクトウィンドウを IDEU8 のメインウィンドウの左側にドッキングして表示します。このチェックを外すと、プロジェクトウィンドウを、メインウィンドウから独立したウィンドウとして表示します。

<sup>3</sup> 拡張子.TXTと.LOGは、常に、IDEU8のエディタウィンドウに開きます。

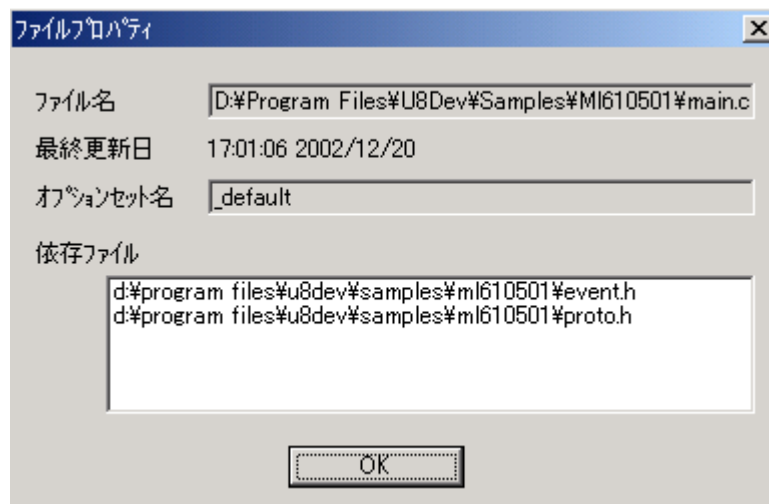
## 4章 プロジェクト管理

### ■ 非表示

プロジェクトウィンドウを非表示にします。プロジェクトは閉じません。

### ■ プロパティ...

選択ファイルのプロパティを表示します。本メニューを選択すると、次のようなダイアログボックスが表示されます。



「[ファイル名]フィールドは、選択したファイルの置かれている絶対パスを表示しています。プロジェクトフォルダが選択中であれば、プロジェクトファイル(.PID)の情報を表示します。

「[最終更新日]フィールドは、選択したファイルが更新された時刻と年月日(ファイルのタイムスタンプ)を表示します。

「[オプションセット名]フィールドは、選択したファイルまたはプロジェクトに割り当てられているオプションセットを表示します。

「[依存ファイル]フィールドは、ビルド処理の判断に参照されるファイルの一覧を表示しています。Source フォルダの下ソースファイルの場合には、インクルードしているファイルを表示することになります。プロジェクトフォルダの場合には、リンク処理でリンクするオブジェクトファイルとライブラリファイルの一覧になります。

## 4.3 ファイル管理

IDEU8 は、ターゲット MCU やプロジェクトウィンドウに登録した各種ソースファイル、コンパイルオプション、各種ウィンドウの状態、環境設定等をプロジェクトファイルに保存します。

開発する製品毎にプロジェクトファイルを用意することにより、複数の製品をプロジェクト単位で管理することができます。

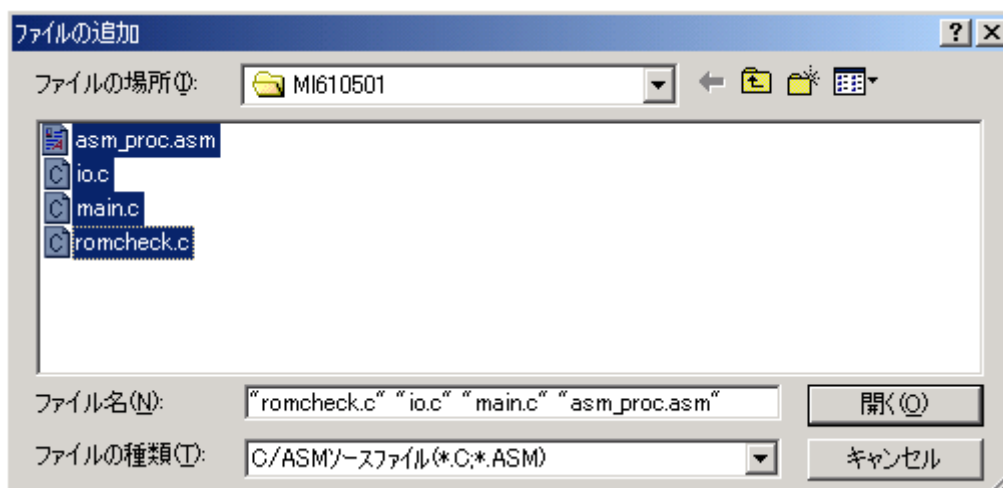
### 4.3.1 プロジェクトへの登録

プロジェクトを構成するソースファイル、オブジェクトファイル、ライブラリファイル、その他のドキュメントファイルを、プロジェクトに登録するには、以下の方法があります。

- ・ [プロジェクト]メニューの[ファイルの追加...]

- ・ プロジェクトウィンドウのポップアップメニューの[プロジェクトに追加]
- ・ プロジェクトウィンドウを選択して、[INS]キー

これらのメニューを実行すると、次のようなダイアログが表示され、登録したいファイルを選択できます。



この[ファイルの追加]ダイアログボックスは、同じフォルダ内の複数のファイルを同時に選択することができます。

選択したファイルの種類に応じて、プロジェクトウィンドウの Source、Library フォルダ、もしくは、プロジェクトウィンドウの直下にファイルが追加されます。

### 4.3.2 プロジェクトからの削除

プロジェクトに登録したファイルを削除する場合は、プロジェクトウィンドウ上で削除するファイルを選択し、ポップアップメニューの[プロジェクトから削除]を実行します。<sup>4</sup>

### 4.3.3 ソースの設定

IDEU8 では、プロジェクトに対するターゲットオプションセットとデフォルトのコンパイル／アセンブルオプションセット、ソースファイル毎のコンパイル／アセンブルオプションセットを定義することができます。

プロジェクトウィンドウで、プロジェクト名を選択してポップアップメニューの[設定]を実行すると、ターゲットオプションのオプションセットを指定することができます。指定したオプションセット名は、プロジェクトウィンドウのプロジェクト名に続けて表示されます。

「Source」フォルダを選択してポップアップメニューの[設定]を実行すると、ターゲットに対するデフォルトのコンパイル／アセンブルオプションのオプションセットをまとめて指定することができます。指定したオプションセット名は、プロジェクトウィンドウの「Source」フォルダ名に続けて表示されます。ソースファイルを選択してポップアップメニューの[設定]を実行すると、選択ファイルに対するコンパイル／アセンブルオプションのオプションセットを指定するこ

<sup>4</sup> この削除は、プロジェクトから登録情報を削除するだけであり、実際のディスク上のファイルの実体を消すわけではありません。

## 4章 プロジェクト管理

とができます。指定したオプションセットがデフォルトと違っている場合は、プロジェクトウィンドウのソースファイル名に続けて指定したオプションセット名が表示されます。また各ファイルの設定状態は、ポップアップメニューの[プロパティ...]でも確認できます。

### 4.3.4 リンクの順序

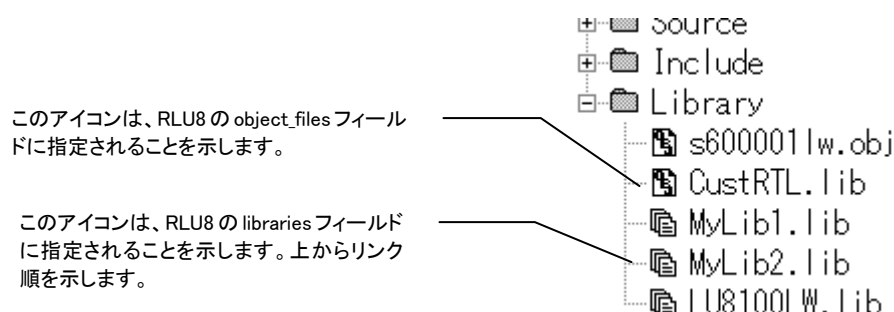
IDEU8 は、RLU8 リンカのobject\_filesフィールドに、以下のファイルを指定します。<sup>5</sup>

- ・ プロジェクトウィンドウのSource フォルダに登録されているソースをコンパイル/アセンブルして生成したオブジェクトファイル(.OBJ)
- ・ プロジェクトウィンドウの Libray フォルダに登録されているオブジェクトファイル(.OBJ)
- ・ プロジェクトウィンドウの Libray フォルダに登録されているライブラリファイル(.LIB)のうち、ポップアップメニューの[設定]で開く [LIB ファイルの設定] ダイアログボックスで、[強制的にリンクする]をチェックしたもの
- ・ [プロジェクト]メニューの[オプション→ターゲット]で開くダイアログボックスの[オブジェクトフィールドへ追加]に指定されたもの

IDEU8 は、RLU8 リンカのlibrariesフィールドに、以下のファイルを指定します。<sup>6</sup>

- ・ プロジェクトウィンドウの Libray フォルダに登録されているライブラリファイル(.LIB)のうち、ポップアップメニューの[設定]で開く [LIB ファイルの設定] ダイアログボックスで、[強制的にリンクする]をチェックしていないもの
- ・ [プロジェクト]メニューの[オプション→ターゲット]で開くダイアログボックスの[ライブラリフィールドへ追加]に指定されたもの

プロジェクトウィンドウのLibrayフォルダのライブラリファイルは、ポップアップメニューの[設定]で開く [LIBファイルの設定] ダイアログボックスで、順序の指定をできます。さらに、プロジェクトウィンドウのLibrayフォルダのライブラリファイルの後に、[プロジェクト]メニューの[オプション→ターゲット]で開くダイアログボックスの[ライブラリフィールドへ追加]の指定を続けます。<sup>7</sup>



<sup>5</sup> RLU8 リンカのobject\_filesフィールドは、RLU8 がリンクするファイルを列挙するためのフィールドです。詳細は、MACU8 ユーザーズマニュアルのRLU8 の説明を参照してください。

<sup>6</sup> RLU8 リンカのlibrariesフィールドは、RLU8 が未解決の外部参照を探すライブラリファイルを列挙するためのフィールドです。詳細は、MACU8 ユーザーズマニュアルのRLU8 の説明を参照してください。

<sup>7</sup> RLU8 リンカのlibrariesフィールドに指定するライブラリファイルの指定順序は、未解決参照を探す順序になります。

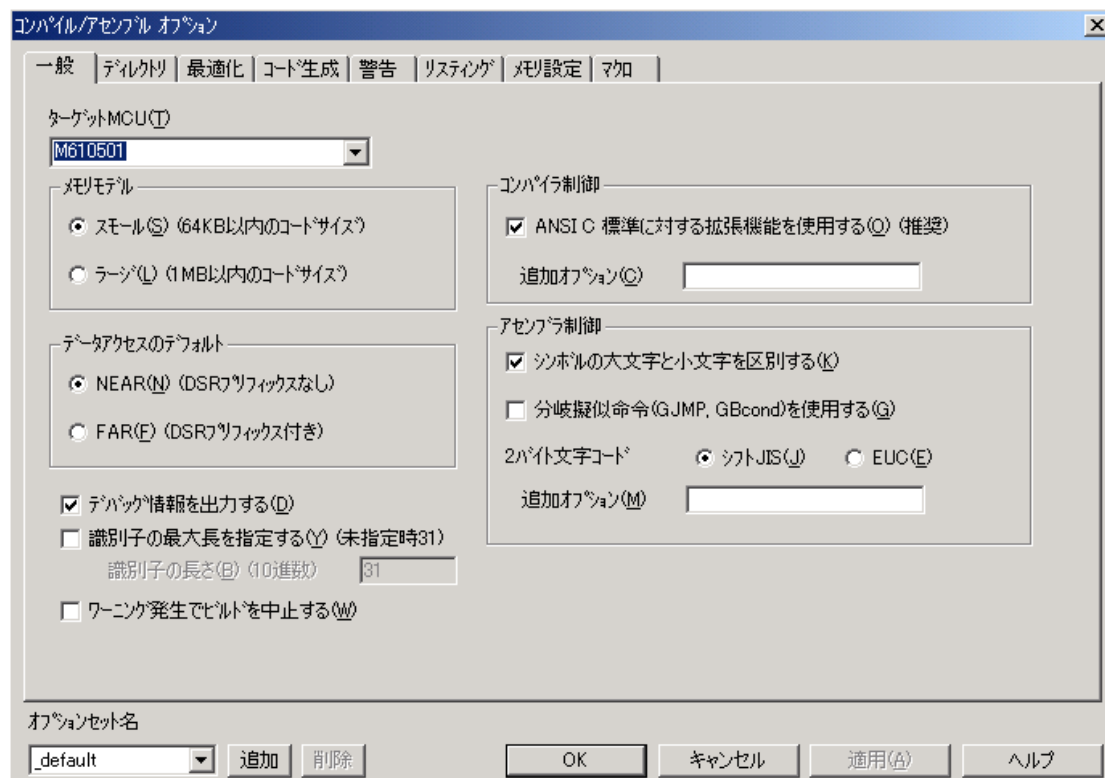
## 4.4 コンパイル／アセンブルオプション設定

IDEU8 上でプログラムのビルドを行なう前に、まず、CCU8 コンパイラ、RASU8 アセンブラへ渡すオプション設定を行なう必要があります。これらのオプション設定は、[プロジェクト]メニューの[オプション]→[コンパイル／アセンブル...]で行ないます。

