

Code Composer Studio™ クイックスタート

Stellaris® Code Composer Studio™ 用開発・評価キット

Stellaris 開発・評価キットによって、Texas Instruments 製 Code Composer Studio 開発ツールを使用する Stellaris マイクロコントローラの設計を低コストで始めることができます。評価キットは、完全な評価ターゲットとして機能するほか、外部 Stellaris デバイスとのデバッガ・インターフェイスとして使用することもできます。

要件

- Microsoft® Windows 7 上で動作する、USB インターフェイスを搭載した PC
- 『Stellaris Evaluation Kit Documentation and Software』DVD または開発キットに同梱の『Code Composer Studio』DVD



注意: 基板上のインサーキット・デバッグ・インターフェイス (ICDI) で使用されている FT2232 デバイスには、電気的な問題があります。USB ハブによってはデバイスの誤動作を招くこともあり、データ列挙の失敗やデータ転送の異常などの現象が発生します。基板上のICDI を使用して問題が発生した場合は、ご使用しているPC の USB ポートにUSB ケーブルを直接接続してください。

Code Composer Studio

このクイックスタートでは、Code Composer Studio 開発ツールを Stellaris 評価・開発ボードへインストールする方法、および本ツールを使用してサンプル・アプリケーションをビルド、実行する方法を示します。

ステップ 1: Code Composer Studio のインストール

1. PC の DVD-ROM ドライブに『Evaluation Kit Documentation and Software』DVD または『Code Composer Studio』DVD を挿入します。オートプレイが有効になっている場合、index.htm ファイルがデフォルトの Web ブラウザで自動的に開きます。オートプレイが無効になっている場合は、Windows Explorer を使用して DVD を開きます。
2. 『Evaluation Kit』DVD 上で [Tools] ボタンをクリックし、[Code Composer Studio] のロゴをクリックするとセットアップ・プログラムが起動します。

注: Code Composer Studio のインストーラ実行ファイルは、DVD 上の他のインストーラ実行ファイルも利用しています。そのため、Code Composer Studio ロゴのクリックで Code Composer Studio をインストールするには、使用する Web ブラウザがファイルをハード・ドライブに保存することなく実行できる機能をサポートしている必要があります。この機能をサポートしていない場合は、Windows Explorer を使用して、DVD の ¥Tools¥CCS¥ ディレクトリから Code Composer Studio インストーラ (setup_CCS_n.n.n.nnnnn.exe) を実行する必要があります。

Code Composer Studio™ クイックスタート

『Code Composer Studio』 DVD の場合は、インストーラ・ダイアログの指示に従います。

3. Code Composer Studio インストール・プログラムの指示に従います。[Complete Feature Set] インストールまたは [Custom] インストールを選択します。[Custom] インストールを選択した場合、[Stellaris® ARM® Cortex™-M MCUs] は必ず選択してください。他のすべてのオプションは、デフォルト値のままにしておきます。

ステップ2: StellarisWare® パッケージのインストール

C 言語ベースのペリフェラル・ドライバがフルセットで提供されており、すべてのペリフェラルと Stellaris デバイスの機能をカバーしています。StellarisWare パッケージには、Code Composer Studio を含めたさまざまなサンプル・アプリケーションが収録されており、また Stellaris をサポートする主なツール・ベンダー用のプロジェクト・ファイルも含まれています。StellarisWare コンポーネントをインストールするには、次の手順を実行します。

1. 『Evaluation Kit Documentation and Software』 DVD または『Development Kit Documentation and Software』 DVD で [Tools] タブを開きます。

注: Windows Explorer 等を使用して DVD を操作している場合は、Tools¥StellarisWare¥ ディレクトリを開きます。

2. DVD の StellarisWare セクションの横にある [Install] リンクをクリックし、StellarisWare インストーラを実行します。StellarisWare を手動でインストールする場合は、Tools¥StellarisWare ディレクトリにある自己解凍形式 ZIP ファイルのインストーラを使用します。この ZIP ファイルの中身は、WinZip などの ZIP ファイル解凍ユーティリティを使用して手動で解凍することもできます。

3. StellarisWare ドキュメントを参照するには、インストール・ディレクトリに移動し、『Stellaris Peripheral Driver Library User's Guide』の PDF ファイルをクリックします。

注: StellarisWare の最新バージョンは、www.ti.com/stellaris でご確認ください。

ステップ3: LM4F デバイス用 Stellaris ドライバのインストール

カスタム・アプリケーションをデバッグしてマイクロコントローラのフラッシュ・メモリにダウンロードし、Stellaris ベースのインサーキット・デバッグ・インターフェイス (ICDI) を使用して仮想 COM ポート接続を利用するには、以下のドライバをホスト・コンピュータにインストールします。

- Stellaris Virtual Serial Port
- Stellaris ICDI JTAG/SWD
- Stellaris ICDI DFU

Code Composer Studio™クイックスタート

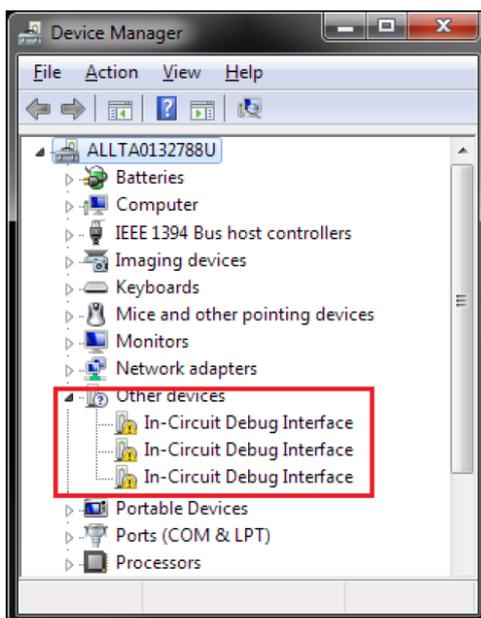
これらのドライバは、http://www.ti.com/tool/stellaris_icdi_drivers からダウンロードすることができます。ドライバをインストールするには、コンピュータ上の所定の場所にドライバをダウンロードして解凍し、次の手順を実行します。

