

Technical Note

TecStar

Silicon Labs 社 Legacy 8bit MCU クイックスタートガイド

2015 年 9 月

株式会社 **マクニカ**
テクスター カンパニー

クイックスタートガイド

目次

1 はじめに	3
2 開発環境のご紹介	4
2-1 ハードウェア	4
2-1-1 C8051 Development Kit	4
2-1-2 C8051 ToolStick	4
2-2 ソフトウェア	5
2-2-1 Silicon Labs IDE	5
2-2-2 Configuration Wizard 2	6
2-2-3 Keil PK51	6
3 各種ドキュメント・サンプルコードの入手先	7
3-1 C8051 のドキュメント	7
3-2 C8051 のサンプルコード	7
4 ソフトウェア・インストール	8
4-1 Silicon Labs IDE、Configuration Wizard 2、Keil PK51 のインストール	8
4-2 KEIL コンパイラのライセンス設定	8
4-3 ツールチェーンの設定	8
5 ハードウェア・セットアップ	9
5-1 C8051 Development Kit のセットアップ	9
5-2 C8051 ToolStick のセットアップ	10
6 使用方法	11
6-1 デバッグ経路を指定する	11
6-2 サンプルコードを動かしてみる	12
6-3 デバッグ機能を使ってみる	13
6-4 ピン設定やペリフェラル設定をしてみる (Configuration Wizard 2)	16
改版履歴	18
参考文献	18

1 はじめに

この資料は、Silicon Laboratories (以下、Silicon Labs) 社製 MCU C8051 ファミリの開発環境について簡易にまとめたものです。多くの C8051 ファミリーは Simplicity Studio に開発環境を移行していますが、一部製品については旧環境をご使用頂く必要があり、この資料は旧環境についてまとめたものです。内容に誤りがないよう注意は払っておりますが、もし Silicon Labs 社が提供するドキュメント等と差異がございましたら、メーカー提供のものを優先してご参照ください。

また、Silicon Labs 社の ナレッジベース (FAQ) やコミュニティフォーラム (ユーザ同士で問題解決。Silicon Labs のエンジニアも頻りにコメントしています) には、本資料で取り上げていない様々な情報が記載されております。

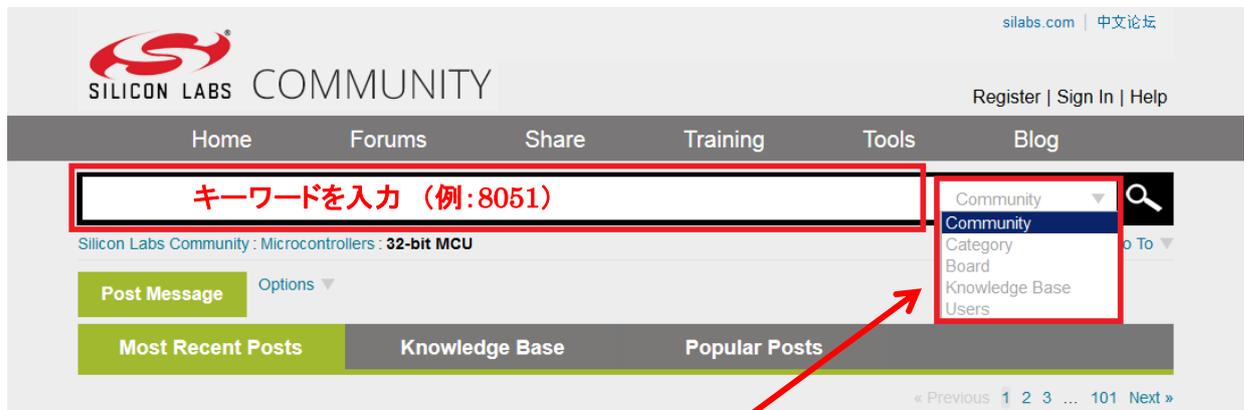
製品をご使用頂く過程で疑問や課題が生じることもあると思いますが、他のユーザが既に解決方法を見つけている場合も多々ございます。非常に有益ですので、ぜひご活用下さい。

