

概要

このチュートリアルでは、Altium Designer を使って組込みプロジェクトを作成する方法を説明します。

Tutorial

TU0122 (v2.0) April 21, 2008

このチュートリアルは C とアセンブラのプログラミングの経験があり、組込みソフトウェアの基本的な知識がある方を対象として作成されています。このチュートリアルは Altium Designer に含まれる組込み開発用ツールである TASKING ツールの概要に続き、組込みプロジェクトへ新規または既存のソースファイルを追加、編集、ビルドする方法を説明します。組込みソフトウェアプロジェクトは、一般に FPGA プロジェクトのサブプロジェクトとして使用され、ビルド後に FPGA デバイスにダウンロードされ実行されます。

このチュートリアルで使用する例題は C の Hello World プログラムです。他のサンプルは、`\Examples\NanoBoard Common\Processors Examples` フォルダに保存されています（インストールされたフォルダからの相対パス）。

組込みソフトウェアツール

Altium Designer の TASKING では、TSK51x/TSK52x、TSK80x、TSK165x、PowerPC、TSK3000、MicroBlaze、Nios II、ARM などをターゲットにして、アプリケーションを作成 / コンパイル / アセンブル / リンクすることができます。図 1 に TASKING ツールセットのすべてのコンポーネントとその入出力ファイルが示されています。

C コンパイラ、アセンブラ、リンカ、デバッガはターゲット毎に用意されていますが、ライブラリはターゲットに依存しません。次ページの図の太字の名前はツールの実行ファイルの名前です。target の文字列を使用する MPU の名前に置き換えることにより各ツールの名前になります。すなわち **cppc** は PowerPC 用 C コンパイラ、**c3000** は TSK3000 用 C コンパイラ、**as165x** は TSK165x 用のアセンブラです。