1. Windows の [スタート] メニューを開き、[コンピュータ] を右クリックして、ドロップダウン・メニューから [プロパティ] を選択します。

2. 画面左側の [コントロールパネル] から、[デバイスマネージャ] をクリックします。[デバイスマネージャ] 画面が開きます。[デバイスマネージャ] 画面には、そのコンピュータにインストールされているハードウェア・デバイスの一覧が表示されており、ここから各デバイスのプロパティを設定することができます。

開発ボードまたは評価ボードをコンピュータに初めて接続した場合は、コンピュータがオンボード ICDI インターフェイスと Stellaris® マイクロコントローラを検出します。[デバイスマネージャ] 画面上、コンピュータにインストールされていないドライバには黄色い感嘆符が付いています。

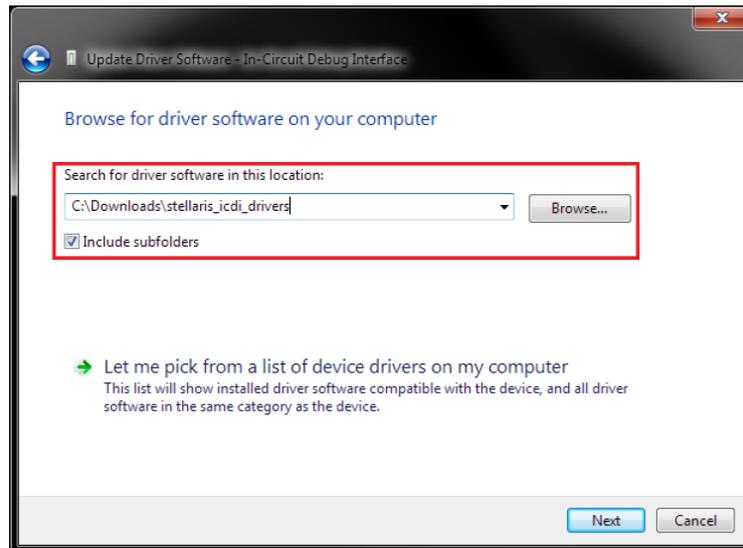
3. ドライバをインストールするには、[ほかのデバイス] 内の [In-Circuit Debug Interface] を右クリックして、[ドライバソフトウェアの更新] オプションを選択します。



4. [Update Driver Software - In-Circuit Debug Interface] 画面が表示されます。[コンピュータを参照してドライバ ソフトウェアを検索します] を選択して、コンピュータ上でドライバが格納されている場所を指定します。

Code Composer Studio™クイックスタート

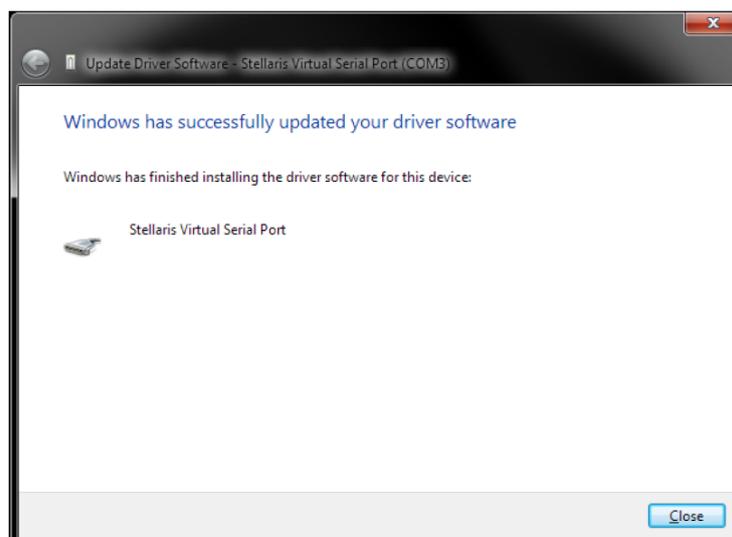
5. [参照] ボタンをクリックして、ドライバの格納場所を選択します。開発キットまたは評価キットに収録されている『Documentation and Software』 CD/DVD からドライバをインストールする場合、光ディスク・ドライブにディスクが入っているか、またはコンピュータのハード・ドライブ上の所定の場所に解凍後のドライバが格納されている必要があります。[Next] ボタンをクリックして、ドライバのインストールを開始します。



ドライバのインストール実行中は、[Update Driver Software In-Circuit Debug Interface] 画面が表示されます。

- 注:** [Windows セキュリティ] 画面が表示されたら、[このドライバ ソフトウェアをインストールします] を選択します。

6. ドライバが正常にインストールされると、以下の画面が表示されます。



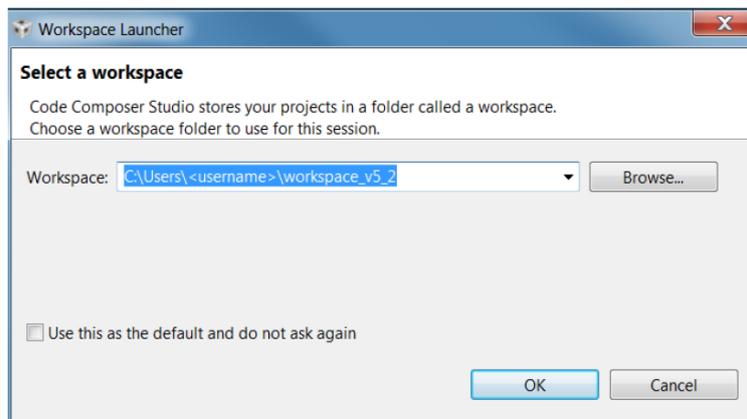
Code Composer Studio™クイックスタート

7. デバイス・マネージャを開き、ツールバー上の [ハードウェア変更のスキャン] ボタンをクリックします。この操作により、デバイス・マネージャのプロパティが更新されます。この時点で、[ポート (COM と LPT)] 内に [Stellaris Virtual Serial Port] が表示され、Stellaris Virtual Serial Port ドライバが正しくインストールされていることを示します。
8. 手順 3 から 7 を繰り返し、同様に Stellaris ICDI DFU Device ドライバと Stellaris ICDI JTAG/SWD Interface をインストールします。各ドライバが正常にインストールされると、[デバイス・マネージャ] 画面にそれぞれ表示されます。