◆ アクセス方法

Web Site から

<http://community.silabs.com/t5/Forum/ct-p/Forum>

◆ 使用方法



Community か Knowledge Base を選択

2 開発環境のご紹介

C8051 の開発環境について、ハードウェアとソフトウェアに分けてご紹介します。

2-1 ハードウェア

開発環境としては、Development Kit と ToolStick を用意しています。

2-1-1 C8051 Development Kit

ベーシックな評価基板が入った開発キットです。ターゲットボード、USB Debug Adaptor(PC とターゲットボードとをつなぐ機材)、AC/DC アダプタが同梱されています。MCU ファミリごとに Development Kit が用意されており、ターゲットボードに実装されている MCU が異なります。

ターゲットボードには LED やスイッチが実装されている他、アナログ入力やタッチボタンなど、各 MCU ファミリが持つ特徴的な機能を評価できる作りになっています。また、全ての I/O ピンが引き出されていますので、拡張性にも富んでいます。



Development Kit

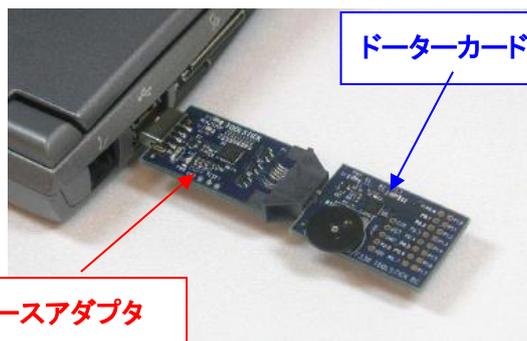


ターゲットボード

2-1-2 C8051 ToolStick

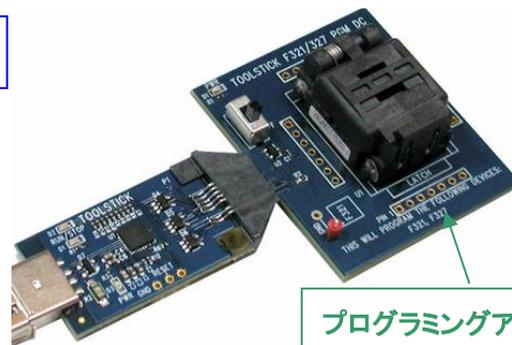
USB 給電で動作する、非常にコンパクトな評価基板です。PC に繋がるベースアダプタ、評価対象を搭載したドーターカード、の 2 つで構成されています。

ドーターカードは、MCU ファミリごと、パッケージごとに多種用意しており、評価したい MCU に最適なドーターカードをお選び頂くことができます。またソケットが載ったプログラミングアダプタも用意しており、プログラミングにご使用頂けます。