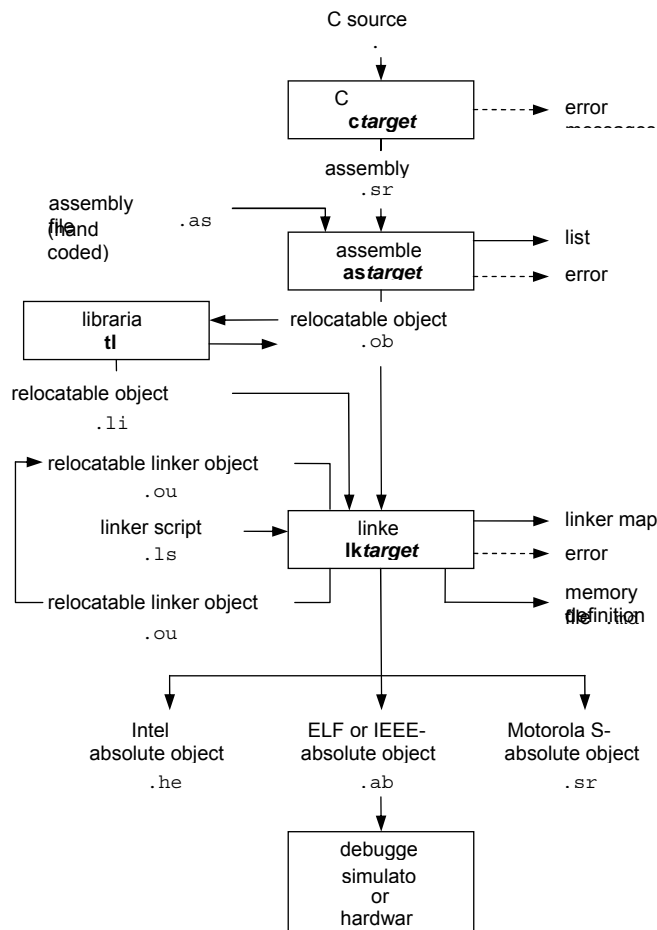


図 1: ツールセットの概要

TASKING ツールセットで使用するファイルの種類

拡張子	説明
ソースファイル	
.c	C ソースファイル
.asm	ユーザの作成したアセンブラ・ソースファイル
.lsl	リンカ・スクリプトファイル
ツールにより生成されるソースファイル	
.src	C コンパイラにより作成されるアセンブラ・ソースファイル(マクロを含まない)
オブジェクトファイル	
.obj	アセンブラより出力されるリロケータブル・オブジェクトファイル
.lib	オブジェクトファイルのアーカイブ
.out	リンカで作成されるリロケータブルファイル
.abs	リンカの一部であるロケータで作成される IEEE-695 または ELF/DWARF 2 アブソリュート・オブジェクトファイル
.hex	インテル Hex フォーマットのアブソリュート・オブジェクトファイル
.sre	モトローラ S フォーマットのアブソリュート・オブジェクトファイル
リストファイル	
.lst	アセンブラ・リストファイル
.map	リンカ・マップファイル
.mcr	MISRA C レポートファイル
.mdf	メモリ定義ファイル
エラーリストファイル	
.err	コンパイラ・エラーメッセージファイル
.ers	アセンブラ・エラーメッセージファイル
.elk	リンカ・エラーメッセージファイル

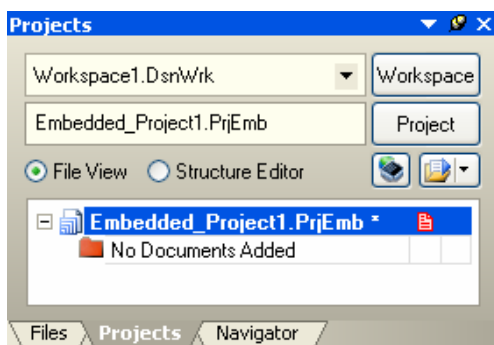
表 1: ファイル拡張子

組み込みプロジェクトの作成

Altium Designer を使用するにはまずプロジェクトを作成する必要があります。プロジェクトにより、ソースファイルや出力されたファイルを楽に管理することができます。組み込みソフトウェアの場合 Embedded Software project を作成する必要があります。

組み込みソフトウェアプロジェクトを作成するには:

1. メニューから **File » New » Project » Embedded Project** を選択するか **Files** パネルの **New** の項目で **Blank Project (Embedded)** をクリックします。このパネルが表示されていない場合は、デザイン・マネージャパネルの下にある **Files** タブをクリックしてください。
2. **Projects** パネルが新規のプロジェクトファイル Embedded_Project1.PrjEmb を表示します。



3. **File » Save Project As** を選択して、新規のプロジェクトファイル (拡張子 .PrjEmb) の名称を変更します。ハードディスク内のプロジェクトの保存先を指定します。ファイル名 GettingStarted.PrjEmb を入力し、**Save** をクリックします。

新しいソースファイルをプロジェクトに追加

あたらしいソースファイル(C、アセンブラ、その他のテキストファイル)をプロジェクトに追加するには下記の手順で行います。

1. **Projects** パネルの GettingStarted.PrjEmb を右クリックして、**Add New to Project » C File** を選択します。新しいCソースファイル Source1.C が **Projects** パネル内の Source Documents というフォルダの下で、Embedded Software project に追加されます。テキストエディタが開き入力ができる状態になります。
2. 必要なソースコードを入力します。このチュートリアルでは下記のコードを入力します。