これらのドライバにより、デバッガから JTAG/SWD インターフェイスへのアクセス、およびホスト PC から仮想 COM ポートへのアクセスが可能になります。また、これらのドライバをインストールした状態で Stellaris ベースの ICDI を持つ Stellaris ボードを新しくコンピュータに接続すると、Windows が自動的にそのボードを検出して必要なドライバをインストールします。

ステップ 4: Code Composer Studio の起動とワークスペースのオープン

1. Windows の [スタート] メニューから Code Composer Studio IDE を選択するか、またはデスクトップ上のアイコンをダブルクリックして、Code Composer Studio を起動します。Code Composer Studio IDE がロードされると、ワークスペース・フォルダの場所を指定する画面が開きます。



2. Workspace Launcher では、デフォルトで次のパスを指定しています。

C:\Users\<username>\workspace_v5_2

デフォルトのワークスペース・ロケーションをそのまま使用する場合は [OK] をクリックします。

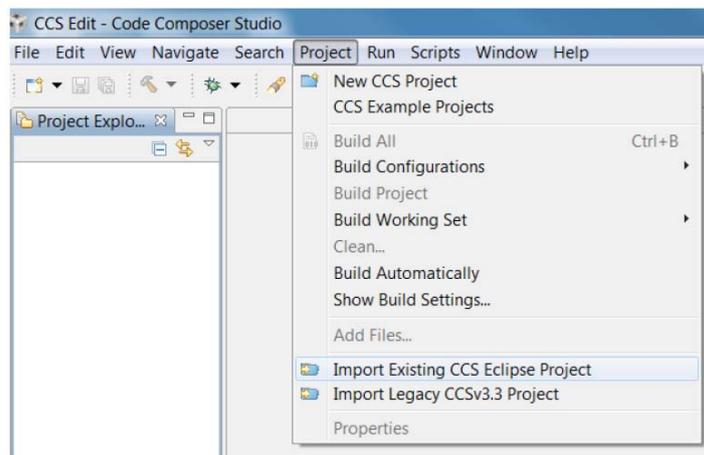
注: Code Composer Studio IDE を DVD からインストールしている場合、ライセンスが自動的にインストールされます。Code Composer Studio IDE をダウンロードしてインストールしている場合、ライセンス認証の手順が必要になります。

3. Code Composer Studio IDE が開き、ウェルカム画面が表示されます。タブ上の [X] ボタンをクリックしてこの画面を閉じます。これで空のワークスペースが現れます。

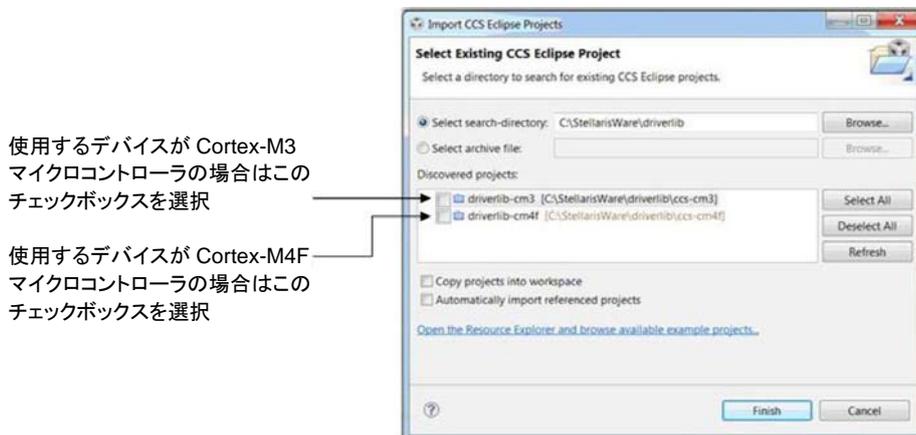
Code Composer Studio™クイックスタート

ステップ5: ライブラリのインポート

1. IDE の [Project] メニューから [Import Existing CCS Eclipse Project] を選択します。



2. [Import CCS Eclipse Projects] 画面で、Stellaris ペリフェラル・ドライバ・ライブラリのルート・ディレクトリ (StellarisWare\driverlib) を指定します。[Discovered project] フィールドには 2 つのプロジェクトが表示されています。使用するデバイスが Stellaris Cortex-M3 マイクロコントローラの場合は [driverlib-cm3] チェックボックスを、Stellaris Cortex-M4F マイクロコントローラの場合は [driverlib-cm4f] チェックボックスを選択します。選択が終わったら [Finish] をクリックします。



- 重要:** USB ライブラリやグラフィックス・ライブラリを使用しないボードで StellarisWare Firmware Development パッケージをインストールしている場合は、次の手順を省略します。以下の手順で示すディレクトリが StellarisWare インストール・ディレクトリに存在しない場合、[Cancel] をクリックして [Import CCS Eclipse Projects] ダイアログ・ボックスを閉じます。