コンパイル／アセンブルオプションは、ビルドツールのうち中間オブジェクトファイルの生成に関する設定を行ないます。

### 4.4.1 一般タブ

コンパイル／アセンブルオプションダイアログの一般タブを選択すると、次のようなダイアログボックスが表示されます。



#### ■ ターゲット MCU

コンパイラに指定するターゲット MCU の機種名を指定します。  
CCU8 コンパイラの /T オプションに設定されます。

#### ■ メモリモデル

これらのラジオボタンに、ターゲット MCU をどのメモリモデルで使用するかを設定します。  
[スモール (64KB 以内のコードサイズ)]を選択すると、CCU8 コンパイラと RASU8 アセンブラへ /MS オプションが指定され、スモールモデルが設定されます。スモールモデルでは、使用可能な ROM 空間が 64K バイトとなります。  
[ラージ (1MB 以内のコードサイズ)]を選択すると、CCU8 コンパイラと RASU8 アセンブラへ /ML オプションが指定され、ラージモデルが設定されます。ラージモデルでは、使用可能な ROM 空間

## 4章 プロジェクト管理

が 1M バイトとなります。

### ■ データアクセスのデフォルト

これらのラジオボタンには、アドレッシング指定子を省略したときのデータアクセスモデルを設定します。

[NEAR (DSR プリフィックスなし)]を選択すると、CCU8 コンパイラに/near オプションが指定され、RASU8 アセンブラに/DN オプションが指定されます。

[\_\_far 記述の禁止および割り込み関数内での DSR 退避を抑止する]をチェックすると、CCU8 コンパイラに/nofar オプションが指定されます。

[FAR (DSR プリフィックスあり)]を選択すると、CCU8 コンパイラに/far オプションが指定され、RASU8 アセンブラに/DF オプションが指定されます。

### ■ コンパイラ制御

[ANSI C 標準に対する拡張機能を使用する (推奨)]チェックボックスのチェックを外すと、CCU8 コンパイラに/Za オプションが指定されます。

[追加オプション]入力フィールドには、IDEU8 が対応していない CCU8 コンパイラのオプションを指定できます。

### ■ アセンブラ制御

[シンボルの大文字と小文字を区別する]をチェックすると、RASU8 アセンブラに/CD オプションを指定します。

[分岐擬似命令 (GJMP、GBcond) を使用する]をチェックすると、RASU8 アセンブラに/G オプションを指定します。

[2 バイト文字コード]ラジオボタンで、使用する全角文字コード(日本語)を選択します。[EUC] ボタンを選択すると、RASU8 アセンブラに/KE オプションを指定します。

[追加オプション]入力フィールドには、IDEU8 が対応していない RASU8 アセンブラのオプションを指定できます。

### ■ デバッグ情報を出力する

このチェックボックスをチェックすると、C 言語のソースファイル(.C)をコンパイルする場合は、CCU8 コンパイラと RASU8 アセンブラに/SD オプションを指定し、アセンブリ言語のソースファイル(.ASM)をアセンブルする場合には、RASU8 アセンブラに/D オプションを指定します。

### ■ 識別子の最大長を指定する

このチェックボックスをチェックすると、CCU8 コンパイラと RASU8 アセンブラに/SL オプションを指定します。このチェックボックスをチェックした場合は、シンボルサイズ入力フィールドが有効になり、シンボル長を指定できるようになります。

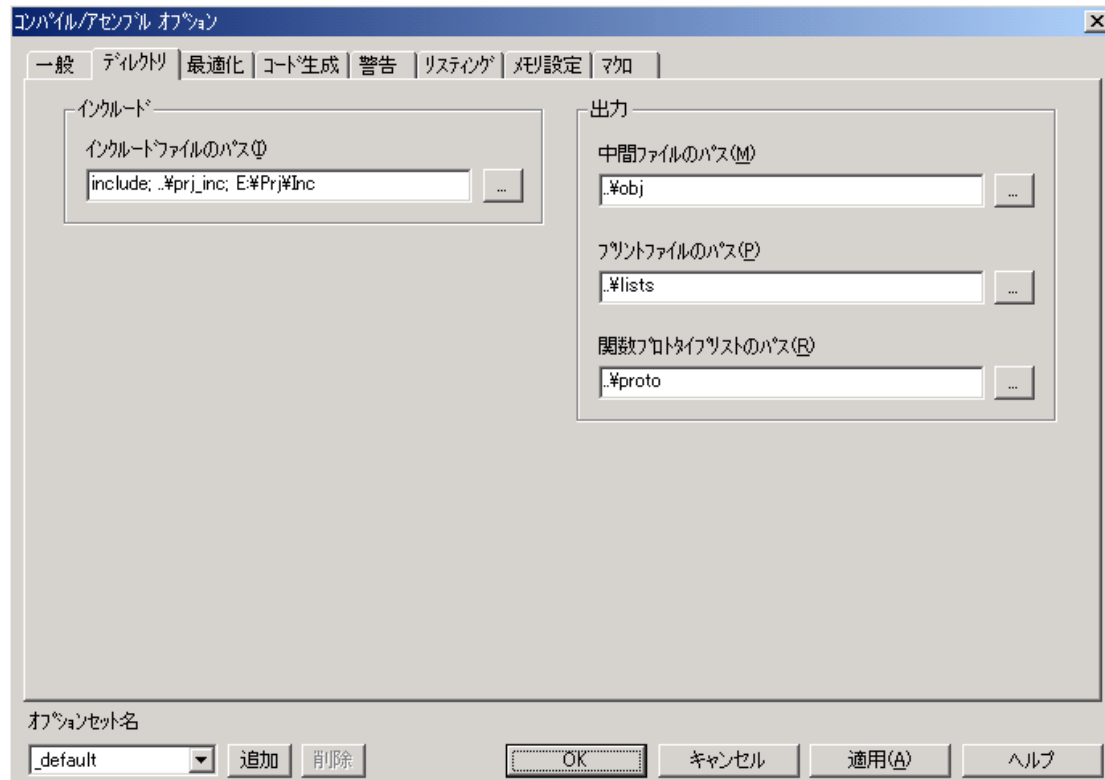
### ■ ワーニング発生でビルドを中止する

このチェックボックスをチェックすると、CCU8 コンパイラと RASU8 アセンブラがワーニングレベルのエラーを報告すると、ビルド処理を中止します。

## 4.4.2 ディレクトリタブ

コンパイル／アセンブルオプションダイアログのディレクトリタブを選択すると、次のような

ダイアログボックスが表示されます。<sup>8</sup>



## ■ インクルード

インクルードフィールドは、Cソースファイル中の#include文やアセンブリ言語ソースファイルの中のINCLUDE擬似命令で指定されているインクルードファイルの検索ディレクトリを指定するためのものです。<sup>9</sup>

CCU8 コンパイラの/I オプションおよびRASU8 アセンブラの/I オプションとして渡されます。

インクルードディレクトリを複数指定する場合は、セミコロン(;)やカンマ(,)で区切って指定します。

## ■ 出力

出力フィールドには、CCU8 コンパイラとRASU8 が生成するファイルのディレクトリを指定します。

[中間ファイルのパス]には、CCU8 コンパイラが生成する中間ファイル(.ASM)の出力パス、および、RASU8 アセンブラが生成するリロケータブルオブジェクトファイル(.OBJ)の出力パスを指定します。このフィールドにパス名を指定すると、CCU8 コンパイラへ/Fa オプション、RASU8 アセンブラに/O オプションが指定されます。

[プリントファイルのパス]には、[リスティング]タブで出力設定する、以下の各種ファイルの出力先を指定します。

- CCU8 コンパイラが出力するエラーファイル(.LER)

<sup>8</sup> IDEU8 の全てのディレクトリ指定は、IDEU8 のプロジェクトファイル(.PID)が置かれているディレクトリをカレントとして解釈されます。

<sup>9</sup> [ファイル]メニューの[環境設定]→[システムパス...]で指定するシステムインクルードパスにも、CCU8 コンパイラのためのインクルードファイルの検索パスを指定できます。



## 4章 プロジェクト管理

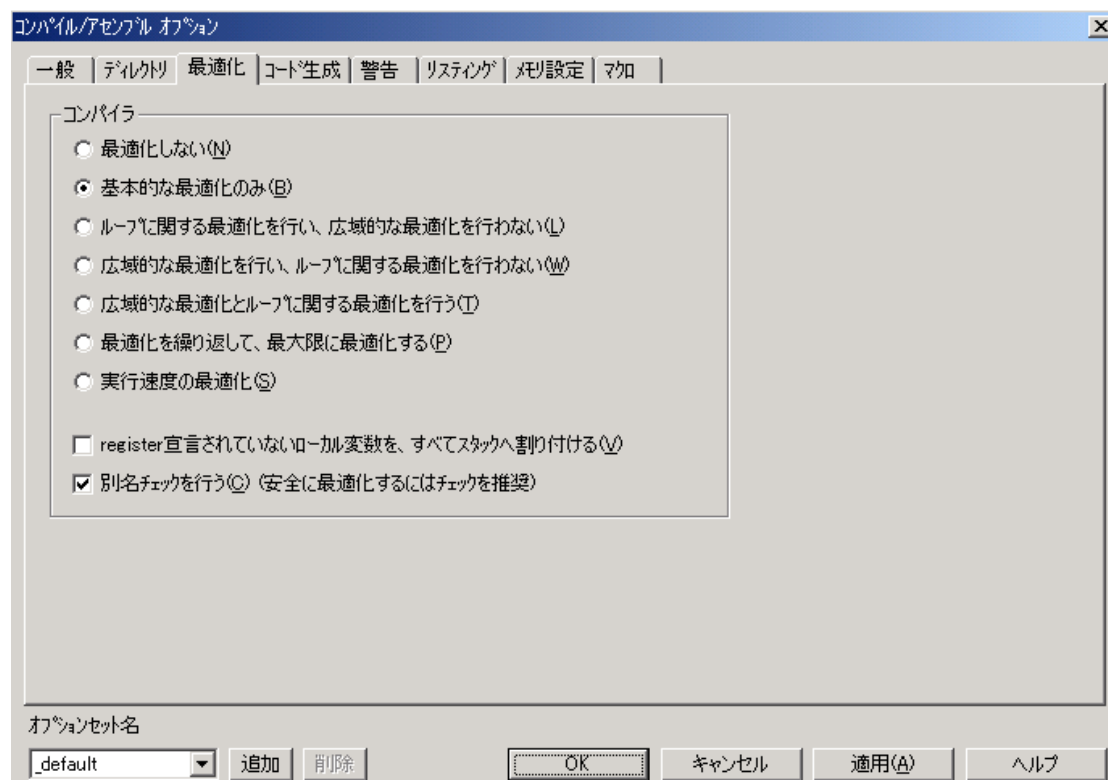
- ・ CCU8 コンパイラが出力するコールツリーファイル(.CAL)
- ・ CCU8 コンパイラが出力するプリプロセス結果のリスト(.I)
- ・ RASU8 アセンブラが生成するプリントファイルの出力パス(.PRN)

[関数プロトタイプリストのパス]には、以下の出力ファイルの出力先を指定します。

- ・ CCU8 コンパイラが出力する関数プロトタイプリストファイル(.PRO)
- ・ RASU8 アセンブラが出力する EXTRN 宣言ファイル(.EXT)