アセンブラファイルの時はC Fileの代わりにAssembly File、テキストファイルのときはText Documentを選択してください

```
#include <stdio.h>

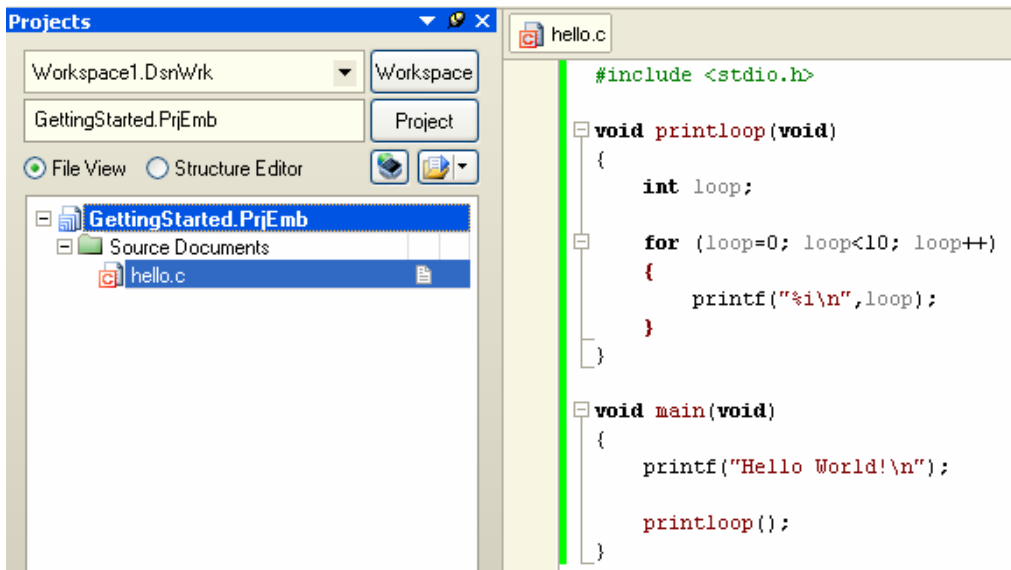
void printloop(void)
{
    int loop;

    for (loop=0; loop<10; loop++)
    {
        printf("%i\n", loop);
    }
}

void main(void)
{
    printf("Hello World!\n");

    printloop();
}
```

3. **File » Save As** を選択して、ソースファイルを保存します。ハードディスク内の適当な場所に移動し、ファイル名を入力する場所に hello.c とタイプして **Save** をクリックします。
4. **Projects** パネルでプロジェクト GettingStarted.PrjEmb を右クリックし、**Save Project** でプロジェクトを保存します。プロジェクトは次のように表示されます。



既存のファイルをプロジェクトに追加

既存のファイルをプロジェクトに追加するには下記の手順で行います。

1. **Projects** パネルでプロジェクト名を右クリックし、**Add Existing to Project** を選択。 *Choose Documents to Add to Project* ダイアログが表示されます。
2. 追加したいファイルを選択し、**Open** をクリックします。
3. ソースファイルがプロジェクトに追加され、**Projects** パネルに表示されます。 ファイル名をダブルクリックするとテキストエディタでファイルの中身を確認したり編集したりすることができます。
4. プロジェクトを保存します。(**Projects** パネルのプロジェクト名を右クリックし **Save Project** を選択します。

組込みプロジェクトオプションを設定する

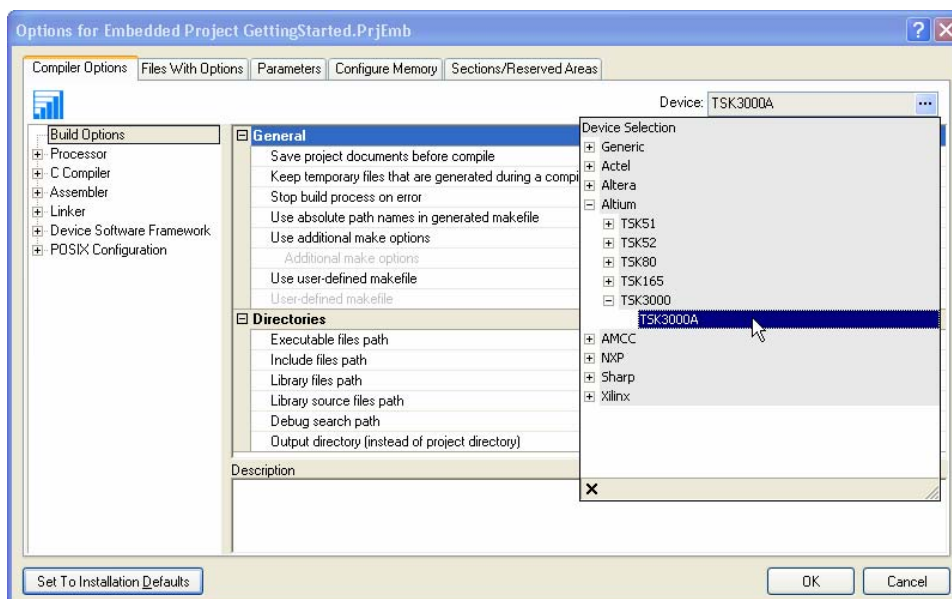
Altium Designer の組込みプロジェクトには組込みプロジェクトの一式があります。 プロジェクトにファイルを追加しアプリケーションを作成 (この例では hello.c) したら、次のステップでプロジェクトをビルドします。