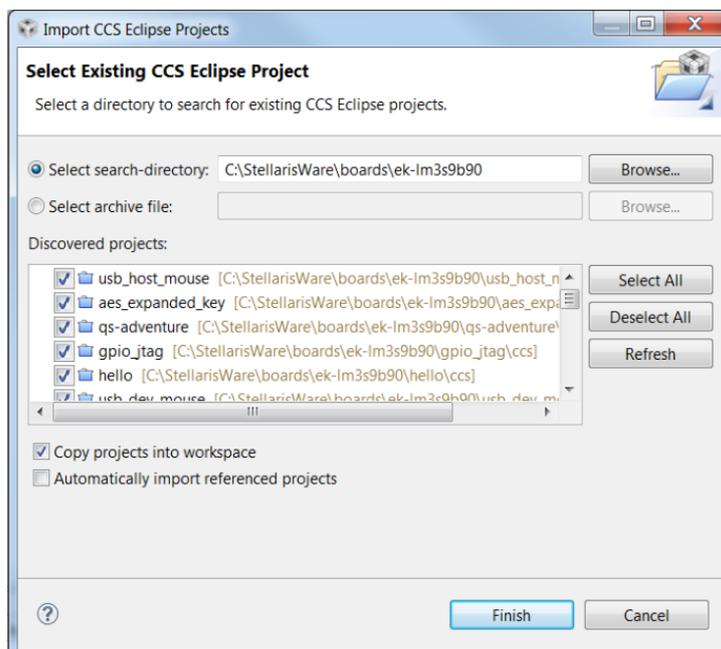
Code Composer Studio™クイックスタート

次の手順にしたがって手順 2 を繰り返し、USB ライブラリ (usblib) とグラフィックス・ライブラリ (grlib) をインポートします。

3. IDE の [Project] メニューから [Import Existing CCS Eclipse Project] を選択します。USB ライブラリのルート・ディレクトリ (StellarisWare\usblib) を指定します。使用するデバイスが Stellaris Cortex-M3 マイクロコントローラの場合は [usb-lib-cm3] チェックボックスを、Stellaris Cortex-M4F マイクロコントローラの場合は [driverusb-lib-cm4f] チェックボックスを選択します。選択が終わったら [Finish] をクリックします。
4. IDE の [Project] メニューから [Import Existing CCS Eclipse Project] を選択します。グラフィックス・ライブラリのルート・ディレクトリ (StellarisWare\grlib) を指定します。使用するデバイスが Stellaris Cortex-M3 マイクロコントローラの場合は [gr-lib-cm3] チェックボックスを、Stellaris Cortex-M4F マイクロコントローラの場合は [gr-lib-cm4f] チェックボックスを選択します。選択が終わったら [Finish] をクリックします。

ステップ 6: ボード・サンプルのインポート

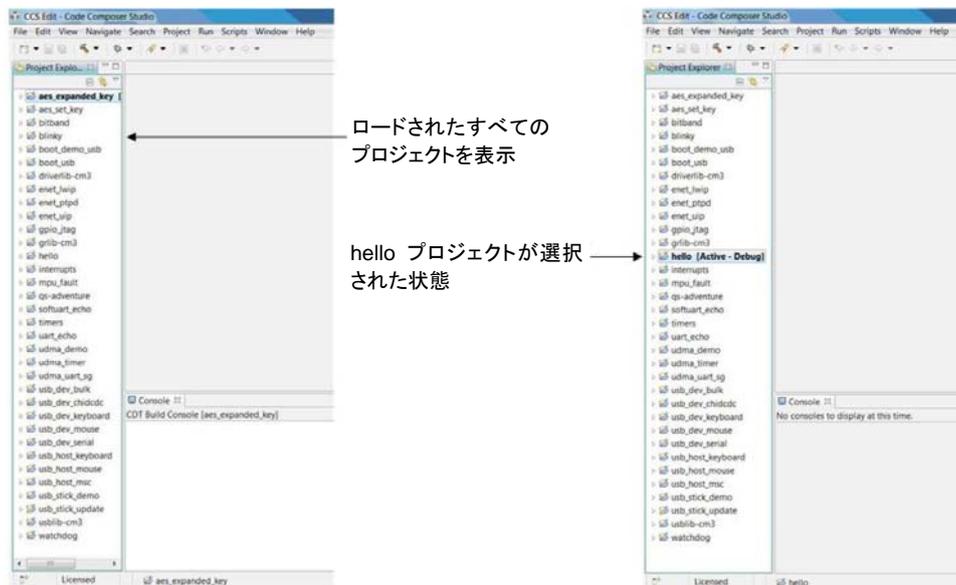
1. IDE の [Project] メニューから [Import Existing CCS Eclipse Project] を選択します。
2. [Import] ダイアログ・ボックスが表示されます。使用しているボードのルート・ディレクトリを選択します。次の図では、EK-LM3S9B90 ボード (StellarisWare\boards\ek-lm3s9b90) を例として示しています。[Discovered projects] フィールドに選択したボード・サンプルを確認し、[Finish] をクリックします。



注: 通常はプロジェクトをワークスペースにコピーする必要はありません。ただし、ボード・サンプルの場合は、プロジェクトをワークスペースにコピーしておくことをお勧めします。

Code Composer Studio™ クイックスタート

3. [Projects Explorer] にすべてのプロジェクトが表示されます。



ステップ7: プロジェクトのビルドとデバッグ

1. [Project Explorer] 画面で hello プロジェクトをクリックして、[Active] プロジェクトとして設定します (上図参照)。
2. [Project] メニューから [Build Project] を選択し、ビルドを実行します。この処理には数分かかります。プロジェクトのビルド中は、コンソール画面にメッセージがスクロール表示されます。ビルドが完了すると、コンソール画面に Build Finished と表示されます。
3. [Run] メニューから [Debug] を選択します。または、ツールバーの [Debug Launch] アイコンをクリックするかキーボード上の F11 ファンクション・キーを押します。
4. Code Composer Studio デバッガが開発ボードまたは評価ボードに自動的に接続し、フラッシュ・メモリをプログラミングして、main() 関数の最初からデバッグを開始します。この時点で、メモリの検査と修正、変数やプロセッサ・レジスタのプログラミング、ブレークポイントの設定、ステップ実行など、一般的なデバッグ操作を行うことができます。プログラムを実行するときは、[Run] メニューから [Resume] を選択します。

