

第3章 統合開発環境 Embedded Workbench の使い方

Cプログラムの作成と デバッグの方法

渡辺 明禎
Akiyoshi Watanabe

統合開発環境IAR Embedded Workbenchには、機能に制限のない製品版、使用が30日に限定されるIAR Embedded Workbench Evaluation版、出力するコードの大きさが4Kバイトまでに制限される(C/C++の場合)IAR Embedded Workbench KickStart版の3種類があります。

CD-ROMに収録されているのはKickStart版のV3.41Aです。

使用する開発環境の構造

IAR Embedded Workbench KickStart for MSP430 (EW430-KS, 一般にKickStart versionと呼ばれる。本特集ではEW430と呼ぶ)は、MSP430マイコン・シリーズの統合開発環境です。

言語としては、アセンブラ、C/C++コンパイラが使えます。C/C++の場合、出力できるコードの大きさが4Kバイトに制限されているものの、アセンブラでは制限なくコードを生成できるなど、利用価値の非常に高いツールです。

開発言語のほかに、プロジェクト・マネージャ、エディタ、ビルダ、デバッガ・ツールを含んでおり、ハードウェアやシミュレータ上でリアルタイムのデバッグ環境を提供しています。

● IAR Embedded Workbenchのセットアップ

IAR Embedded Workbenchのセットアップは以下の手順で行います。

- ① slac050k.zip(付録CD-ROMに収録済み)からFET_R452.exeを解凍し実行する
- ② MSP430 Development Tool-Setupダイアログで[Setup]ボタンをクリックする

- ③ IAR Systems Product Setupダイアログで[Next]ボタンをクリックする
- ④ IAR Systems Software License Agreement画面で[Accept]ボタンをクリックする
- ⑤ Select Destination Folder画面で[Next]ボタンをクリックする。インストール・フォルダを変える場合はここで指定する
- ⑥ Installation Type画面で[Next]ボタンをクリックする。インストール・ファイルを選択したい場合はCustomを選択する
- ⑦ Select Program Folder画面で[Next]ボタンをクリックする
- ⑧ Review Settings Before Copying Files画面で[Next]ボタンをクリックする
- ⑨ Setup Complete画面で[Finish]ボタンをクリックする
- ⑩ PCをリスタートし、セットアップを終了する

● IAR Embedded Workbenchの実行

EW430に慣れるためには、実際に使用してみるのが早道です。サンプル・プログラム「Flashing the LED」でLEDを点滅動作してみます。なお、ここでは、¥4270フォルダに特集関連プログラムが存在していると仮定して解説します。EW430のより詳しい使い方は後述します。

まず、PCにJTAGインターフェースを接続します。次にJTAGインターフェースにMSP430-CQを接続します。ここでは、第1章で紹介したマザーボードを使っています。

EW430を起動します。図1のような画面が表示されます。起動できたら、以下の方法でサンプル・プログラムを実行します。

Keywords

IAR Embedded Workbench, KickStart, EW430

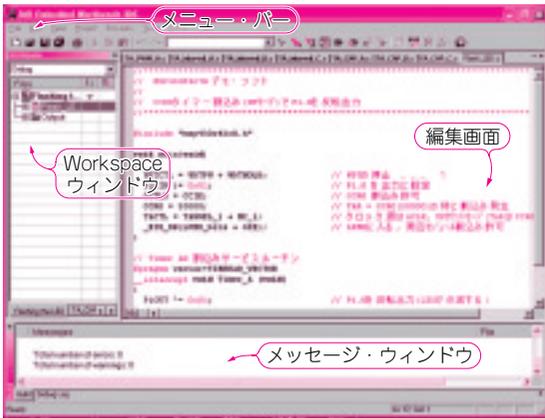


図1 統合開発環境 IAR Embedded Workbench KickStart for MSP430(EW430)の実行画面(編集モード)
C/C++では出力できるコードの大きさが4Kバイトに制限されている。しかし、アセンブラでは制限がない