- デバイス (ターゲットプロセッサ) の選択 (その結果としてのツールセットの選択)
- ツールセット内の C コンパイラ、アセンブラ、リンカなどのツールオプションを指定 (異なるツールセットは異なるオプションを持つことがあります)。

デバイス (ターゲットプロセッサ) の選択

組込みプロジェクトでは、最初に組込みプロジェクトをビルドしようとするデバイスを指定する必要があります。

1. **Projects** パネルの GettingStarted.PrjEmb を右クリックして、**Project Options** を選択します。 または、**Project » Project Options** をメニューから選択します。
Options for Embedded Project ダイアログが表示されます。



2. **Compiler Options** タブで **Device** を選択します。製造ベンダに基づいて選択できます。互換製品を選ぶこともできます。製造ベンダからデバイスを選んだ場合は、正しいプロセッサタイプが自動的に選択されます。互換製品を選んだ場合は、ターゲットプロセッサのタイプは手入力してください。

ターゲットプロセッサのタイプを手入力は、以下のとおりです（互換デバイス）。

3. ダイアログ左側の **Processor** の項目を広げ **Processor Definition** を選択します。
4. パネルの右側の **General** を広げ、**Select processor** でターゲットプロセッサを選択します。
5. **OK** をクリックして、新しいデバイスを選びます。

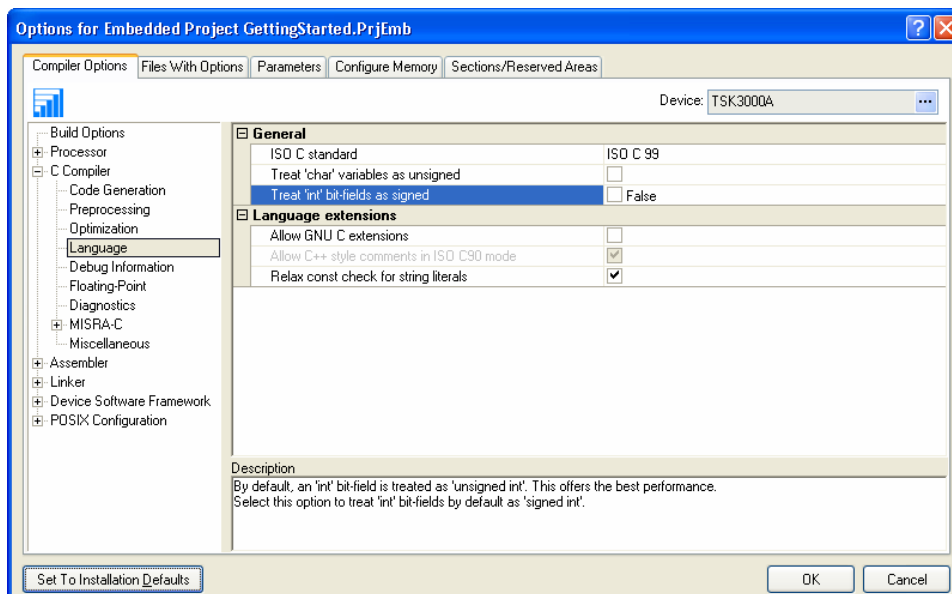
ツール・オプションの選択

プロジェクト内のファイルに共通なオプションとファイル毎のオプションを設定することが出来ます。

プロジェクト内のファイルに共通のオプション

1. **Projects** パネルの `GettingStarted.PrjEmb` を右クリックして、**Project Options** を選択します。または、**Project » Project Options** をメニューから選択します。

Options for Embedded Project ダイアログが表示されます。



2. パネルの左側の **C Compiler** を広げます。ここには、いくつかのページがあり、C コンパイラの設定ができます。
3. ダイアログの右側で各オプションの値を設定します。これをすべてのページで行います。

各ツールの **Miscellaneous** ページの **Command line options** フィールドでオプションの設定をどのようにコマンドラインに反映されたか確認することができます。