ステップ8: その他のサンプル・プログラムのビルドと実行

[Project Explorer] 画面には、他にもサンプル・プロジェクトが表示されています。他のサンプル・プログラムのビルドも、ステップ 7 に従って実行します。評価ボードには、そのほかのサンプル・プロジェクトとしてリストされている qs-xxxxxx がクイックスタート・アプリケーションとしてあらかじめロードされています。

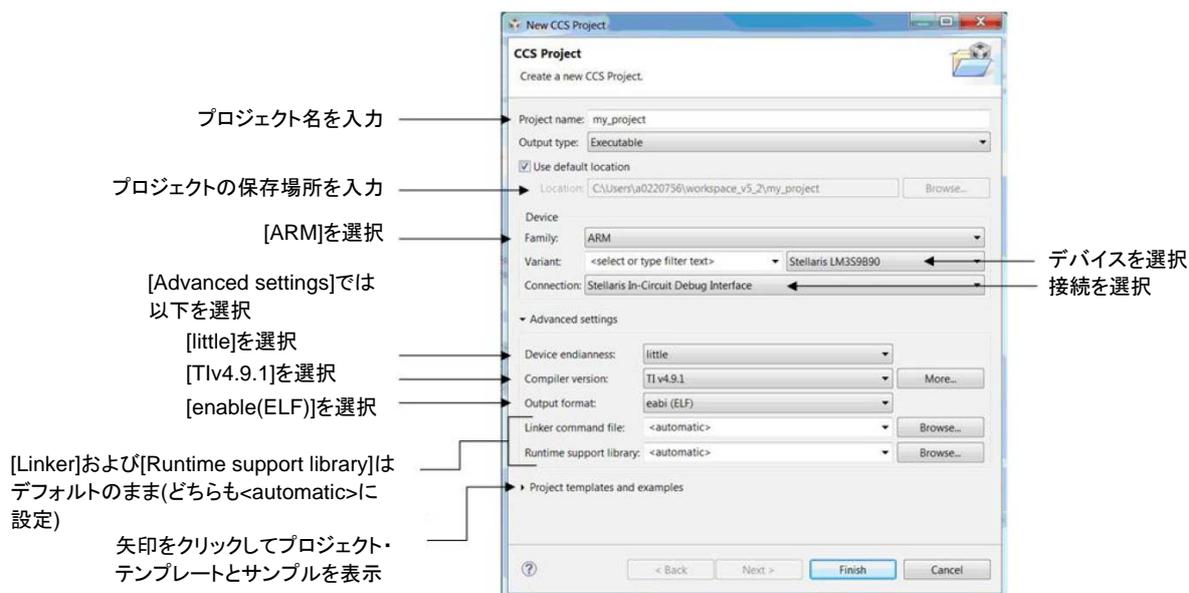
Code Composer Studio™ クイックスタート

新規プロジェクトの作成

StellarisWare サンプル・アプリケーションのビルドが終わったら、独自のプロジェクトを作成して開発を開始します。独自のプロジェクトを作成するには、既存のプロジェクトを利用するか、または新規に作成します。

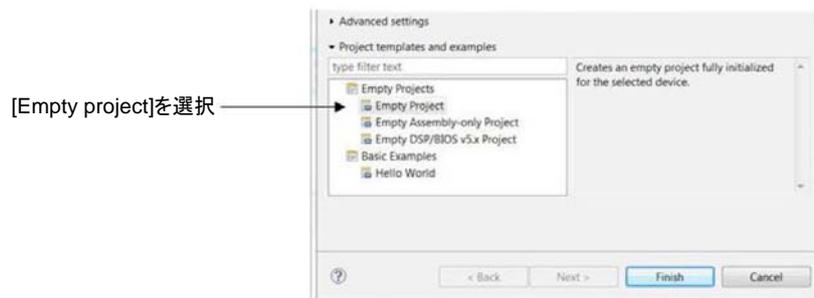
次の手順では、新規プロジェクトを作成し、既存のプロジェクトからコードをコピーして、作成した新規プロジェクトをビルドする例を示します。

1. 新規プロジェクトをワークスペースに追加するために、メイン・メニュー・バーから [File] > [New] > [CCS Project] を選択します。
2. [New CCS Project] 画面が表示されます。この画面上で新規 CSS プロジェクトを作成します。[Project name]、[Location]、[Family]、[Variant (device)]、[Connection]、[Device endianness]、[Compiler version]、[Output format] の各フィールドのパラメータを設定します（次の画面参照）。[Linker command file] と [Runtime support library] はデフォルト設定 (<automatic>) のままにしておきます。



Code Composer Studio™ クイックスタート

3. [Project templates and examples] の矢印をクリックして、表示を展開します。[Empty project] を選択して、[Finish] をクリックします。

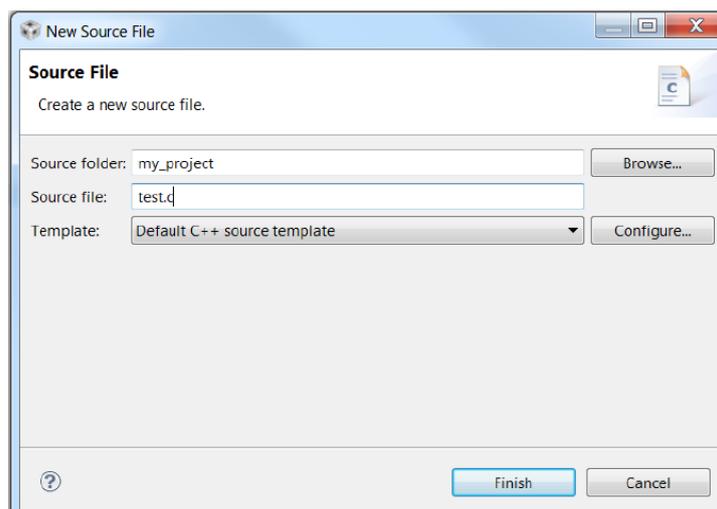


4. 新規プロジェクトが作成されます。このプロジェクトには main.c ソース・ファイルのみが含まれ、他のソース・コードは含まれていません。main.c ファイルに無限ループ関数を追加して、main.c ファイルを保存します。

```
in.c - Code Composer Studio
Search Project Run Scripts Window Help
main.c
1 /*
2  * main.c
3  */
4 void main(void)
5 {
6     //
7     // Loop forever.
8     //
9     while(1)
10    {
11    }
12
13 }
14
```