ベースアダプタ

ドーターカード



プログラミングアダプタ

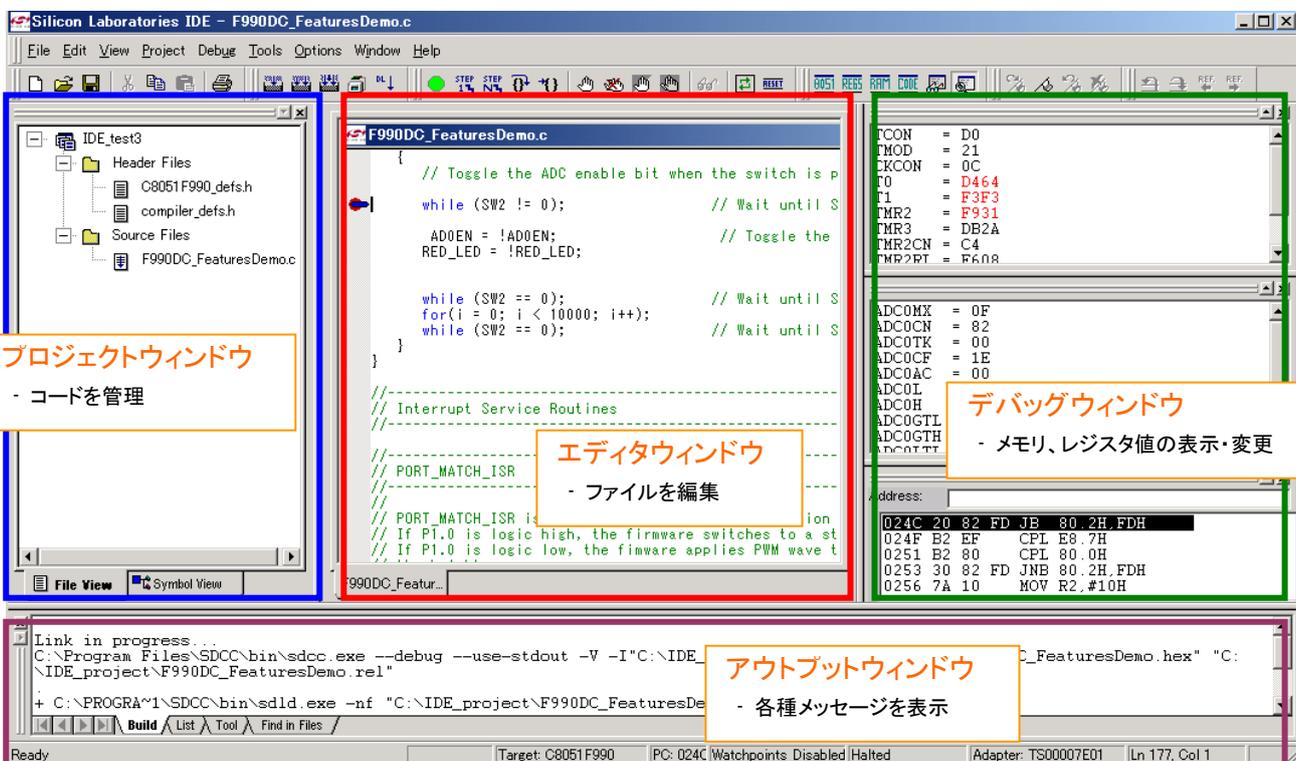
2-2 ソフトウェア

C8051 の開発環境である Silicon Labs IDE を使用して設計を行うこととなります。C・アセンブラのコンパイラについては、KEIL 社のコンパイラ(フルライセンス)を無償提供しています。

2-2-1 Silicon Labs IDE

Silicon Labs IDE は、ソースコードの編集、コンパイル、ダウンロード、デバッグといった一連の開発作業を、一つのソフトウェア上で行うことができる統合開発環境です。

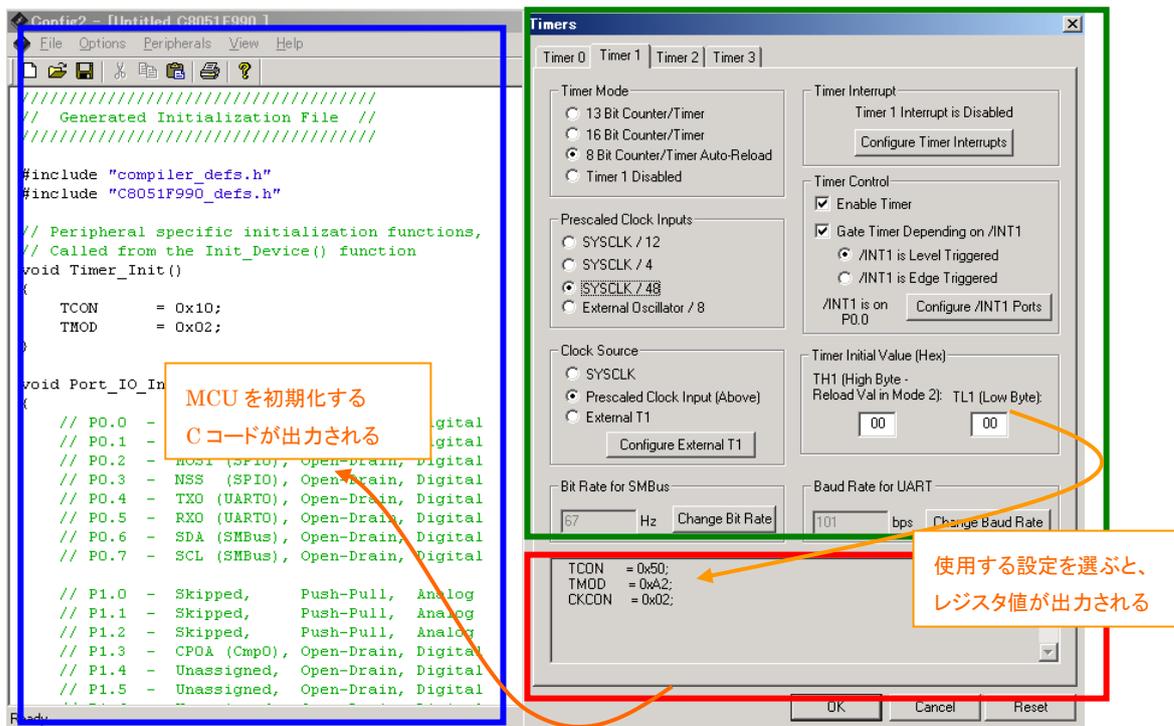
C8051 はオンチップ・デバッグ回路を搭載しているためエミュレータは不要で、オンボードでのデバッグが容易に行えます。ブレークポイント、ウォッチポイント、ステップ実行などのベーシックなデバッグ手法はもちろん、その他にも MCU のレジスタやメモリ領域を自由に書き換えることができますので、例外発生時の動作検証などを強力にアシストします。