4. アセンブラとリンクも 2 と 3 を繰り返し設定します。
5. 設定が終了したら **OK** をクリックします。

Altium Designer は組み込みプロジェクトのオプションに基づき、組み込みアプリケーションをビルドするときに使用される *makefile* を作成します。


個々のファイルのオプション

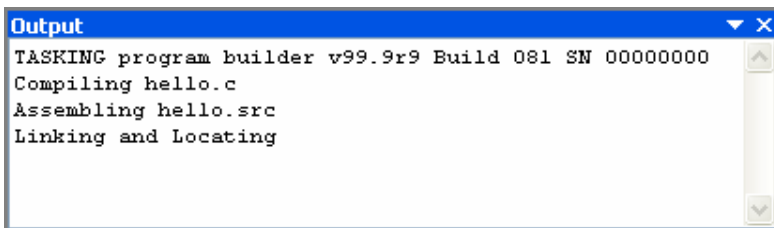
1. **Projects** パネルの `hello.c` を右クリックして、**Document Options** を選択します。または、**Project » Document Options** をメニューから選択します。 *Options for Document* ダイアログが表示されます。

ステップ 2 から 5 まではプロジェクト・オプションと同じです。 *Options for Embedded Project* の **Files With Options** タブで、どのファイルがプロジェクト・オプションと違う設定をしているか表示されます。 このオプションタブのファイル名を右クリックすると、個々のファイルに設定したオプションを他のファイルにコピーしたり、削除したりすることのできるメニューが表示されます。

組み込みアプリケーションのビルド

以上でビルドの準備ができました。

1. **Project » Compile Embedded Project GettingStarted.PrjEmb** を選択するか、 ボタンをクリックします。TASKING プログラム・ビルダは組み込みプロジェクト内で、時間が無効か、前回のビルドから変更があったファイルのコンパイル、アセンブル、リンクとロケートを行います。生成されるファイルはアブソリュート・オブジェクトファイル `GettingStarted.abs` です。
2. **Output** パネル (**View » Workspace Panels » System » Output**) にビルド結果が表示されます。



Compiling a single source file

単一のソースファイルをコンパイルしたいときは下記の手順で行います。

1. コンパイルしたいソースファイル(`hello.c`)名を右クリックし **Compile Document hello.c** を選択するか、テキストエディタを開いて **Project » Compile Document hello.c** を選択します。
2. Messages パネルを開いて、コンパイル中に発生したコンパイル・エラーを確認します。 **View » Workspace Panels » System » Messages** を選択するか、**System » Messages** を **Panels** から選択します。
3. エラーがあれば修正し、ファイルを保存します。

プロジェクト全体のリビルド

個々のファイルの日付と無関係に組み込みプロジェクト全体をビルドしたいときは下記の手順で行います。

1. **Project » Recompile Embedded Project GettingStarted.PrjEmb** を選択します。
2. TASKING プログラム・ビルダは組み込みプロジェクトのすべてのソースファイルのコンパイル、アセンブル、リンクとロケートを行います。

以上でオブジェクトファイル `GettingStarted.abs` のデバッグを行うことができます。

組み込みアプリケーションのデバッグ

組み込みプロジェクトのビルドが終了したら、完成したアブソリュート・オブジェクトファイルをシミュレータでデバッグすることができます。

デバッグを開始するには、1つまたはそれ以上のソース行を実行しなくてはなりません。

- ソースレベルまたはインストラクションレベルの step オプション(**Debug** » **Step Into, Step Over**) を選択し、ソースをステップさせるか、**Debug** » **Run** を選択して、シミュレーションの起動します。