Code Composer Studio™ クイックスタート

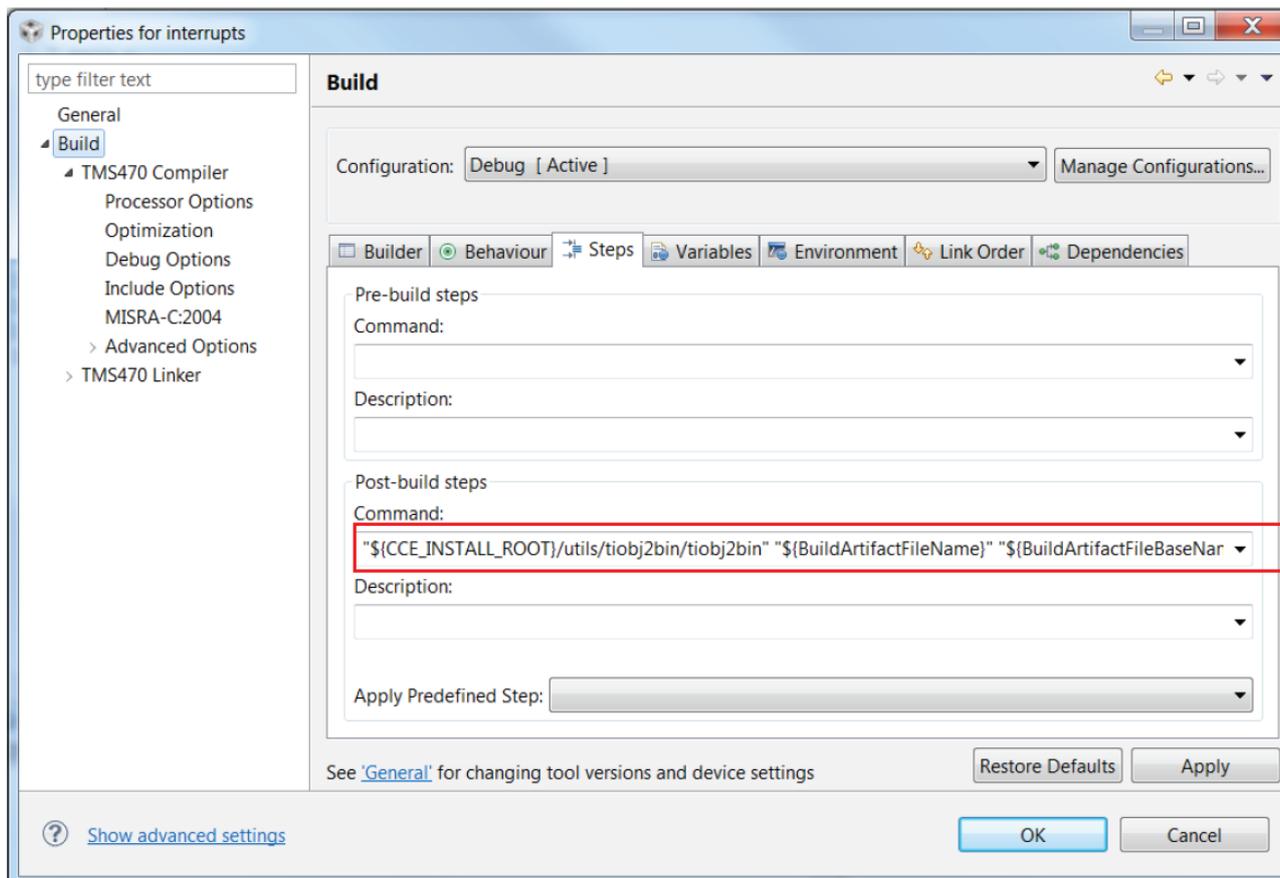
- プロジェクトにスタートアップ・コードを追加します。StellarisWare の既存のサンプルディレクトリ (¥StellarisWare¥boards¥<board>¥blinky など) から、手順 2 と 3 で作成した新しいプロジェクト・ディレクトリに startup_ccs.c ファイルをコピーします。
- この時点で、main.c ファイルと startup_ccs.c ファイルが自動的にプロジェクトに追加されています。ファイルが自動的に追加されていない場合は、プロジェクトを右クリックし、ドロップダウン・メニューから [Add Files...] を選択して、新規プロジェクト・フォルダに入れるファイルを探し選択します。
- メイン・メニューから [Project] > [Build Project] を選択して、新規プロジェクトをビルドします。
作成された新規プロジェクトに対し、既存の StellarisWare ボード・サンプルを参照しながら、独自のコードを追加します。
- 新しい C ソース・ファイルを作成するときは、メイン・メニューから [File] > [New] > [Source File] を選択します。
- [New Source File] ダイアログ・ボックスが表示されたら、ファイル名を入力して [Finish] をクリックします。



Code Composer Studio™クイックスタート

オプション: プロジェクトのバイナリ・ファイル出力設定

プロジェクトをバイナリ・ファイル (.bin) で出力するよう設定することもできます。バイナリ・ファイルで出力することにより、既存の StellarisWare サンプル・プロジェクトからポストビルド・ステップをコピーして、LM Flash Programmer などのツールで使用できるようになります。プロジェクトを右クリックして、[Show Build Settings...] を選択します。