### 4.4.3 最適化タブ

コンパイル／アセンブルオプションダイアログの最適化タブを選択すると、次のようなダイアログボックスが表示されます。



#### ■ コンパイラ

コンパイラフィールドは、CCU8 コンパイラの最適化オプションを指定するためのフィールドです。

[最適化しない]を選択すると、CCU8 コンパイラへ/0d オプションが指定され、最適化が行われなくなります。

[基本的な最適化のみ]を選択すると、CCU8 コンパイラに、最適化オプションを指定しないデフォルトの最適化が行われます。

[ループに関する最適化を行い、広域的な最適化を行わない]を選択すると、CCU8 コンパイラに/0l オプションを指定し、ループに関する最適化のみが行われます。

[広域的な最適化を行い、ループに関する最適化を行わない]を選択すると、CCU8 コンパイラに/0g オプションを指定し、広域的な最適化が行われます。

[広域的な最適化とループに関する最適化を行う]を選択すると、CCU8 コンパイラに/0i オプシ

ョンと/0g オプションを指定します。

[最適化を繰り返して、最大限の最適化する]を選択すると、CCU8 コンパイラに/0m オプションを指定します。

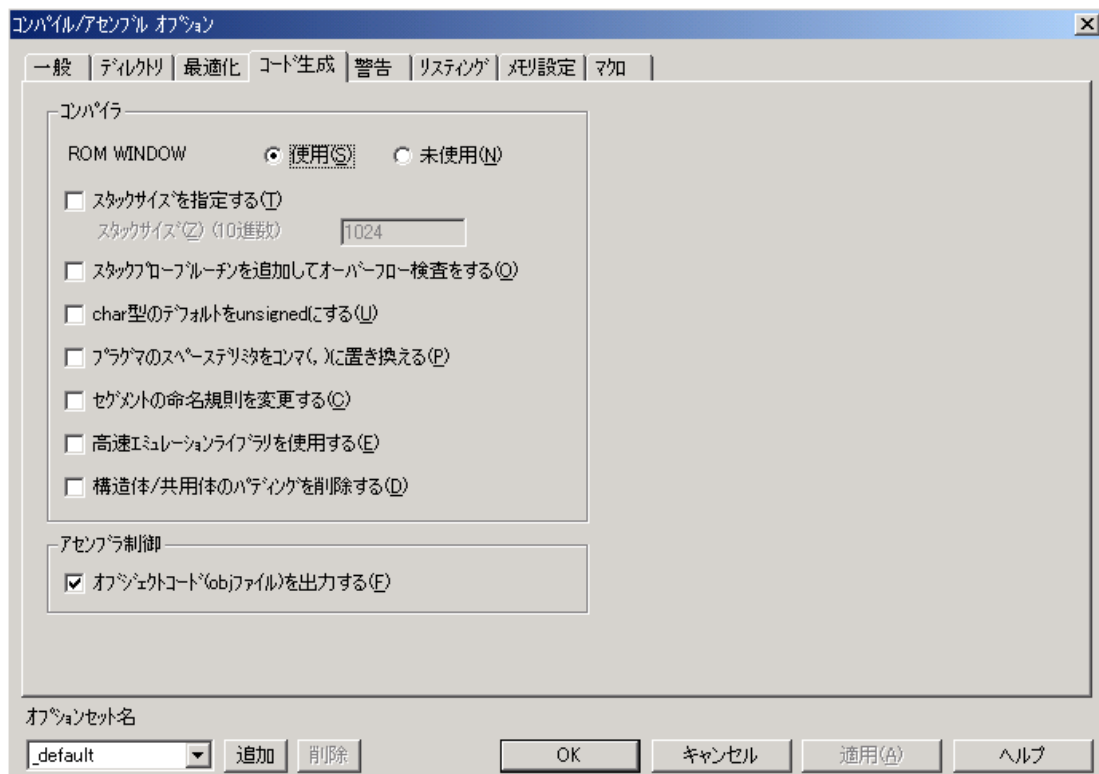
[実行速度の最適化]を選択すると、CCU8 コンパイラに/0t オプションを指定し、実行速度の最適化が行なわれます。

[register 宣言されていないローカル変数を、すべてスタックへ割り付ける]をチェックすると、CCU8 コンパイラに/Zs オプションを指定します。

[別名チェックを行う（安全に最適化するにはチェックを推奨）]をチェックすると、CCU8 コンパイラに/0a オプションを指定し、安全に最適化できるように、別名チェックを行います。

#### 4.4.4 コード生成タブ

コンパイル／アセンブルオプションダイアログのコード生成タブを選択すると、次のようなダイアログボックスが表示されます。



#### ■ コンパイラ

[ROM WINDOW]ラジオボタンで、[未使用]を選択すると、CCU8 コンパイラに、/NOWIN オプションを指定して、ROM WINDOW 領域を使用しないようにします。

[スタックサイズを指定する]をチェックすると、CCU8 コンパイラに/SS オプションを指定します。このチェックボックスをチェックする場合は、スタックサイズ入力フィールドが有効になり、スタックサイズを指定できるようになります。

[スタックプローブルーチンを追加してスタックオーバーフローを検査する]をチェックすると、CCU8 コンパイラに/ST オプションを指定します。

[char 型のデフォルトを unsigned にする]をチェックすると、CCU8 コンパイラへ/J オプションが指定され、ソースプログラム中で記述された char 型の変数が、unsigned char 型として扱われ

## 4章 プロジェクト管理

ます。

[プラグマのスペースデリミタをカンマに置き換える]をチェックすると、CCU8 コンパイラに/PF オプションを指定します。

[セグメントの命名規則を変更する]をチェックすると、CCU8 コンパイラに/SYS オプションを指定します。このオプションは、システムファイルをコンパイルするときに使用されます。

[高速エミュレーションライブラリを使用する]をチェックすると、CCU8 コンパイラに/Ff オプションを指定して、float 型の実行時間を高速化します。

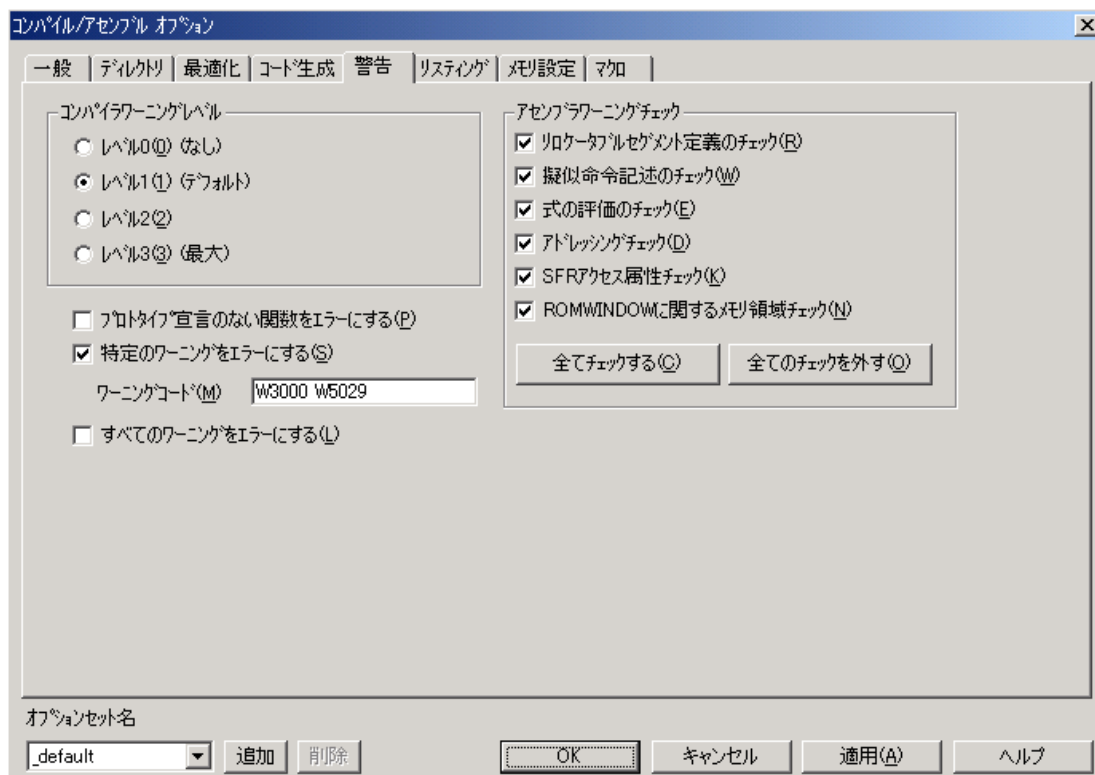
[構造体／共用体のパディングを削除する]をチェックすると、CCU8 コンパイラに/Zp オプションを指定します。

### ■ アセンブラ制御

[オブジェクトコード(obj ファイル)を出力する]のチェックを外すと、RASU8 アセンブラに/NO オプションを指定して、オブジェクトファイルの生成を抑制します。

### 4.4.5 警告タブ

コンパイル／アセンブルオプションダイアログの警告タブを選択すると、次のようなダイアログボックスが表示されます。



### ■ コンパイラワーニングレベル

コンパイラワーニングレベルフィールドは、CCU8 コンパイラのワーニングレベルを指定するフィールドです。レベル 0～3 のラジオボタンを選択して、CCU8 コンパイラのワーニングレベルを指定します。CCU8 コンパイラの/W オプションとして指定します。

#### ■ アセンブラワーニングチェック

アセンブラワーニングチェックフィールドは、RASU8 アセンブラのワーニングチェックに関して設定を行います。

[リロケータブルセグメント定義のチェック]をチェックすると、RASU8 アセンブラに/WR オプションを指定します。これにより、ROMWINDOW 領域が指定されていない場合の弊害やスタックセグメントのサイズに関するチェックが行なわれます。

[擬似命令記述のチェック]をチェックすると、RASU8 アセンブラに/WP オプションを指定します。これにより、リスティング擬似命令や MODEL 擬似命令などのチェックが行なわれます。

[式の評価のチェック]をチェックすると、RASU8 アセンブラに/WE オプションを指定します。これにより、アドレス式や数値式、各種演算の書式がチェックされます。

[アドレッシングチェック]をチェックすると、RASU8 アセンブラに/WA オプションを指定します。これにより、ユーセージタイプやアドレス空間、NEAR 分岐命令に関するチェックが行なわれます。

[SFR アクセス属性チェック]をチェックすると、RASU8 アセンブラに/WS オプションが指定され、SFR のアクセス制限に関するチェックが行なわれます。

[ROMWINDOW に関するメモリ領域チェック]をチェックすると、RASU8 アセンブラへ/WT オプションが指定され、ROM WINDOW 領域範囲未定が起因となるメモリ領域のチェックが行なわれます。

#### ■ プロトタイプ宣言のない関数をエラーにする

このチェックボックスをチェックすると、CCU8 コンパイラに/Wc W5031 オプションを指定します。

#### ■ 特定のワーニングをエラーにする

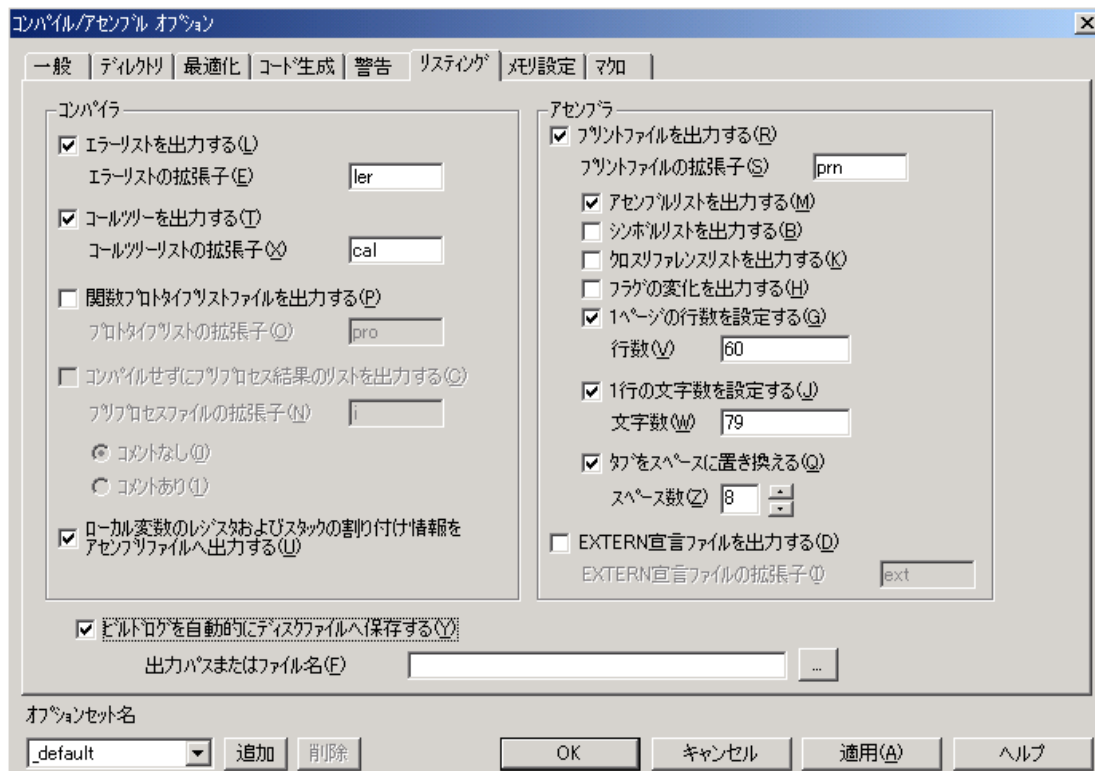
このチェックボックスをチェックすると、CCU8 コンパイラに/Wc オプションを指定します。このチェックボックスをチェックする場合は、エラーにするワーニング番号を指定できるようになります。複数のワーニング番号を指定するときは、カンマ(,)または空白で区切って指定してください。