青の行は現在の実行位置を表します。

レジスタ、ローカル変数、メモリ、ブレークポイントをなど調べるパネルを開くには、

- View** » **Workspace Panels** » **Embedded** » (*a_panel*(パネルの名前)) を選択します。

デバッグを終了するには

- Debug** » **Stop Debugging** を選択します。

ブレークポイントの設定

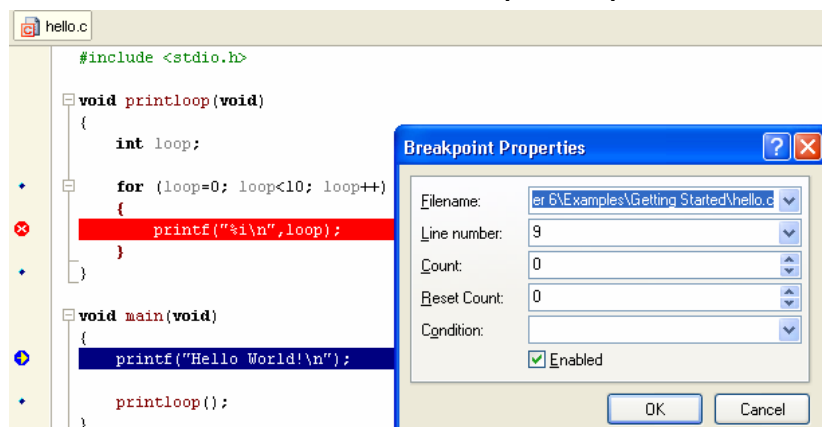
ソース行が開かれている状態でブレークポイントを設定することができます。ブレークポイントを設定することのできる行には小さい青い点が表示されています。

- ソース行の左の余白をクリックすることでブレークポイントをオン/オフすることができます。

ブレークポイントには赤い十字の入った丸印が表示され行が赤でマークされます。

ブレークポイントのプロパティを変更するには、

- ブレークポイントを右クリックして **Breakpoint Properties...** を選択します。



ブレークポイントを無効(disable)、または有効(enable)にするには、

- ブレークポイントを右クリックし **Disable Breakpoint** (Disable されているときは **Enable Breakpoint**) を選択します。

Disable されているブレークポイントは緑でマークされます。

Breakpoint Panel で Disable されているブレークポイントも含めすべてのブレークポイントとそのプロパティを確認することができます。

- View** » **Workspace Panels** » **Embedded** » **Breakpoints** を選択します。

式の評価

Evaluate パネルを使用して式の評価を行うことができます。

- View** » **Workspace Panels** » **Embedded** » **Evaluate** を選択し **Evaluate** パネルを開きます。

- 左上の入力欄に評価したい式を入力して **Evaluate** をクリックします。

式とその値が **Evaluate** パネルに表示されます。コード中の変数が変更されるたびに **Evaluate** ボタンをクリックします。

式を連続してウォッチするには、*watch* を設定してください。

- Add Watch** を選択します。

新しい式とその値が **Watches** パネルに表示されます。 **Watches** パネルの値が、コードの実行に従って連続的に更新されます。

式の評価方法はオブジェクトファイルに含まれるデバッグ情報の量に大きく依存します。また、最適なレベルはデバッグのやりやすさに大きく影響します。

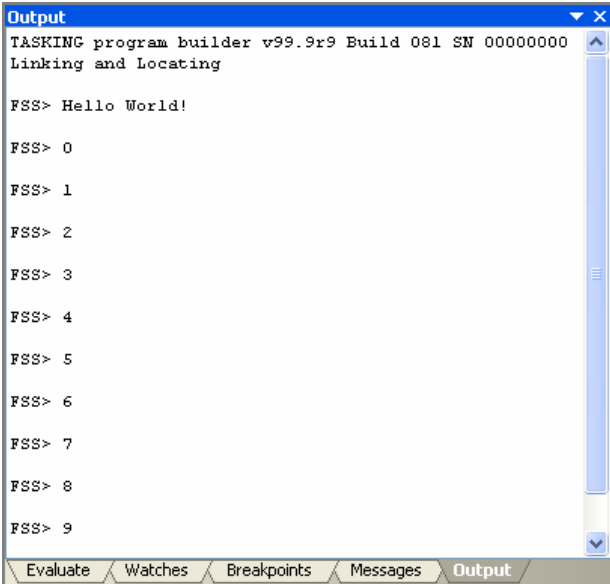
もうひとつの方法 : Debug » Add Watch を選択し、式を入力して OK をクリックします。