2-2-2 Configuration Wizard 2

MCU のレジスタ設定を補助するソフトウェアです。直感的に使用できる GUI になっています。

クロック、割り込みなどの基本設定や、GPIO、ADC、タイマーなどのペリフェラル設定を GUI で行うと、それに応じたレジスタ値が算出され、初期化コード(C、アセンブラ)が出力されます。



2-2-3 Keil PK51

KEIL 社のコンパイラです。ご登録頂くことで、コードサイズ制限なくご使用頂けます。

3 各種ドキュメント・サンプルコードの入手先

C8051 の最新ドキュメント・サンプルコードの入手方法について紹介します。

3-1 C8051 のドキュメント

C8051 のドキュメントは、Silicon Labs 社の Web Site からご入手可能です。

<http://www.silabs.com/support/pages/document-library.aspx>

製品型番を入力すると、関連ドキュメントがリストアップされます。

Document Library

The document library has all of Silicon Labs' technical documents conveniently located in one place. To find the documents you need, start by selecting one of the product categories below.

Product Line**Family****Part Number**

3-2 C8051 のサンプルコード

C8051 のサンプルコードは、Silicon Labs IDE をインストールすると、下記フォルダにコピーされます。

C:\SiLabs\MCU\Examples

4 ソフトウェア・インストール

C8051 の評価に必要なソフトウェアをインストールします。

4-1 Silicon Labs IDE、Configuration Wizard 2、Keil PK51 のインストール

Silicon Labs IDE、Configuration Wizard 2、Keil PK51 for 8-bit MCU は、下記アドレスからダウンロードして頂けます。ダウンロード後、インストールを行います。

<http://www.silabs.com/products/mcu/Pages/8-bit-microcontroller-software.aspx>

4-2 KEIL コンパイラのライセンス設定

下記アドレスの Register Now!からライセンス申請を行ってください。

<http://www.silabs.com/products/mcu/Pages/8-bit-microcontroller-software.aspx>

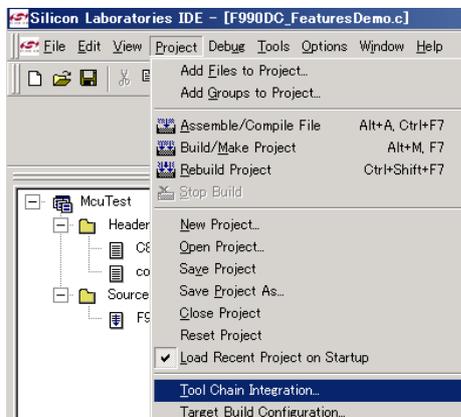
詳細な手順は AN104 をご参照ください。AN104 の中で、Keil uVision4 というツールが (Keil の IDE) もインストールされるのですが、このツールは Keil PK51 for 8-bit MCU をインストールした際に同時にインストールされます。