#### ■ すべてのワーニングをエラーにする

「すべてのワーニングをエラーにする」チェックボックスがチェックされると、CCU8 コンパイラに/Wa オプションを指定します。このチェックボックスをチェックされた場合は、ワーニングレベルの[レベル 0]ラジオボタン、[特定のワーニングをエラーにする]チェックボックスや[プロトタイプ宣言のない関数をエラーにする]チェックボックスがグレーになり無効になります。

#### 4.4.6 リスティングタブ

コンパイル／アセンブルオプションダイアログのリスティングタブを選択すると、次のようなダイアログボックスが表示されます。



## ■ コンパイラ

コンパイラフィールドは、CCU8 コンパイラの出力ファイルに関するオプションを指定するためのフィールドです。

[エラーリストを出力する]をチェックすると、CCU8 コンパイラに/LEオプションを指定し、CCU8 コンパイラはエラーリストファイルを生成します。[エラーリストの拡張子]に、エラーリストファイルの拡張子を指定してください。<sup>10</sup>

[コールツリーを出力する]をチェックすると、CCU8 コンパイラに/CTオプションを指定し、CCU8 コンパイラはコールツリーリストファイルを生成します。[コールツリーリストの拡張子]に、コールツリーリストファイルの拡張子を指定してください。<sup>11</sup>

[関数プロトタイプリストファイルを出力する]をチェックすると、CCU8 コンパイラに/Zgオプションを指定し、CCU8 コンパイラは関数プロトタイプリストファイルを生成します。[関数プロトタイプリストの拡張子]に、関数プロトタイプリストファイルの拡張子を指定してください。<sup>12</sup>

[コンパイルせずにプリプロセス結果のリストを出力する]をチェックし、[コメントなし]ラジオボタンを選択すると、CCU8 コンパイラに/PC オプションを指定します。この場合 CCU8 コンパイラは、入力ファイル毎のプリプロセス結果をコメントなしでリストファイルに出力します。

[コンパイルせずにプリプロセス結果のリストを出力する]をチェックし、[コメントあり]ラジオボタンを選択すると、CCU8 コンパイラに/LP オプションを指定します。この場合 CCU8 コンパイラは、入力ファイル毎のプリプロセス結果をコメントつきでリストファイルに出力します。

なお、[エラーリストを出力する]、[コールツリーを出力する]、[関数プロトタイプリストファイルを出力する]をひとつでもチェックすると、[コンパイルせずにプリプロセス結果のリストを

<sup>10</sup> CCU8 のエラーファイルのデフォルト拡張子は、LERです。

<sup>11</sup> IDEU8 のコールツリーリストファイルのデフォルト拡張子は、CALです。

<sup>12</sup> IDEU8 の関数プロトタイプリストファイルのデフォルト拡張子は、PROです。

出力する]がグレー表示になり選択できなくなります。

また、[コンパイルせずにプリプロセス結果のリストを出力する]をチェックすると、[エラーリストを出力する]、[コールツリーを出力する]、[関数プロトタイプリストファイルを出力する]がグレー表示になり選択できなくなります。

[ローカル変数のレジスタおよびスタックの割り付け情報をアセンブリファイルへ出力する]をチェックすると、CCU8 コンパイラに/Lv オプションを指定します。

### ■ アセンブラ

アセンブラフィールドは、RASU8 アセンブラの出力ファイルに関するオプションを指定するためのフィールドです。

[プリントファイルを出力する]をチェックすると、RASU8 アセンブラに/PR オプションを指定します。このチェックボックスをチェックすると、その下のプリントファイルに出力する内容を選択することができます。

[アセンブルリストを出力する]をチェックすると、RASU8 アセンブラに/L オプションを指定します。

[シンボルリストを出力する]をチェックすると、RASU8 アセンブラに/S オプションを指定します。

[クロスリファレンスリストを出力する]をチェックすると、RASU8 アセンブラに/R オプションを指定します。

[フラグの変化を出力する]をチェックすると、RASU8 アセンブラに/SF オプションを指定します。

[1 ページの行数を設定する]をチェックすると、RASU8 アセンブラに、行数フィールドで指定した 1 ページの行数と併せて/PL オプションを指定します。

[1 行の文字数を設定する]をチェックすると、RASU8 アセンブラに、文字数フィールドで指定した 1 行の文字数と併せて/PW オプションを指定します。

[タブをスペースに置き換える]をチェックすると、RASU8 アセンブラに、スペース数フィールドで指定したスペースの数と併せて/T オプションを指定します。

[エラーファイルを出力する]をチェックすると、RASU8 アセンブラに/E オプションを指定します。

[EXTERN 宣言ファイルを出力する]をチェックすると、RASU8 アセンブラに/X オプションを指定します。

### ■ ビルドログを自動的にディスクファイルへ保存する

このチェックボックスをチェックすると、CCU8 コンパイラと RASU8 アセンブラが表示するビルドログを指定されたディレクトリに保存します。このログは、コンパイル／アセンブルする単位で、それぞれ、個別に作成されます。

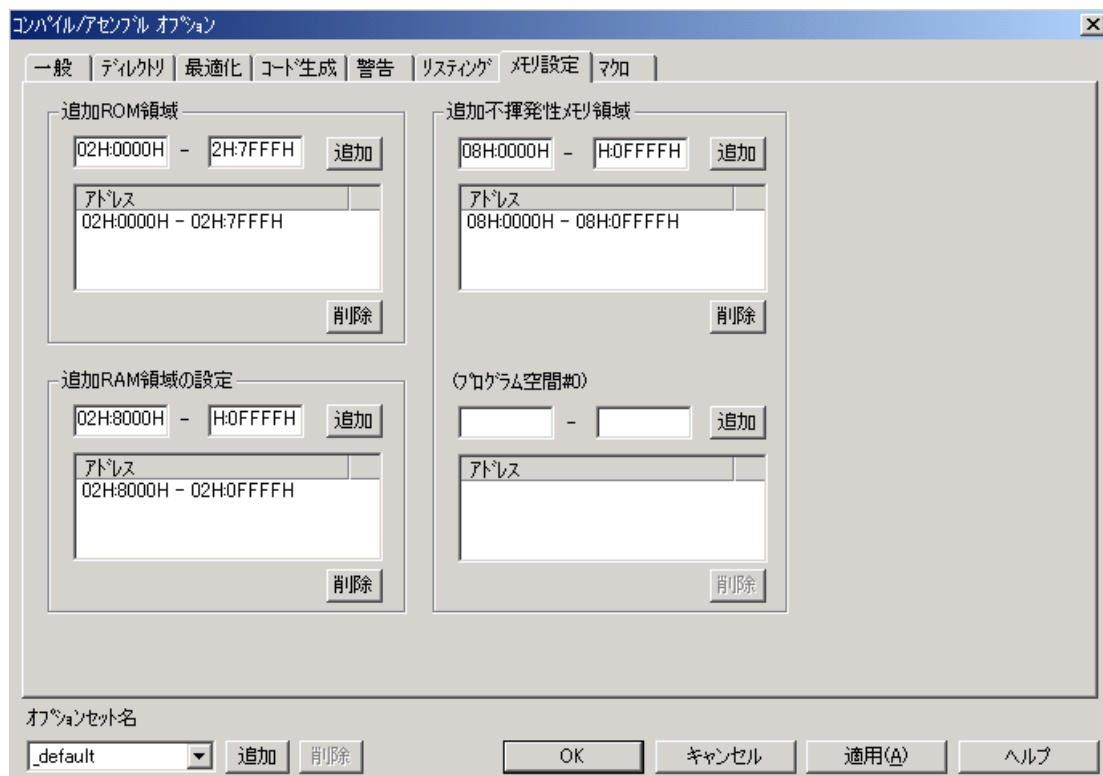
ログファイルの出力先は、[出力パスまたはファイル名]に指定します。空白の場合には、IDEU8 のプロジェクトファイル(.PID)と同じディレクトリに作成します。ディレクトリパスを指定すると、ビルドしたソースファイルのベース名に.LOG という拡張子を付けたファイル名になります。

## 4.4.7 メモリ設定タブ

コンパイル／アセンブルオプションダイアログのメモリ設定タブを選択すると、次のようなダ

## 4章 プロジェクト管理

イアログボックスが表示されます。



このタブでは、ユーザシステム上にメモリを追加した場合に、追加メモリの種類と領域を指定します。各フィールドには、複数のアドレス範囲を追加できます。

### ■ 追加 ROM 領域

ROM 領域を追加します。RASU8 アセンブラに/BROM オプションを指定します。

### ■ 追加 RAM 領域

RAM 領域を追加します。RASU8 アセンブラに/BRAM オプションを指定します。

### ■ 追加不揮発性メモリ領域

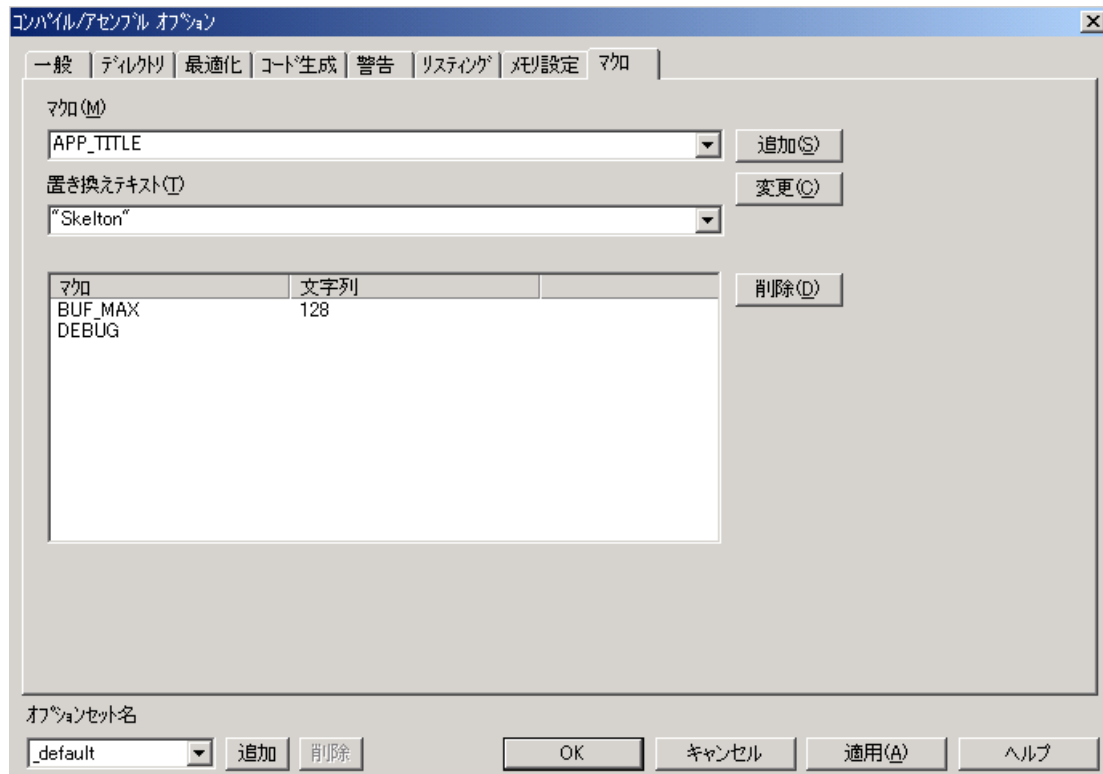
不揮発性メモリ領域を追加します。RASU8 アセンブラに/BNVRAM オプションを指定します。