出力の確認

組み込みアプリケーションによって生成された出力を表示することができます。必ずデバッグモードになっている必要があります。Output パネルを開くには、

- View » Workspace Panels » System » Output を選択します。

Output パネルが開き、組み込みアプリケーションの出力が表示されます。

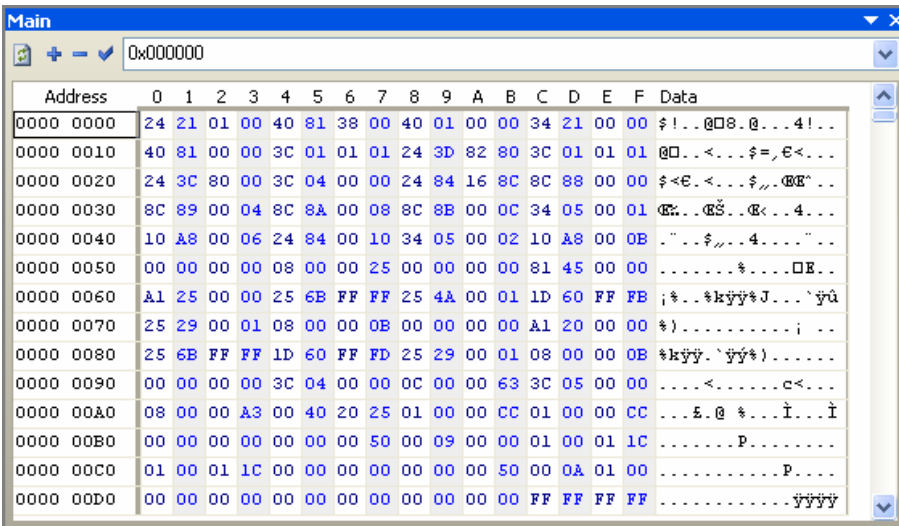


メモリの確認

デバッグモードではいくつかのメモリ・ウィンドウを開きメモリの内容の確認をすることができます。必ずデバッグモードになっている必要があります。開くことのできるメモリ・ウィンドウは選択しているターゲット・プロセッサにより異なります。プログラムメモリ・ウィンドウを開く手順は以下のとおりです。

- View » Workspace Panels » Embedded » Main を選択します。

Program Memory window が開きコード・メモリの内容を表示します。



- 一番上の編集フィールドに確認したいメモリの先頭アドレスを入力します。

更新履歴

Date	Version No.	Revision
20-Jan-2004	1.0	New product release
22-Apr-2005	1.1	Changes/updates according to SP3 release.
08-Jun-2005	1.2	Changes/updates according to SP4 release.
01-Nov-2005	1.3	Changes/updates according to Altium Designer 6 release.
09-Nov-2006	1.4	Changes/updates according to Altium Designer 6.6 release and information on Output panel expanded. Code example expanded to include a simple loop statement as well as the hello world statement.
12-Mar-2007	1.5	Changes/updates according to Altium Designer 6.7 release.
21-Apr-2008	2.0	Updates for Altium Designer Summer 08 release.

ソフトウェア、ハードウェア、文書、および関連資料

Copyright © 2003 Altium Limited. All rights reserved.

以下の注意書きとともに提供される文書とその情報は、様々な形による国内、海外の知的財産権の保護 - 著作権の保護を含むがそれに限定されない - が目的です。この注意書きの閲覧者には、非独占的なライセンスが付与されており、このような文書とその情報を、その用途について規定している使用許諾契約書（エンドユーザライセンスアグリーメント）に記載の目的のために使用することができます。いかなる場合においても、あなたにライセンスされた文書から、あるいはその他の手段を利用して、リバースエンジニア、逆コンパイル、複製、配布、派生物の作成を行うことは、明白に規定された同意書による許諾を得ない限りできません。かかる制限条項が遵守されない場合、罰金や実刑を含む民事罰と刑事罰の対象となることがあります。しかしながら、バックアップの目的に限り、提供される文書のまたは情報を一個だけ記録に残し、オリジナルコピーが不能の場合のみ、その複製にアクセスし、利用することは許可されます。Altium、Altium Designer、Board Insight、CAMtastic、CircuitStudio、Design Explorer、DXP、Innovation Station、LiveDesign、NanoBoard、NanoTalk、OpenBus、Nexar、nVisage、P□CAD、Protel、SimCode、Situs、TASKING、Topological Autorouting、およびそれぞれに対応するロゴは、Altium Limited またはその子会社の商標または登録商標です。本書に記載されているそれ以外の登録商標や商標はそれぞれの所有者の財産であり、商標権を主張するものではありません。