<http://www.silabs.com/Support%20Documents/TechnicalDocs/an104.pdf>

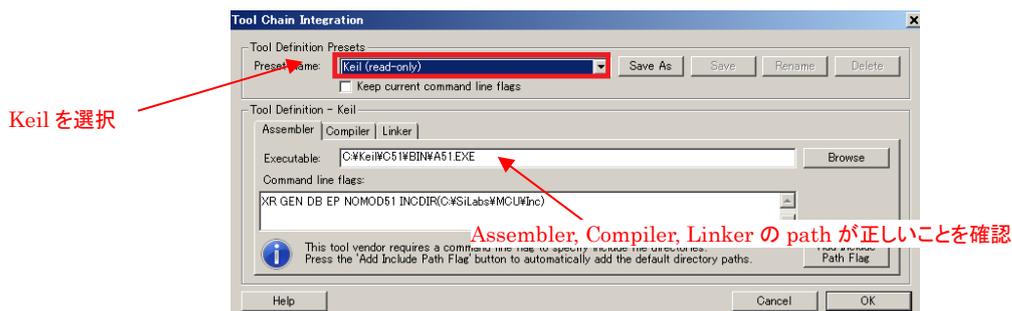
4-3 ツールチェーンの設定

使用するコンパイラの指定を行います。

“Project”メニューから“Tool Chain Integration”を選択します。



Tool Chain Integration ウィンドウでは、使用するコンパイラの設定を行うことができます。ここでは “Keil (read-only)” を選択して下さい。また、ここで指定されている Keil コンパイラへの path が、インストールしたフォルダと一致していることを確認して、“OK” ボタンをクリックして下さい。



5 ハードウェア・セットアップ

C8051 の評価に必要なハードウェアの設定を行います。

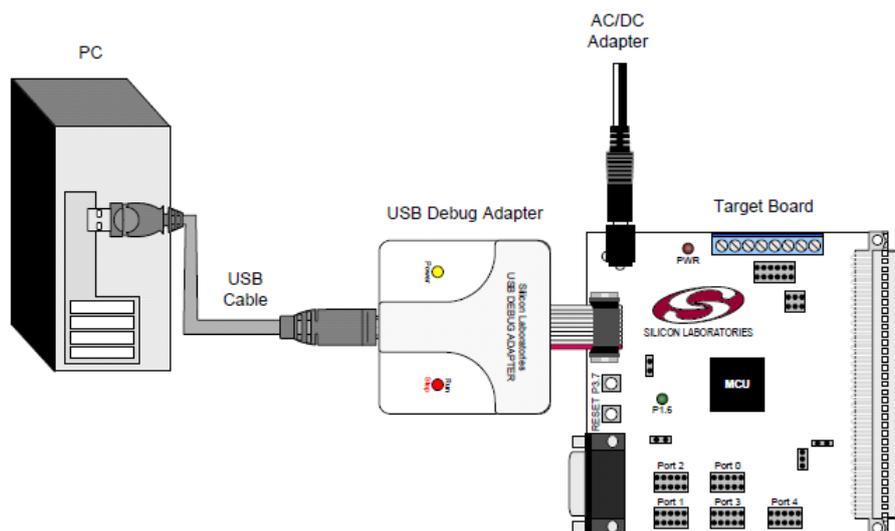
5-1 C8051 Development Kit のセットアップ

以下の手順で設定していきます。

1. USB デバッグアダプタを、ターゲットボードの DEBUG ポートに接続します。
2. USB ケーブルを、USB デバッグアダプタに接続します。
3. ターゲットボードにショートブロックがある場合には、ユーザガイドの指示に従って、正しく結線されているか確認してください。

例) F912DK, F930DK, F996DK の場合: J17 をショート、SW5 を ON に設定。

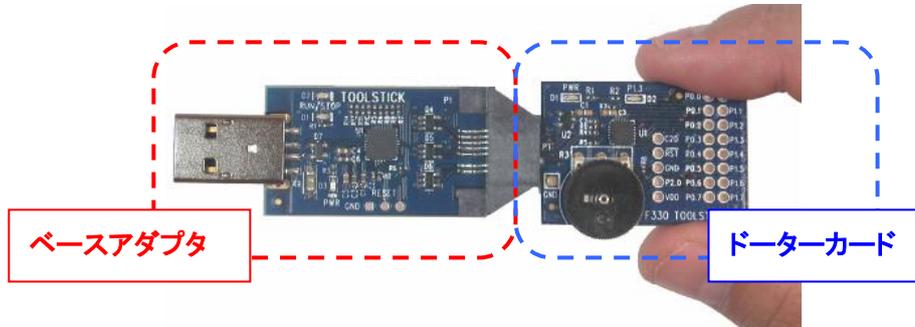
4. USB ケーブルのもう一方を PC に接続します。
5. AD/DC アダプタをターゲットボード、コンセントに接続します。