### ■ 追加不揮発性メモリ領域(プログラムメモリ空間#0)

プログラムメモリ空間上の物理セグメントアドレス#0 に不揮発性メモリを追加します。RASU8 アセンブラに/BNVRAMP オプションを指定します。

## 4.4.8 マクロタブ

コンパイル／アセンブルオプションダイアログのマクロタブを選択すると、次のようなダイアログボックスが表示されます。



#### ■ マクロ

マクロフィールドには、CCU8 コンパイラの/D オプションと RASU8 アセンブラの/DEF オプションに指定するマクロ名を入力します。

#### ■ 置き換えテキスト

置き換えテキストフィールドには、マクロフィールドで指定したマクロ名に置き換える文字列を指定します。

#### ■ 追加ボタン

マクロフィールドと置き換えテキストフィールドに入力して、追加ボタンを押すと、リストボックスに登録されます。

#### ■ 削除ボタン

リストボックスからマクロを選択し、削除ボタンを押すと、選択したマクロを削除できます。

## 4.5 ターゲットオプション設定

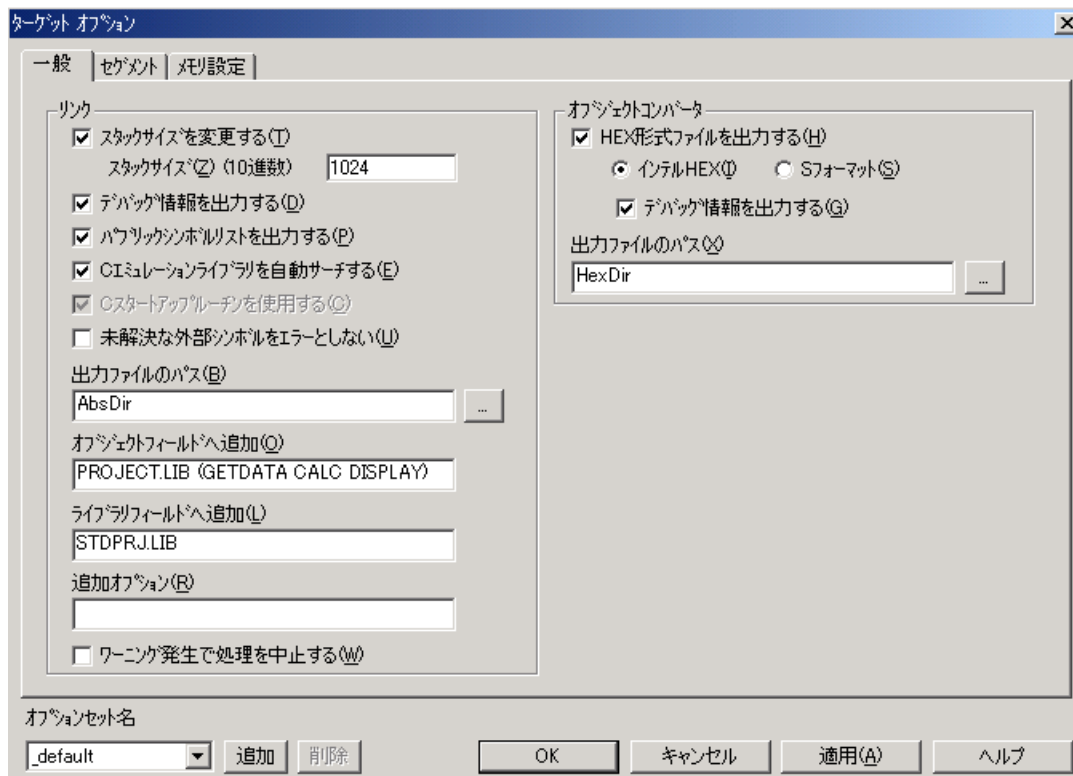
ターゲットオプションは、ビルドツールのうちターゲットファイルの生成に関する設定を行います。

ターゲットオプションダイアログは、一般、セグメント、メモリ設定の3つのタブに分類されています。

### 4.5.1 一般タブ

ターゲットオプションダイアログの一般タブを選択すると、次のようなダイアログボックスが表示されます。





## ■ リンク

リンクフィールドには、RLU8 リンカに設定するオプションを指定します。

[スタックサイズを変更する]をチェックすると、スタックセグメントのサイズを変更するために、RLU8 リンカに/STACK オプションを指定します。

[デバッグ情報を出力する]をチェックすると、RLU8 リンカに/D オプションと/SD オプションを指定します。このオプションをチェックすると、DTU8 デバッガでデバッグすることができます。

[パブリックシンボルリストを出力する]をチェックすると、RLU8 リンカに/S オプションを指定し、パブリックシンボルリストをマップファイルに出力します。

[Cエミュレーションライブラリを自動サーチする]をチェックすると、RLU8 リンカに/CC オプションを指定します。このチェックを外すと、[C スタートアップルーチンを使用する]チェックボックスが有効になります。

[Cエミュレーションライブラリを自動サーチする]をチェックせずに、[C スタートアップルーチンを使用する]をチェックすると、スタートアップルーチンを使用するために、/COMB(\$\$init\_info \$\$init\_info\_end) /COMB(\$\$content\_of\_init \$\$end\_of\_init)オプションを指定します。

[未解決な外部シンボルをエラーとしない]をチェックすると、RLU8 リンカに/EXC オプションを指定し、未使用のイクスターナルシンボルに対しては未解決エラーになりません。

[出力パス]には、RLU8 リンカが生成するアブソリュートオブジェクトファイル(.ABS)の出力先のパスを指定します。

[オブジェクトフィールドへ追加]入力フィールドに指定したテキストは、そのまま、RLU8 リンカの object\_files フィールドに付加されます。プロジェクトウィンドウにオブジェクトファイル(.OBJ)を登録できないとき(絶対パスやライブラリファイルの中のモジュール指定など)、このフ

フィールドを使用します。

[ライブラリフィールドへ追加]入力フィールドに指定したテキストは、そのまま、RLU8 リンカの `libraries` フィールドに付加されます。プロジェクトウィンドウにライブラリファイルを登録できないとき(システムライブラリ検索パスや絶対パスなど)、このフィールドを使用します。

[追加オプション]入力フィールドには、IDEU8 が対応していない RLU8 リンカのオプションを指定します。

[ワーニング発生でビルドを中止する]チェックボックスをチェックすると、リンク処理でワーニングレベルのエラーが発生したときに、リンク処理に続く処理を中止します。[プロジェクト]メニューの[デバッグ]に伴うビルド処理の場合は、DTU8 デバッガが起動しません。

### ■ オブジェクトコンバータ

オブジェクトコンバータフィールドには、OHU8 オブジェクトコンバータに設定するオプションを指定します。

[HEX 形式ファイルを出力する]をチェックすると、リンク後に OHU8 オブジェクトコンバータが起動します。

[インテル HEX]を選択すると、インテル HEX 形式の HEX ファイルを生成するために、OHU8 オブジェクトコンバータに/H オプションを指定します。

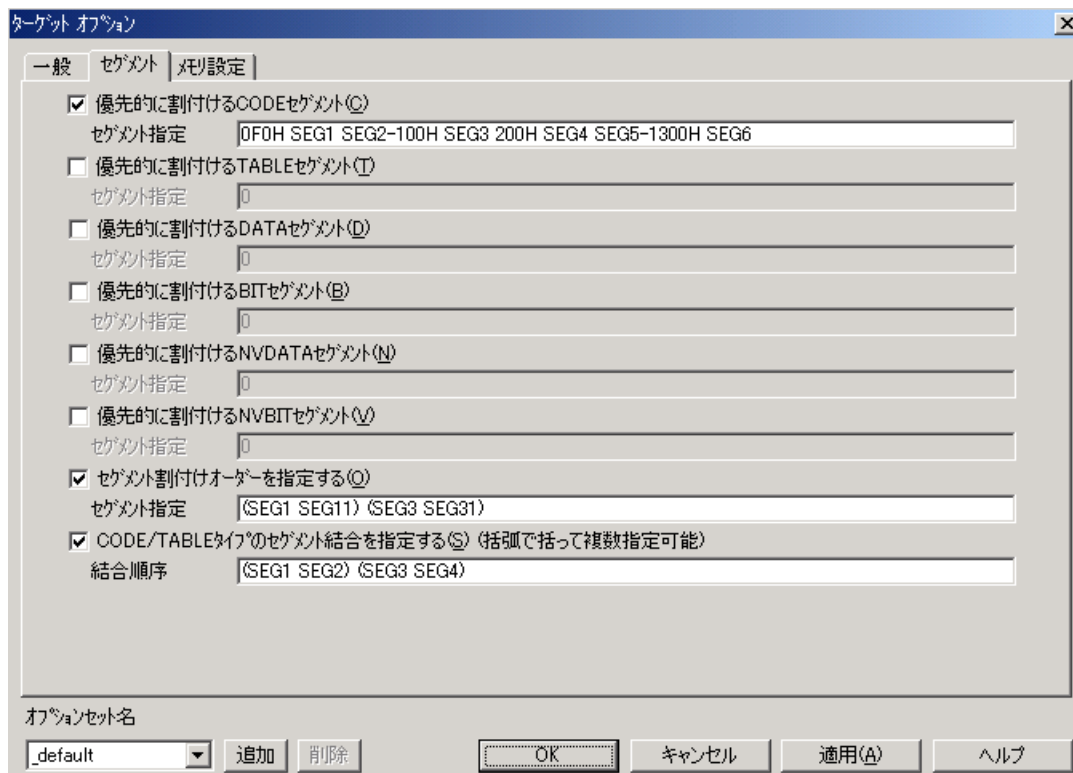
[S フォーマット]を選択すると、モトローラ S2 フォーマットの HEX ファイルを生成するために、OHU8 オブジェクトコンバータに/S オプションを指定します。

[デバッグ情報を出力する]をチェックすると、OHU8 オブジェクトコンバータへ/D オプションが指定され、HEX ファイルと共にシンボル情報が出力されます。

[出力パス]には、OHU8 オブジェクトコンバータが生成する HEX ファイルの出力先のパスを指定します。

### 4.5.2 セグメントタブ

ターゲットオプションダイアログのセグメントタブを選択すると、次のようなダイアログボックスが表示されます。



### ■ セグメント割付け

「優先的に割付ける CODE セグメント」には、RLU8 リンカの /CODE オプションの書式で、CODE セグメント割付け制御オプションを指定します。

「優先的に割付ける TABLE セグメント」には、RLU8 リンカの /TABLE オプションの書式で、TABLE セグメント割付け制御オプションを指定します。

「優先的に割付ける DATA セグメント」には、RLU8 リンカの /DATA オプションの書式で、DATA セグメント割付け制御オプションを指定します。

「優先的に割付ける BIT セグメント」には、RLU8 リンカの /BIT オプションの書式で、BIT セグメント割付け制御オプションを指定します。

「優先的に割付ける NVDATA セグメント」には、RLU8 リンカの /NVDATA オプションの書式で、NVDATA セグメント割付け制御オプションを指定します。

「優先的に割付ける NVBIT セグメント」には、RLU8 リンカの /NVBIT オプションの書式で、NVBIT セグメント割付け制御オプションを指定します。

「セグメント割付けオーダーを指定する」をチェックすると、RLU8 リンカに /ORDER オプションを指定し、セグメント指定フィールドに入力したセグメントの並びが設定され、セグメント割り付け処理の順序を制御できます。複数の割り付けの指定を行うには、丸括弧でくくり、複数の /ORDER オプションを指定できます。

「CODE/TABLE タイプのセグメント結合を指定する」をチェックすると、RLU8 リンカの /COMB オプションを指定できます。結合するセグメント名を丸括弧でくくり、複数の /COMB オプションを指定できます。

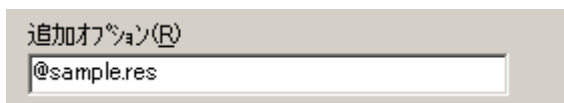
## 4.5 ターゲットオプション設定

各セグメント指定フィールドおよび結合順序フィールドに入力するパラメータの書式については、「MACU8 アセンブラパッケージ ユーザーズマニュアル」を参照してください。

各セグメント指定フィールドおよび結合順序フィールドに入力する情報が多くなる場合は、ダイアログ上での設定内容の確認がしづらくなります。そのような場合には、RLU8 リンカの応答ファイルを利用されることをお勧めします。例えば、以下のように応答ファイルを作成します。ここでは、ファイル名を sample.res とします。

```
//応答ファイル (sample.res)
/COFF(0F0H SEG1 SEG2-100H SEG3 200H SEG4 SEG5-1300H SEG6)
/ORDER(SEG1 SEG11)
/ORDER(SEG3 SEG31)
/COMB(SEG1 SEG2)
/COMB(SEG3 SEG4)
```

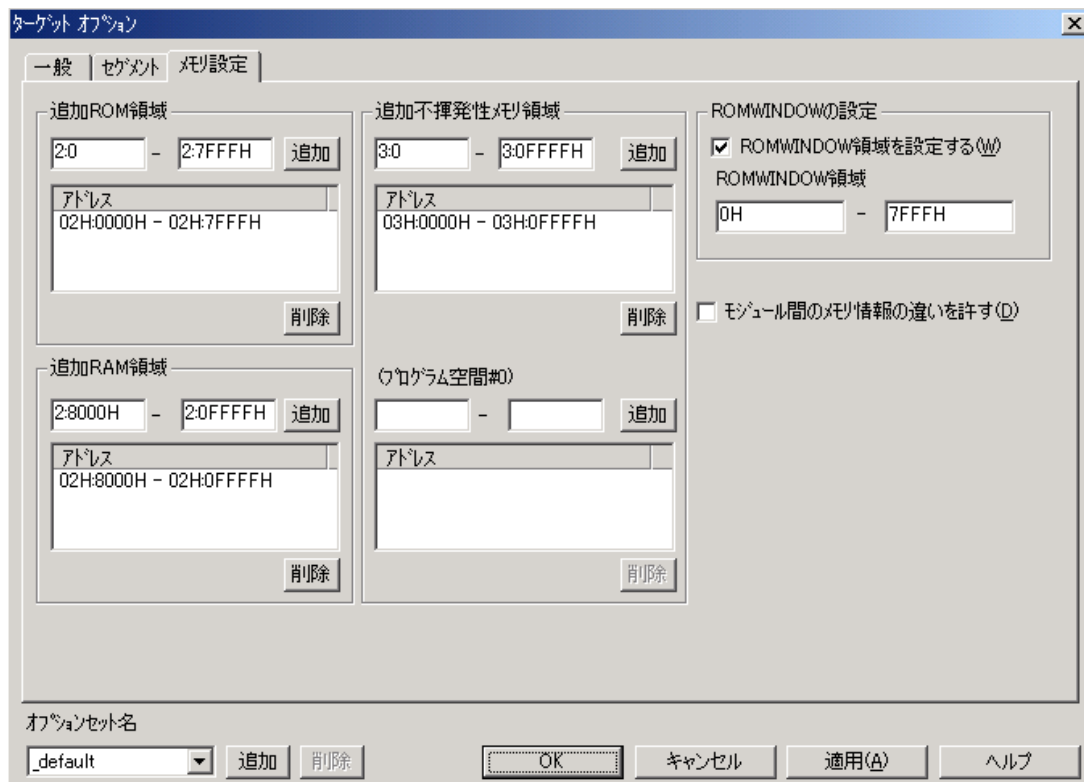
作成したファイルをプロジェクトファイルと同じフォルダへ置き、[ターゲットオプション]ダイアログの[一般]タブにある[追加オプション]フィールドに、以下のように応答ファイル名の先頭に@を付加して指定します。