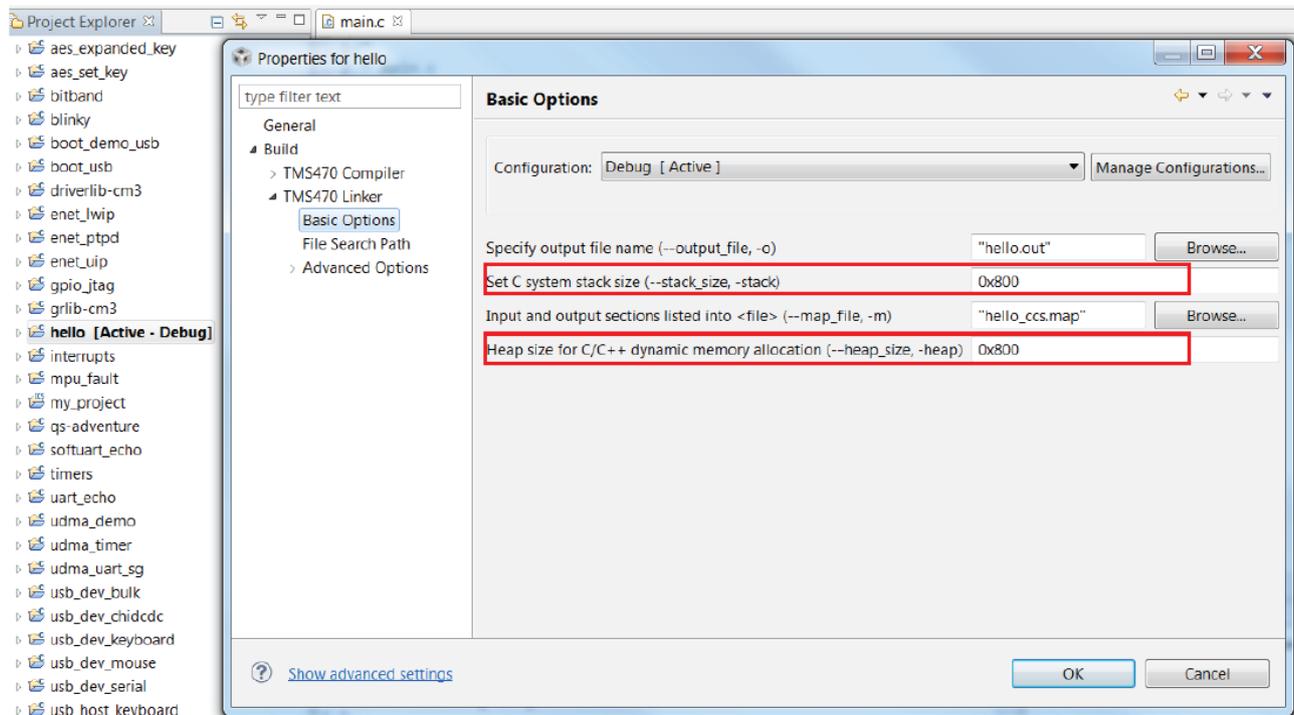


printf や fread などの標準 C I/O 関数を使用する場合は、次の手順でプロジェクトのスタック割り当てとヒープ割り当てを増やす必要があります。

1. プロジェクトを右クリックして [Show Build Settings...] を選択し、[Properties] ダイアログ・ボックスを開きます。
2. [Properties] ダイアログ・ボックスの左側のペインにある、[TMS470 Linker] の下の [Basic Options] を選択します。

Code Composer Studio™クイックスタート

3. システム・スタックとヒープに対して「0x800」と入力します。



最後に

以上の操作により、Code Composer Studio 開発ツールをインストールし、それを使用してサンプル・アプリケーションをビルドして、Stellaris 開発ボードまたは評価ボードにロードしました。この後、デバッガを使用したり、サンプル・プロジェクトを参照しながら独自のアプリケーションを作成したりすることもできます。Code Composer Studio の詳細については、CCS Developer Site をご参照ください。CCS Developer Site には、メイン・メニューの [Help] から [CCSv5 Developer Site] を選択するとアクセスできます。

参考ドキュメント

『Stellaris Evaluation Kit Documentation and Software』DVD には、以下の参考ドキュメントが収録されています。また、これらのドキュメントは www.ti.com/stellaris からダウンロードすることができます。

- Stellaris Evaluation Kit User's Manual 英語版
- StellarisWare Software 英語版(文書番号: SW-LM3S)
- StellarisWare Peripheral Driver Library User's Guide 英語版(文書番号: SW-DRL-UG)

以下の Web サイトもご参照ください。

- Code Composer Studio ウェブサイト (<http://www.ti.com/ccstudio>)

ご注意

日本テキサス・インスツルメンツ株式会社（以下TIJといいます）及びTexas Instruments Incorporated (TIJの親会社、以下TIJないしTexas Instruments Incorporatedを総称してTIといいます)は、その製品及びサービスを任意に修正し、改善、改良、その他の変更をし、もしくは製品の製造中止またはサービスの提供を中止する権利を留保します。従いまして、お客様は、発注される前に、関連する最新の情報を取得して頂き、その情報が現在有効かつ完全なものであるかどうかご確認下さい。全ての製品は、お客様とTIJとの間取引契約が締結されている場合は、当該契約条件に基づき、また当該取引契約が締結されていない場合は、ご注文の受諾の際に提示されるTIJの標準販売契約約款に従って販売されます。

TIは、そのハードウェア製品が、TIの標準保証条件に従い販売時の仕様に対応した性能を有していること、またはお客様とTIJとの間で合意された保証条件に従い合意された仕様に対応した性能を有していることを保証します。検査およびその他の品質管理技法は、TIが当該保証を支援するのに必要とみなす範囲で行なわれております。各デバイスの全てのパラメーターに関する固有の検査は、政府がそれ等の実行を義務づけている場合を除き、必ずしも行なわれておりません。

TIは、製品のアプリケーションに関する支援もしくはお客様の製品の設計について責任を負うことはありません。TI製部品を使用しているお客様の製品及びそのアプリケーションについての責任はお客様にあります。TI製部品を使用したお客様の製品及びアプリケーションについて想定される危険を最小のものとするため、適切な設計上および操作上の安全対策は、必ずお客様にてお取り下さい。