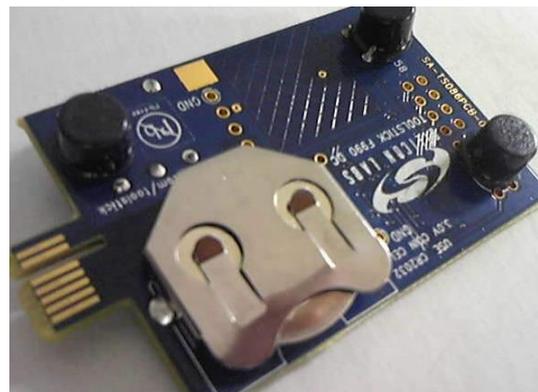
5-2 C8051 ToolStick のセットアップ

以下の手順で設定していきます。

1. ベースアダプタとドーターカードとを、下の写真のように接続します。



2. F912DC や F990DC など、いくつかのドーターカードでは電池でも動作するようになっています。USB 給電で動作するか、電池で動作するかを切り変えるスイッチがありますので、USB 給電 (TS PWR) を選択して下さい。



給電用スイッチ

3. USB コネクタを PC に接続します

6 使用方法

ここでは、コンパイルからデバッグまでの一連の手順についてご紹介します。

6-1 デバッグ経路を指定する

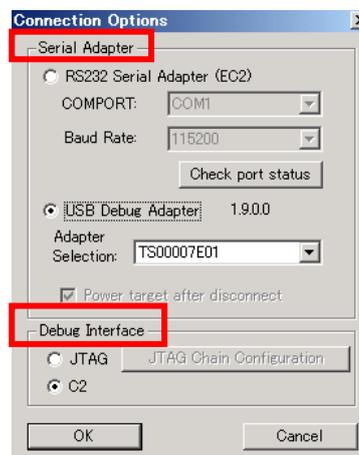
C8051 は製品によって C2 もしくは JTAG を介してダウンロードを行いますので、使用するデバッグ経路を指定する必要があります。

“Options”メニューから“Connection Options”を選択します。

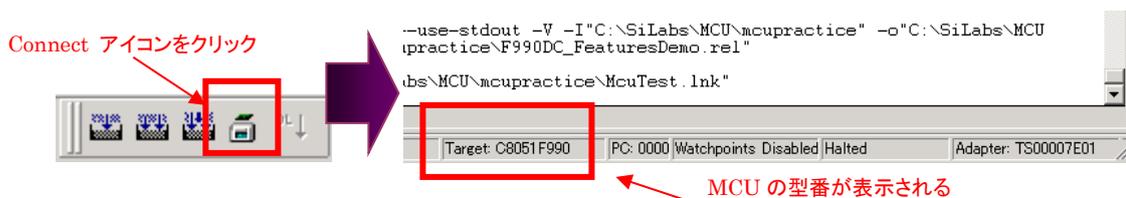


“Serial Adapter”では、“USB Debug Adaptor”を選択して下さい。“USB Debug Adaptor”が選択できない場合は、USB Debug Adaptor もしくは ToolStick が正しく USB ポートに挿入されていないか、何かしらの原因で認識されていません。他の USB ポートに接続する、給電機能付き USB Hub を介して接続する、PC を再起動する、などをお試し下さい。

“Debug Interface”では、使用する MCU がサポートしているデバッグインタフェースを選択します。F0xx, F1xx, F2xx ファミリは JTAG を、その他のファミリは C2 を選択して下さい。設定が終わったら“OK”ボタンをクリックします。



“Connect”アイコンをクリックし、MCUと接続します。正しく認識されると、IDEの下部に接続したMCUの型番が表示されます。

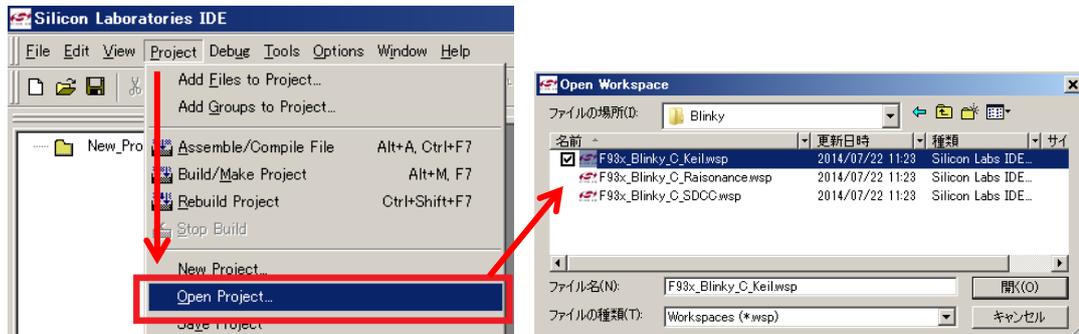


6-2 サンプルコードを動かしてみる

評価基板上の LED を点滅させるサンプルコードを、ダウンロードして動作を見てみます。

Project → Open Project で、サンプルコードフォルダにある wsp ファイルを選択します。wsp ファイルは Silicon Labs IDE のプロジェクトファイルです。下図は、F93x_92x 用のサンプルコードを使用してみます。