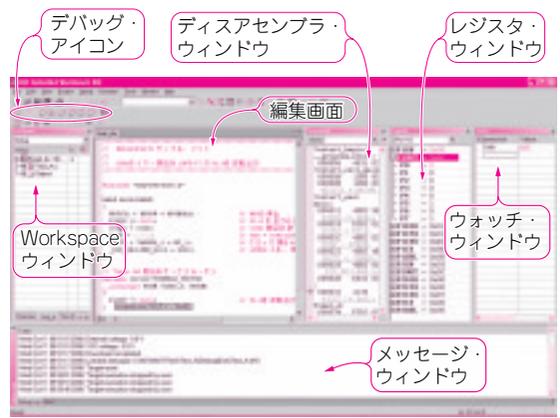


図2 統合開発環境 EW430のデバッグ画面
アセンブラ・ソース・コードによるステップ動作、さまざまなデバッグ情報などを表示可能

- ① メニューから [File] - [Open] - [Workspace] を選択する
- ② Open Workspace 画面で ¥4270¥4270TA ¥4270TA.eww を選択する
- ③ Workspace ウィンドウの下部にある Flashing the LED タブを選択する。これは、Timer_A3 モジュールの C 言語によるサンプルとなる
- ④ Workspace ウィンドウの Flash_LED.c をダブルクリックし、ソース・プログラムを編集画面に表示させる
- ⑤ JTAG インターフェースをパラレル・インターフェースに設定しているの、もし USB インターフェースを使う場合は、プロジェクト名をクリックし、メニューの [Project] - [Options] で、Debugger → FET Debugger の Setup タブの Connection で自分の開発環境に合わせ、TI USB FET を設定する
- ⑥ メニューの [Project] - [Debug] を選択し、

プログラムをデバッグ・モードにする

これにより、マイコンのフラッシュROMは消去され、開発したオブジェクト・コードがフラッシュROMに書き込まれます。

EW430の画面は図2のように変わり、アセンブラ・ソース・コードによるステップ動作、さまざまなデバッグ情報などを表示することができます。

メニューの [Debug] - [Go] を選択し、マイコンに書き込んだコードを実行します。問題がなければ、ターゲット・ボードのLEDが点滅するはずで

- ⑦ メニューの [Debug] - [Stop Debugging] を選択すると、デバッグ動作が終了し、編集モードに戻る。ただし、ターゲット・ボードはスタンドアロンで動作を継続している
- ⑧ メニューの [File] - [Exit] により EW430 を終了する

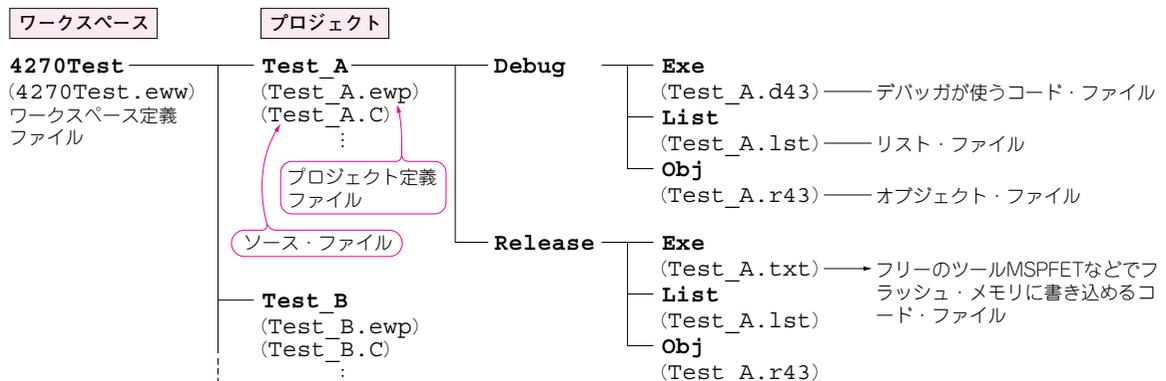


図3 統合開発環境 EW430を使ったMSP430の開発におけるフォルダ、ファイルの構成例
開発の単位はプロジェクト。プロジェクトはソース・ファイルから構成される。プロジェクトをまとめたものがワークスペース