このように指定することで、応答ファイル内に定義したオプションをRLU8 リンカに渡すことができます。

### 4.5.3 メモリ設定タブ

ターゲットオプションダイアログのメモリ設定タブを選択すると、次のようなダイアログボックスが表示されます。



このタブでは、ユーザシステム上にメモリを追加した場合に、追加メモリの種類と領域を指定します。各フィールドには、複数のアドレス範囲を追加できます。

#### ■ 追加 ROM 領域の設定

ROM 領域を追加します。RLU8 リンカに /ROM オプションを指定します。

#### ■ 追加 RAM 領域の設定

RAM 領域を追加します。RLU8 リンカに /RAM オプションを指定します。

#### ■ 追加不揮発性メモリ領域

不揮発性メモリ領域を追加します。RLU8 リンカに /NVRAM オプションを指定します。

#### ■ 追加不揮発性メモリ領域(プログラムメモリ空間#0)

プログラムメモリ空間上の物理セグメントアドレス#0 に不揮発性メモリを追加します。RLU8 リンカに /NVRAMP オプションを指定します。

#### ■ ROMWINDOW の設定

ROMWINDOW の設定フィールドは、RLU8 リンカに ROM WINDOW 領域を指定するためのフィールドです。

[ROMWINDOW 領域を設定する]をチェックすると、RLU8 リンカに /ROMWIN オプションを指定します。

#### ■ モジュール間のメモリ情報の違いを許す

このチェックボックスをチェックすると、RLU8 リンカに /PDIF オプションを指定します。

## 4.6 オプションセットの定義

オプション設定ダイアログボックスで設定するオプションを、オプションセットという単位にまとめることができます。あらかじめ複数のオプションセットを定義しておき、プロジェクトウィンドウで、オプションセットを定義したいファイルを選択し、ポップアップメニューの[設定]で指定します。

なお、オプションセットには、コンパイル／アセンブルオプション用とターゲットオプション用とがあり、コンパイル／アセンブルオプション用は、プロジェクト全体およびソースファイル毎に指定できますが、ターゲットオプション用は、プロジェクトウィンドウのルート（プロジェクト名）にのみ指定可能です。

### ■ オプションセットの追加

ターゲットオプション用のオプションセットを新規に追加する場合は、ターゲットオプションダイアログの左下にある[オプションセット名]フィールドに新しいオプションセット名を入力し、[追加]ボタンを押します。

コンパイル／アセンブルオプション用のオプションセットを新規に追加する場合は、コンパイル／アセンブルオプションダイアログの左下にある[オプションセット名]フィールドに新しいオプションセット名を入力し、[追加ボタン]を押します。

### ■ ターゲットオプション用のオプションセットの設定

新規に追加したターゲットオプション用のオプションセットをプロジェクトに設定する場合は、プロジェクトウィンドウに表示されているプロジェクト名（ツリーの最上位）を選択し、ポップアップメニューの[設定]で表示されるサブメニューから目的のオプションセットを選択します。

現在設定されているオプションセットは、プロジェクトウィンドウ上のプロジェクト名の横に表示されます。

### ■ コンパイル／アセンブルオプション用のオプションセットの設定

新規に追加したコンパイル／アセンブル用のオプションセットを各ソースファイルに個別設定する場合は、プロジェクトウィンドウに表示されているソースファイル名（Source フォルダ内）を選択し、ポップアップメニューの[設定]で表示されるサブメニューから目的のオプションセットを選択します。

また、すべてのソースファイルに同一のオプションセットを設定する場合は、プロジェクトウィンドウに表示されている[Source]フォルダを選択し、ポップアップメニューの[設定]で表示されるサブメニューから目的のオプションセットを選択します。

現在設定されているオプションセットは、[Source]フォルダの横もしくは各ソースファイル名の横に表示されます。

### 使用例 1 C ソースファイルとアセンブリソースファイルにそれぞれ独立したオプションセットを定義する場合

コンパイル／アセンブルオプションで、[C\_Settings]と[ASM\_Settings]の2つのオプションセットを作成します。

[C\_Settings]にC ソースコンパイル用のオプションを設定します。

[ASM\_Settings]にアセンブリソース用のオプションを設定します。

プロジェクトに登録されている各C ソースファイルに[C\_Settings]のオプションセットを指定

## 4章 プロジェクト管理

し、各アセンブリソースファイルに[A\_Settings]のオプションセットを指定します。

### 使用例 2 デバッグ用とリリース用のオプションセットを定義する場合

コンパイル／アセンブルオプションで、[Debug]と[Release]というオプションセットを作成します。[Debug]にデバッグ用のコードを生成するためのマクロ定義や最適化抑制のオプションなどを設定しておきます。[Release]は、通常のオプション設定にしておきます。

[Debug]のオプションセットを参照したいソースコードに、[Release]のオプションセットをデバッグが終了したソースコードに指定します。

## 5章 プログラムのビルド

---

### 5.1 ビルド処理の概要

プロジェクトに登録されたCソースファイルは、CCU8 コンパイラによってコンパイルされ、アセンブリソースファイル(.ASM)に変換されます。続いてRASU8 アセンブラによってアセンブルされ、リロケートブルオブジェクトファイル(.OBJ)に変換されます。更にRLU8 リンカによってリンクされ、アブソリュートオブジェクトファイル(.ABS)にまとめられます。また、HEX ファイル形式のファイルが必要な場合には、OHU8 オブジェクトコンバータによってインテル HEX やモトローラ S2 形式に変換します。

IDEU8 は、これらの処理を行うために、CCU8 コンパイラ、RASU8 アセンブラ、RLU8 リンカ、OHU8 オブジェクトコンバータに、必要なファイルとオプションを指定して起動します。

さらに、IDEU8 のビルド処理では、各ソースファイルおよびソースファイルにインクルードされているファイルとオブジェクトファイルのタイムスタンプが比較され、オブジェクトファイルよりソースファイルの方のタイムスタンプが新しいファイルのみコンパイル、アセンブルされます。

### 5.2 ビルド

[プロジェクト]メニューの[ビルド]を指定するビルド処理では、最終ターゲットファイル(.ABS)を構築するために必要なオブジェクトファイル(.OBJ)に対して、再構築の必要性をチェックします。

それらのオブジェクトファイル(.OBJ)を構築するために必要なファイル（ソースファイルとインクルードファイル）のタイムスタンプが、ひとつでもオブジェクトファイル(.OBJ)より新しいものがあれば、そのソースファイルをコンパイルおよびアセンブルして、オブジェクトファイル(.OBJ)を更新します。オブジェクトファイルの再構築がひとつでも行われたら、最終ターゲットファイル(.ABS)の再構築が行われます。

これら一連の作業により、最小限のビルド処理となります。

IDEU8 は、コンパイル／アセンブルの結果として生成されたオブジェクトファイルを解析して、そのオブジェクトに関連するソースファイルやインクルードファイルのデータベースを作成します。

なお、オプションが変化した場合や、初めてのビルド時には、全てのソースファイルを処理します。

### 5.3 リビルド

[プロジェクト]メニューの[リビルド]を指定すると、プロジェクトに登録されているすべてのソースファイルに対するオブジェクトファイルをすべて削除してからビルド処理を開始します。このため、全てのソースファイルを処理することになります。



## 5章 プログラムのビルド

### 5.4 選択ファイルのビルド

[プロジェクト]メニューの[選択中のファイル名のビルド]を指定すると、プロジェクトウィンドウ上で指定したファイルのみをコンパイル、アセンブルします。この場合、タイムスタンプのチェックを行いません。

### 5.5 ビルドの中止

ビルド処理を中断する場合は、[プロジェクト]メニューの[ビルドの中止]を選択します。<sup>13</sup>

### 5.6 クリーン

IDEU8 のビルド処理により作成するオブジェクトファイルやリストファイルを削除する場合は、[プロジェクト]メニューの[クリーン]を選択します。これにより、次のビルドでは、すべてのソースファイルがビルドの対象となります。

クリーン後のビルドは、リビルドと同じ動作となります。

### 5.7 デバッグ

[プロジェクト]メニューの[デバッグ]を選択すると、ビルド処理を行い、最終ターゲットファイルを更新してから、DTU8 デバッガを起動します。

#### 5.7.1 プログラムの修正

DTU8 デバッガを起動中にプログラムを修正する場合は、再度 IDEU8 をアクティブにして、IDEU8 のエディタウィンドウで行ないます。プログラムの修正後、ビルドを行ない、オブジェクトファイルを再構築します。

最終ターゲットファイルの再構築が完了し、再度 DTU8 デバッガをアクティブにすると、引き続きデバッグを行なうことができます。

#### 5.7.2 IDEU8 と DTU8 デバッガ間のリンク

IDEU8 でプロジェクトファイルを切り替えたり、DTU8 デバッガで別のプログラムファイルを開いたら、IDEU8 と DTU8 デバッガ間のリンクが外れます。この場合には、DTU8 デバッガを一旦終了して、再度、IDEU8 から DTU8 デバッガを起動してください。もしくは、DTU8 デバッガで、IDEU8 で開いているプロジェクトが生成したアブソリュートオブジェクトファイルを開いてください。

### 5.8 アウトプットウィンドウ

アウトプットウィンドウは、コンパイル、アセンブルおよびリンクの経過と結果を表示するウィンドウです。

アウトプットウィンドウは、[表示]メニューの[アウトプット]を指定することにより、開くことができます。

---

<sup>13</sup> ビルドの中断は、ビルドツール (CCU8、RASU8、RLU8、OHU8) の次の呼び出しをやめるもので、各ツールの処理は完了します。

### 5.8.1 メッセージのコピー

アウトプットウィンドウで選択している内容を、Windows のクリップボードにコピーできます。メッセージをコピーするには、アウトプットウィンドウのポップアップメニューから[コピー]を選択します。

### 5.8.2 エラー発生ソース行へジャンプ

コンパイルおよびアセンブル時にエラーが発生した場合は、アウトプットウィンドウ上のエラーメッセージから、エディタウィンドウを開き、エラー発生箇所を表示することができます。

エラー発生ソース行へジャンプするには、アウトプットウィンドウのエラー行にカーソルを移動して[Enter]キーを押すか、ポップアップメニューから[エラー発生ソースへジャンプ]を選択するか、もしくはアウトプットウィンドウのエラー行をマウスでダブルクリックします。

### 5.8.3 メッセージの保存

アウトプットウィンドウに表示されている内容は、ログファイルに保存することができます。メッセージを保存する場合は、アウトプットウィンドウのポップアップメニューから[メッセージの保存]を選択します。

### 5.8.4 メッセージのクリア

アウトプットウィンドウに表示されているメッセージをクリアする場合は、ポップアップメニューから[クリア]を選択します。

### 5.8.5 ドッキングビュー

ポップアップメニューの[ドッキングビュー]がチェックされていると、アウトプットウィンドウを IDEU8 のメインウィンドウの下側にドッキングして表示します。ポップアップメニューの[ドッキングビュー]がチェックを外すと、アウトプットウィンドウをメインウィンドウから独立して表示します。