C:\SiLabs\MCU\Examples\C8051F93x_92x\Blinky\F93x_Blinky_C_Keil.wsp



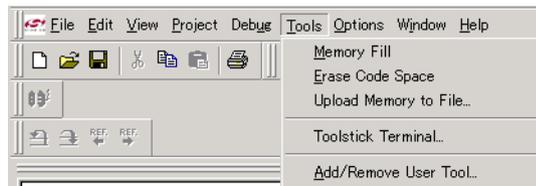
“Rebuild All”アイコンをクリックし、コンパイル、ビルドを実行します。エラー等の情報はアウトプットウインドウに表示されます。



Rebuild All アイコンをクリック

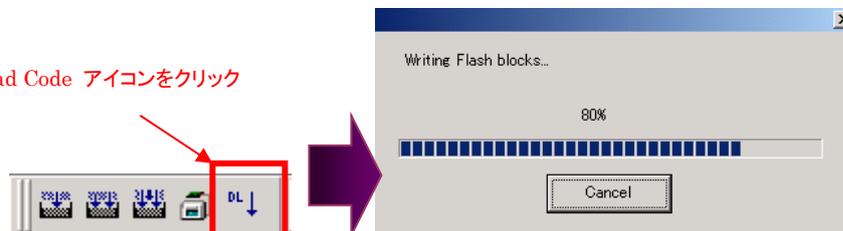
評価基板を接続し、「6-1 デバッグ経路を指定する」を参考に評価基板に Connect します。

“Tools”メニューから“Erase Code Space”を選択し、Flash Rom 内のデータを全消去します。



“Download Code”アイコンをクリックします。ダウンロードが実行されます。

Download Code アイコンをクリック



“Go”アイコンをクリックすると、プログラムが実行されます。“STOP”アイコンで停止します。LED が点滅することを確認できると思います。

Go アイコンをクリック



6-3 デバッグ機能を使ってみる

ソフトウェア・デバッグの際に使用する、ブレークポイント、ステップ実行などの標準的な機能について紹介します。

◆ ブレークポイント

プログラムを任意の場所で停止させるための機能です。

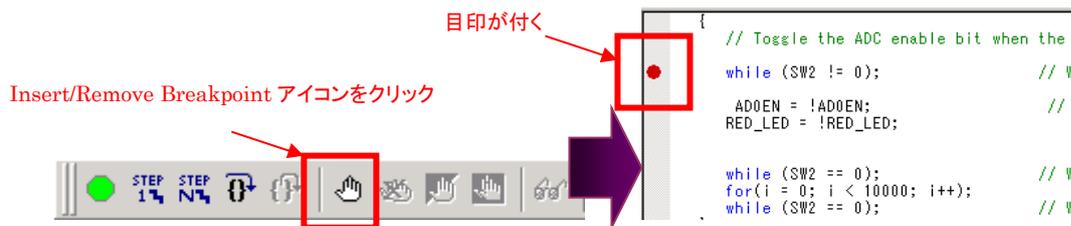
プログラムを停止させたい箇所にカーソルを移動させます。

```

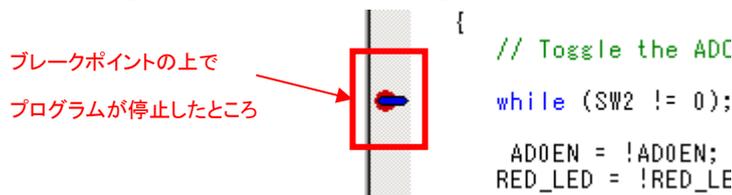
// Toggle the ADC enable bit when the switch is pressed
while (SW2 != 0);           // Wait until SW is pressed
    ADOEN = !ADOEN;         // Toggle the bit
    RED_LED = !RED_LED;

while (SW2 == 0);           // Wait until SW is released
for(i = 0; i < 10000; i++);
while (SW2 == 0);           // Wait until SW is released
    }
    
```