TIは、TIの製品もしくはサービスが使用されている組み合わせ、機械装置、もしくは方法に関連しているTIの特許権、著作権、回路配置利用権、その他のTIの知的財産権に基づいて何らかのライセンスを許諾するということは明示的にも黙示的にも保証も表明もしておりません。TIが第三者の製品もしくはサービスについて情報を提供することは、TIが当該製品もしくはサービスを使用することについてライセンスを与えたり、保証もしくは是認するということの意味しません。そのような情報を使用するには第三者の特許その他の知的財産権に基づき当該第三者からライセンスを得なければならない場合もあり、またTIの特許その他の知的財産権に基づきTIからライセンスを得て頂かなければならない場合もあります。

TIのデータブックもしくはデータシートの中にある情報を複製することは、その情報に一切の変更を加えること無く、かつその情報と結び付けられた全ての保証、条件、制限及び通知と共に複製がなされる限りにおいて許されるものとします。当該情報に変更を加えて複製することは不正で誤認を生じさせる行為です。TIは、そのような変更された情報や複製については何の義務も責任も負いません。

TIの製品もしくはサービスについてTIにより示された数値、特性、条件その他のパラメーターと異なる、あるいは、それを超えてなされた説明で当該TI製品もしくはサービスを再販売することは、当該TI製品もしくはサービスに対する全ての明示的保証、及び何らかの黙示的保証を無効にし、かつ不正で誤認を生じさせる行為です。TIは、そのような説明については何の義務も責任もありません。

TIは、TIの製品が、安全でないことが致命的となる用途ないしアプリケーション(例えば、生命維持装置のように、TI製品に不良があった場合に、その不良により相当な確率で死傷等の重篤な事故が発生するようなもの)に使用されることを認めておりません。但し、お客様とTIの双方の権限有る役員が書面でそのような使用について明確に合意した場合は除きます。たとえTIがアプリケーションに関連した情報やサポートを提供したとしても、お客様は、そのようなアプリケーションの安全面及び規制面から見た諸問題を解決するために必要とされる専門的知識及び技術を持ち、かつ、お客様の製品について、またTI製品をそのような安全でないことが致命的となる用途に使用することについて、お客様が全ての法的責任、規制を遵守する責任、及び安全に関する要求事項を満足させる責任を負っていることを認め、かつそのことに同意します。さらに、もし万一、TIの製品がそのような安全でないことが致命的となる用途に使用されたことによって損害が発生し、TIないしその代表者がその損害を賠償した場合は、お客様がTIないしその代表者にその全額の補償をするものとします。

TI製品は、軍事的用途もしくは宇宙航空アプリケーションないし軍事的環境、航空宇宙環境にて使用されるようには設計もされていませんし、使用されることを意図されておられません。但し、当該TI製品が、軍需対応グレード品、若しくは「強化プラスチック」製品としてTIが特別に指定した製品である場合は除きます。TIが軍需対応グレード品として指定した製品のみが軍需品の仕様書に合致いたします。お客様は、TIが軍需対応グレード品として指定していない製品を、軍事的用途もしくは軍事的環境下で使用することは、もっぱらお客様の危険負担においてなされるということ、及び、お客様がもっぱら責任をもって、そのような使用に関して必要とされる全ての法的要求事項及び規制上の要求事項を満足させなければならないことを認め、かつ同意します。

TI製品は、自動車用アプリケーションないし自動車の環境において使用されるようには設計されていませんし、また使用されることを意図されておられません。但し、TIがISO/TS 16949の要求事項を満たしていると特別に指定したTI製品は除きます。お客様は、お客様が当該TI指定品以外のTI製品を自動車用アプリケーションに使用しても、TIは当該要求事項を満たしていなかったことについて、いかなる責任も負わないことを認め、かつ同意します。

Copyright © 2012, Texas Instruments Incorporated
日本語版 日本テキサス・インスツルメンツ株式会社

弊社半導体製品の取り扱い・保管について

半導体製品は、取り扱い、保管・輸送環境、基板実装条件によっては、お客様での実装前後に破壊/劣化、または故障を起こすことがあります。

弊社半導体製品のお取り扱い、ご使用にあたっては下記の点を遵守して下さい。

1. 静電気

- 素手で半導体製品単体を触らないこと。どうしても触る必要がある場合は、リストストラップ等で人体からアースをとり、導電性手袋等をして取り扱うこと。
- 弊社出荷梱包単位（外装から取り出された内装及び個装）又は製品単品で取り扱いを行う場合は、接地された導電性のテーブル上で（導電性マットにアースをとったもの等）、アースをした作業者が行うこと。また、コンテナ等も、導電性のものを使うこと。
- マウンタやはんだ付け設備等、半導体の実装に関わる全ての装置類は、静電気の帯電を防止する措置を施すこと。
- 前記のリストストラップ・導電性手袋・テーブル表面及び実装装置類の接地等の静電気帯電防止措置は、常に管理されその機能が確認されていること。

2. 温・湿度環境

- 温度：0～40℃、相対湿度：40～85%で保管・輸送及び取り扱いを行うこと。（但し、結露しないこと。）

- 直射日光があたる状態で保管・輸送しないこと。
3. 防湿梱包
 - 防湿梱包品は、開封後は個別推奨保管環境及び期間に従い基板実装すること。
 4. 機械的衝撃
 - 梱包品（外装、内装、個装）及び製品単品を落下させたり、衝撃を与えないこと。
 5. 熱衝撃
 - はんだ付け時は、最低限260℃以上の高温状態に、10秒以上さらさないこと。（個別推奨条件がある時はそれに従うこと。）
 6. 汚染
 - はんだ付け性を損なう、又はアルミ配線腐食の原因となるような汚染物質（硫黄、塩素等ハロゲン）のある環境で保管・輸送しないこと。
 - はんだ付け後は十分にフラックスの洗浄を行うこと。（不純物含有率が一定以下に保証された無洗浄タイプのフラックスは除く。）

以上