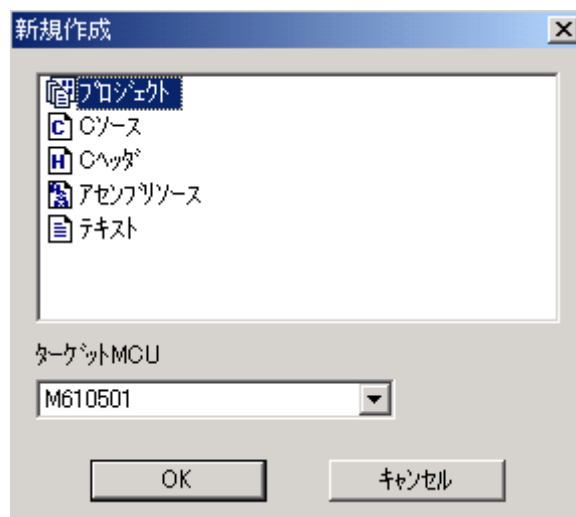
## 6章 その他の機能

---

### 6.1 各種ファイルの新規作成

IDEU8 上で、C ソースファイルやヘッダファイルなどのテキストファイルを作成することができます。

[ファイル]メニューの[新規作成...]を選択すると、次のようなダイアログボックスが表示されます。



#### ■ C ソースファイルの新規作成

新規作成ダイアログのリストから”C ソース”を選択すると、空の C ソースファイルがエディタウィンドウに表示されます。

#### ■ C ヘッダファイルの新規作成

新規作成ダイアログのリストから”C ヘッダ”を選択すると、空の C ヘッダファイルがエディタウィンドウに表示されます。

#### ■ アセンブリソースファイルの新規作成

新規作成ダイアログのリストから”アセンブリソース”を選択すると、空のアセンブリ言語ソースファイルがエディタウィンドウに表示されます。

#### ■ テキストファイルの新規作成

新規作成ダイアログのリストから”テキスト”を選択すると、空のテキストファイルがエディタウィンドウに表示されます。

なお、このダイアログボックスが開いたときに選択されているものは、[ファイル]メニューの

[新規作成...]を指定したときの選択されているウィンドウの種類と同じです。

## 6.2 エディタウィンドウ

エディタウィンドウは、ソースファイルやヘッダファイルを表示／編集するためのウィンドウです。

プログラムの編集には、エディタウィンドウを使用します。ソースファイルをエディタウィンドウで開くには、以下の方法があります。

### ■ メニューから開く

[ファイル]メニューの[開く]を選択し、表示されるダイアログボックスのファイルの種類を設定します。リストに表示されているファイルを選択し、[開く]ボタンを押します。

### ■ プロジェクトウィンドウから開く

プロジェクトウィンドウのツリーから開きたいファイルを選択し、ポップアップメニューの[開く]を選択します。

### ■ アウトプットウィンドウから開く

コンパイルやアセンブル時にエラーが発生すると、エラーの発生したファイル名と行番号がアウトプットウィンドウに表示されます。アウトプットウィンドウ上のエラーメッセージにカーソルを移動し、ポップアップメニューの[エラー発生ソースへジャンプ]を選択するか、または[Enter]キーを押すと、対象ファイルのエラー発生行をエディタウィンドウで開くことができます。

### 6.2.1 編集メニュー

エディタウィンドウでファイルを編集する場合は、[編集]メニューを使用します。

#### ■ 元に戻す

[編集]メニューの[元に戻す]を選択すると、最後のテキスト編集操作を元に戻します。

#### ■ 切り取り

[編集]メニューの[切り取り]を選択すると、現在選択中のテキストをクリップボードに移動します。

#### ■ コピー

[編集]メニューの[コピー]を選択すると、現在選択中のテキストをクリップボードにコピーします。

#### ■ 貼り付け

[編集]メニューの[貼り付け]を選択すると、カーソル位置にクリップボードの内容を挿入します。

#### ■ 削除

[編集]メニューの[削除]を選択すると、現在選択中のテキストを削除します。削除されたテキストは、クリップボードには入りません。

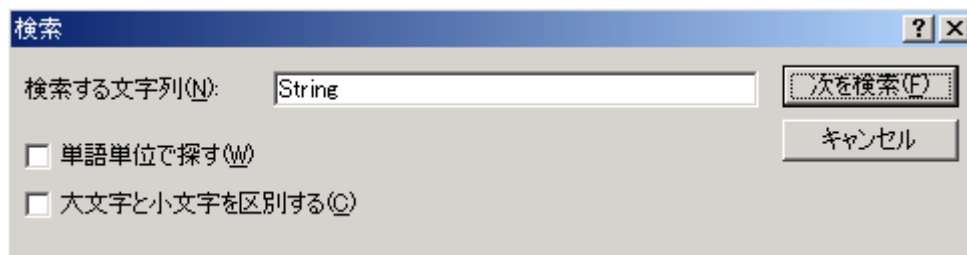
#### ■ すべてを選択

[編集]メニューの[すべてを選択]を選択すると、ファイル中の全テキストを選択状態にします。

## 6章 その他の機能

### ■ 検索

[編集]メニューの[検索]を選択すると、次のようなダイアログボックスが表示されます。



[検索する文字列]に、検索したいテキストを入力します。

[単語単位で探す]をチェックすると、検索テキストを単語毎に検索します。例えば、検索文字列に”rom”を指定した場合、”romwindow”や”from”などの単語は検索されません。

[大文字と小文字を区別する]をチェックすると、検索テキストの大文字と小文字まで比較して検索します。例えば、検索文字列に”abc”を指定した場合、”ABC”や”Abc”などは検索されません。

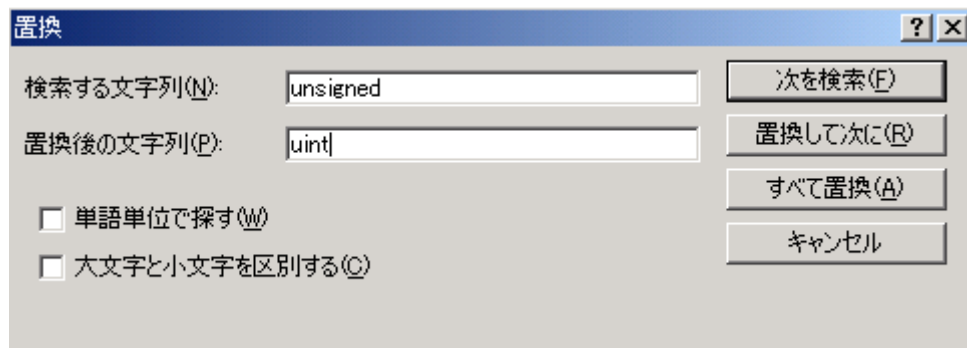
[次を検索]ボタンを押すと、ダイアログボックスを開いた状態で検索が行なわれます。

### ■ 次を検索

[編集]メニューの[次を検索]を選択すると、最後に検索した条件でテキスト検索を続行します。

### ■ 置換

[編集]メニューの[置換]を選択すると、次のようなダイアログボックスが表示されます。



[検索する文字列]に置換対象となるテキストを入力します。

[置換後の文字列]に置き換えるテキストを入力します。

[単語単位で探す]をチェックすると、検索する文字列を単語毎に検索します。例えば、検索する文字列に”rom”を指定した場合、”romwindow”や”from”などの単語は置換されません。

[大文字と小文字を区別する]をチェックすると、検索する文字列の大文字と小文字まで比較します。例えば、検索する文字列に”abc”を指定した場合、”ABC”や”Abc”などは置換されません。

[次を検索]ボタンを押すと、候補が現れるまで検索します。

[置換して次に]ボタンを押すと、検索されたテキストを置換して、次の候補まで検索します。

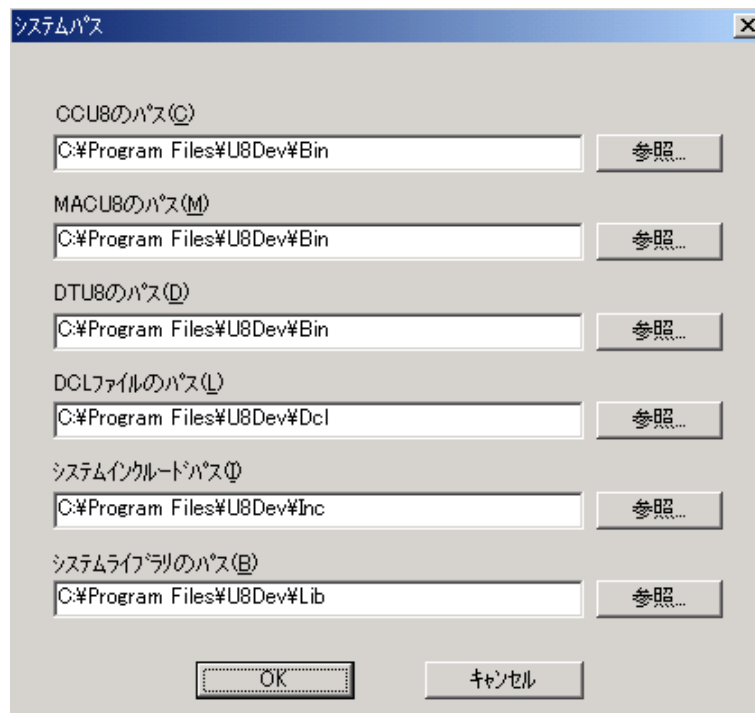
[すべて置換]ボタンを押すと、すべての候補を一括して置換します。

### ■ 行番号でジャンプ

[編集]メニューの[行番号でジャンプ]を選択すると、ファイルの指定行にカーソルを移動することができます。

## 6.3 システムパスの設定

[ファイル]メニューの[環境設定]→[システムパス...]を選択すると、次のようなダイアログボックスが表示され、各ビルドツールのパス設定を行なうことができます。<sup>14</sup>



### ■ CCU8 コンパイラのパス指定

[CCU8 のパス]フィールドには、CCU8 コンパイラの格納ディレクトリを指定します。

### ■ MACU8 アセンブラのパス指定

[MACU8 のパス]フィールドには、MACU8 アセンブラパッケージの格納ディレクトリを指定します。

### ■ DTU8 デバッガのパス指定

[DTU8 のパス]フィールドには、DTU8 デバッガの格納ディレクトリを指定します。

### ■ DCL ファイルのパス指定

[DCL ファイルのパス]フィールドには、DCL ファイルの格納ディレクトリを指定します。

このパスは、RASU8 アセンブラを起動するときに、DCL 環境変数に設定されます。

### ■ システムインクルードのパス指定

[システムインクルードのパス]フィールドには、システムインクルードファイルの検索ディレ

<sup>14</sup> nX-U8 開発ツールのセットアッププログラムでインストールしたときには、インストーラがシステムパスを設定しています。必要に応じて、変更、追加してください。

## 6章 その他の機能

クトリパスを指定します。このフィールドは、複数のパスをカンマ(,)またはセミコロン(;)で区切って指定できます。

このパスは、CCU8 コンパイラを起動するときに、INCLU8 環境変数に設定されます。

### ■ システムライブラリのパス指定

[システムライブラリのパス]フィールドには、システムライブラリファイルの検索ディレクトリパスを指定します。このフィールドは、複数のパスをカンマ(,)またはセミコロン(;)で区切って指定できます。

このパスは、RLU8 リンカを起動するときに、LIBU8 環境変数に設定されます。

## 6.4 その他の環境設定

[ファイル]メニューの[環境設定]→[その他...]を選択すると、次のようなダイアログボックスが表示されます。

その他

エディタ

☒ 内蔵エディタを使用する

フォント設定(E)... ☒ ワードラップを使用する(W) ☒ カラー構文を有効にする(C)

☐ 外部エディタを使用する

外部エディタプログラム(T) C:\Program Files\Hidemaru\Hidemaru.exe ...

ファイルオープン用オプション(O) %FILEPATH%

タグジャンプ用オプション(J) %FILEPATH%

ファイルパス: %FILEPATH% タグジャンプする行番号: %LINE%  
例) ファイルオープン用オプション %FILEPATH%  
タグジャンプ用オプション /L:%LINE% %FILEPATH%

最近使ったファイルの履歴個数(R) 1

最近使ったプロジェクトの履歴個数(P) 1

☐ プロジェクトウィンドウにシステムヘッダファイル表示(H)

☒ レスポンスファイルを削除する(S)

☐ ビルトログを自動的にディスクファイルへ保存する(U)

出力パスまたはファイル名(E) uuu ...

OK キャンセル

[エディタ]フィールドでは、[内蔵エディタを使用する]か、[外部エディタを使用する]かを選択することができます。

[内蔵エディタを使用する]を選択すると、以下の項目が有効となります。

- [フォント設定...]ボタン

このボタンをクリックすると、フォントの設定ダイアログボックスが表示され、エディタウィンドウのフォントを変更することができます。

- [ワードラップを使用する]チェックボックス

このチェックボックスをチェックすると、エディタウィンドウがワードラップで、折り返し表示されます。

- [カラー構文を有効にする]チェックボックス

このチェックボックスをチェックすると、エディタウィンドウで、Cやアセンブリ言語のキーワードをカラーでハイライト表示します。<sup>15</sup>

[外部エディタを使用する]を選択すると、以下の項目が有効となります。

- [外部エディタプログラム]入力フィールド

このフィールドには、外部エディタのファイルパスを指定します。

- [ファイルオープン用オプション]入力フィールド

このフィールドには、登録した外部エディタのファイルオープン用コマンドラインオプション列（ファイルパスおよび任意のオプション）を指定します。

コマンドラインオプション内のオープン対象となるファイルパスには、%FILEPATH%キーワード（すべて大文字）を指定してください。Program Filesのように、ファイルパスに空白文字が含まれる場合は、%FILEPATH%を""（ダブルクォーテーション）で括って指定してください。

IDEU8でソースファイルやヘッダファイルを開く際には、%FILEPATH%をオープン指定されたファイルパスに置き換えて、外部エディタを起動します。

- [タグジャンプ用オプション]入力フィールド

このフィールドには、登録した外部エディタでタグジャンプ（指定ファイルをオープンし、指定行へジャンプ）する場合に、外部エディタへ渡すオプション列（ファイルパス、行番号、任意のオプション）を指定します。

コマンドラインオプション内のオープン対象となるファイルパスには%FILEPATH%キーワード（すべて大文字）を、ジャンプする行番号には%LINE%キーワード（すべて大文字）を指定してください。Program Filesのように、ファイルパスに空白文字が含まれる場合は、%FILEPATH%を""（ダブルクォーテーション）で括って指定してください。

IDEU8でタグジャンプする際には、%FILEPATH%をタグジャンプ対象のファイルパスに置き換え、%LINE%をジャンプ先行番号に置き換えて外部エディタを起動します。

[最近使ったファイルの履歴個数]には、[ファイル]メニューの[最近使ったファイル]に表示される履歴の個数を指定します。

[最近使ったプロジェクトの履歴個数]には、[ファイル]メニューの[最近使ったプロジェクト]に表示される履歴の個数を指定します。

[プロジェクトウィンドウにシステムヘッダファイルを表示]をチェックすると、システムイン

<sup>15</sup> 大きなファイルのカラー構文解析処理はエディタの軽快さがなくなります。必要性和好みに応じて、チェックしてください。



## 6章 その他の機能

クルードパスで指定したパスにあるヘッダファイルを、プロジェクトウィンドウの[Include]フォルダに表示します。

[レスポンスファイルを削除する]をチェックすると、IDEU8 がビルド処理時に作成する各種ツールのレスポンスファイルを削除します。

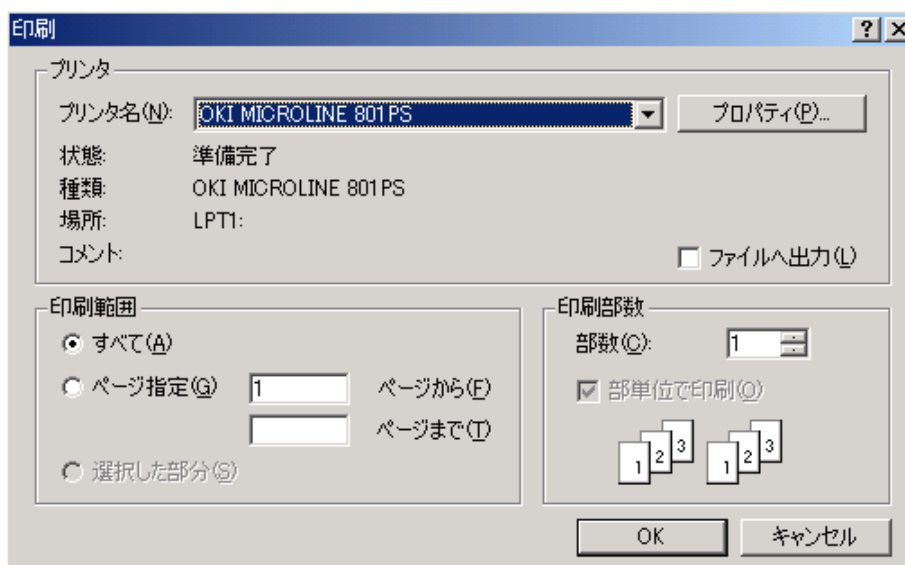
[ビルドログを自動的にディスクファイルへ保存する]をチェックすると、毎ビルド処理時に指定されたファイルに自動的にアウトプットウィンドウの内容をテキスト形式で保存します。（2 度目以降のビルド時には、ファイルをオーバーライトして保存します。）

[出力パスまたはファイル名]指定フィールドにディレクトリパスを指定した場合には、指定されたパスに拡張を.log にしたプロジェクト名で保存します。ファイル名を指定した場合には、指定されたファイル名で保存します。

### 6.5 ファイルの印刷

IDEU8 は、エディタウィンドウに表示されているソースファイルをプリンタへ出力することができます。

[ファイル]メニューの[プリント...]を選択すると、次のようなダイアログボックスが表示されます。

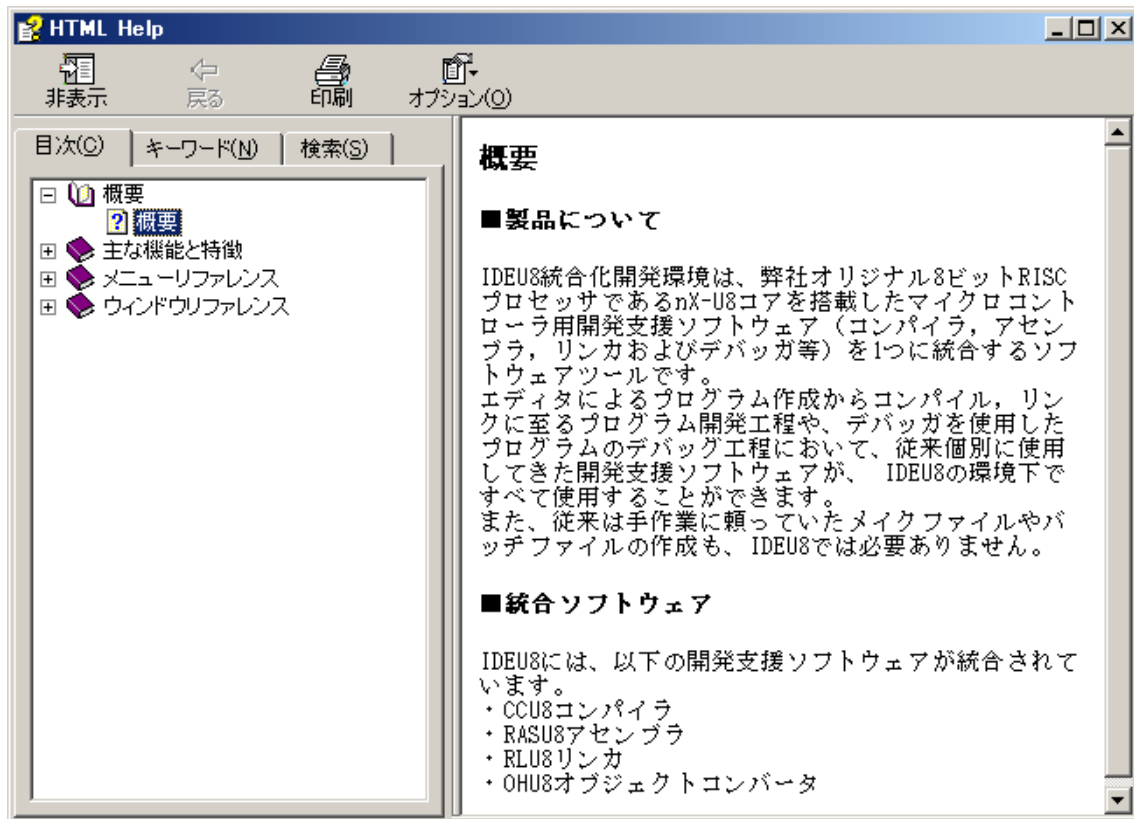


印刷範囲および印刷部数を設定して[OK]ボタンを押すと、印刷を開始します。

### 6.6 オンラインヘルプ

IDEU8 には、HTML ヘルプフォーマットのオンラインヘルプが用意されています。

[ヘルプ]メニューの[トピックの検索]を選択するか、F1 キーを押すと、IDEU8 のヘルプウィンドウが表示されます。



## 6.7 バージョン情報の表示

[ヘルプ]メニューの[IDEU8 のバージョン情報...]を選択すると、次のようなダイアログボックスが表示され、IDEU8 のバージョンと各種ツールのバージョンを確認できます。



索引

*C*

CCU8 コンパイラ

/CT オプション .....	24
/D オプション .....	27
/far オプション .....	18
/Fa オプション .....	19
/Ff オプション .....	22
/I オプション .....	19
/J オプション .....	21
/LE オプション .....	24
/LP オプション .....	24, 25
/ML オプション .....	17
/MS オプション .....	17
/near オプション .....	18
/NOWIN オプション .....	21
/Oa オプション .....	21
/Od オプション .....	20
/Og オプション .....	20, 21
/Ol オプション .....	20, 21
/Om オプション .....	21
/Ot オプション .....	21
/PC オプション .....	24
/PF オプション .....	22
/SD オプション .....	18
/SL オプション .....	18
/SS オプション .....	21
/ST オプション .....	21
/SYS オプション .....	22
/T オプション .....	17
/Wa オプション .....	23
/Wc W5031 オプション .....	23
/Wc オプション .....	23
/W オプション .....	22
/Za オプション .....	18
/Zg オプション .....	24
追加オプション .....	18

*D*

DCL 環境変数 .....	41
DTU8 デバッガ .....	36

*I*

INCLU8 環境変数 .....	42
-------------------	----

*L*

LIBU8 環境変数 .....	42
------------------	----

*O*

OHU8 オブジェクトコンバータ

/D オプション .....	29
/H オプション .....	29
/S オプション .....	29

*R*

RASU8 アセンブラ

/BNVRAMP オプション .....	26
/BNVRAM オプション .....	26
/BRAM オプション .....	26
/BROM オプション .....	26
/CD オプション .....	18
/DEF オプション .....	27
/DF オプション .....	18
/DN オプション .....	18
/D オプション .....	18
/E オプション .....	25
/G オプション .....	18
/I オプション .....	19
/KE オプション .....	18
/L オプション .....	25
/ML オプション .....	17
/MS オプション .....	17
/NO オプション .....	22
/O オプション .....	19
/PL オプション .....	25

/PR オプション .....	25	object_files フィールド .....	16
/PW オプション .....	25	追加オプション .....	29
/R オプション .....	25		
/SD オプション .....	18	<b>あ</b>	
/SF オプション .....	25	アウトプットウィンドウ .....	36
/SL オプション .....	18	アンインストール .....	4
/S オプション .....	25	インストール .....	4
/T オプション .....	25	エディタウィンドウ .....	39
/WA オプション .....	23	オプション	
/WE オプション .....	23	コンパイル／アセンブル .....	17
/WP オプション .....	23	ターゲット .....	27
/WR オプション .....	23	オプションセット .....	33
/WS オプション .....	23	オンラインヘルプ .....	44
/WT オプション .....	23		
/X オプション .....	25	<b>か</b>	
追加オプション .....	18	クリップボード .....	39
RLU8 リンカ		クロスリファレンスリスト .....	25
/BIT オプション .....	30	警告タブ	
/CC オプション .....	28	コンパイル／アセンブル .....	22
/CODE オプション .....	30		
/COMB オプション .....	30	<b>さ</b>	
/DATA オプション .....	30	システムインクルードパス .....	44
/D オプション .....	28	システムパス .....	41
/EXC オプション .....	28	シンボルリスト .....	25
/NVBIT オプション .....	30		
/NVDATA オプション .....	30	<b>た</b>	
/NVRAMP オプション .....	32	ターゲット MCU .....	17
/NVRAM オプション .....	32	デバッグ	
/ORDER オプション .....	30	開始 .....	36
/PDIF オプション .....	32	動作環境 .....	4
/RAM オプション .....	32		
/ROMWIN オプション .....	32	<b>は</b>	
/ROM オプション .....	32	バージョン情報 .....	45
/SD オプション .....	28	パブリックシンボルリスト .....	28
/STACK オプション .....	28	表示メニュー	
/S オプション .....	28	アウトプット .....	36
/TABLE オプション .....	30	ビルド .....	35
libraries フィールド .....	16	選択ファイルのビルド .....	36
		中止 .....	36
		ビルドログ .....	25

ファイルメニュー	オプション.....17
上書き保存.....10	プロジェクトメニュー
環境設定 .....41, 42	クリーン .....36
最近使ったファイル .....43	デバッグ .....36
最近使ったプロジェクト .....43	ビルドの中止 .....36
新規作成... .....10, 38	ファイルの追加... .....10
名前を付けて保存... .....11	ヘルプメニュー
開く .....11	IDEU8 のバージョン情報... .....45
プリント... .....44	トピックの検索 .....44
プロジェクトの保存 .....10	編集メニュー.....39
フォント .....43	
プロジェクト	ら
作成.....10	リスティングタブ
保存.....10	コンパイル／アセンブル .....23
プロジェクトウィンドウ .....11	リビルド.....35
プロジェクトウィンドウ .....11	レスポンスファイル .....44
ポップアップメニュー.....12	わ
プロジェクトファイル	ワードラップ.....43
読み込み .....11	
プロジェクトメニュー	

IDEU8  
ユーザーズマニュアル

---

SQ003096E301

2011 年 10 月      第 9 版発行

©2008-2011 LAPIS Semiconductor Co., Ltd.

---