“Insert/Remove Breakpoint”アイコンをクリックすると、目印の●印が表示されます。



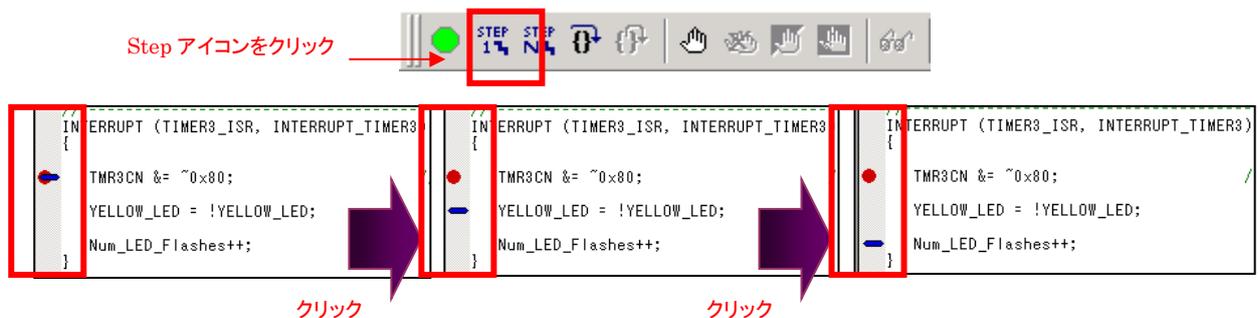
“Go”アイコンをクリックするとプログラムを実行され、ブレークポイントまで来たら停止します。



◆ ステップ実行

プログラムを1行ずつ実行させるための機能です。

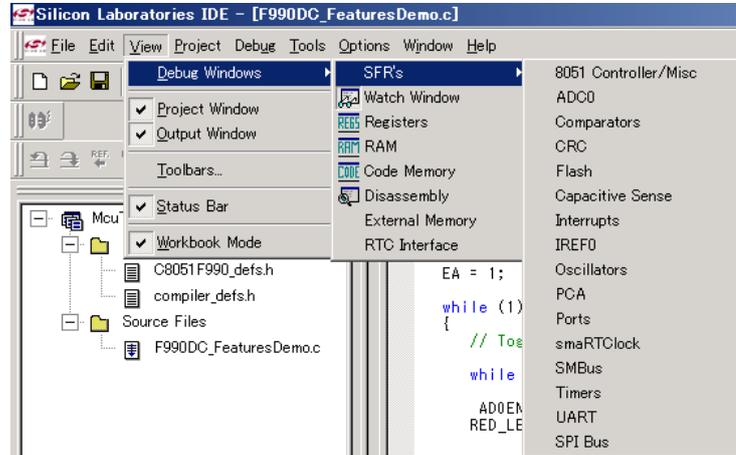
“Step”アイコンをクリックすると、クリックするたびに1行ずつ実行されます。



◆ デバッグウィンドウ

レジスタやメモリの値をモニタすることや、それらの値を変更することができます。

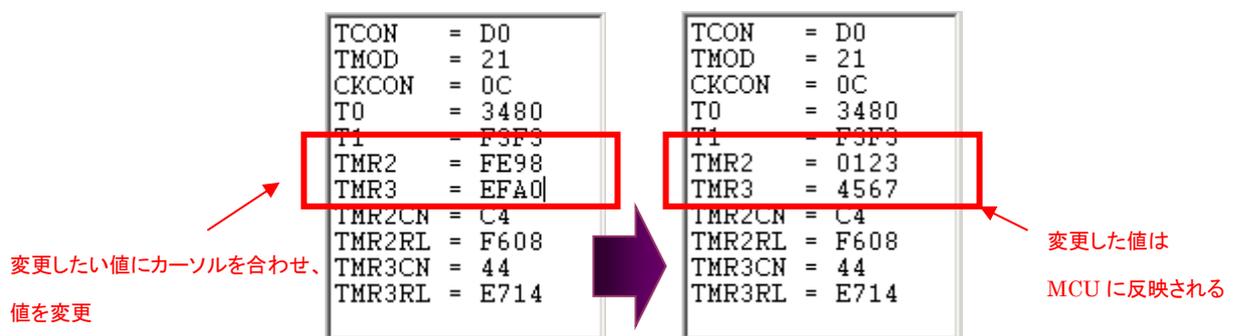
“View”メニューの“Debug Windows”を選択し、モニタする対象を決定します。



項目	概要
SFR's	各種ペリフェラルの関連レジスタ
Watch Window	変数
Registers	R0-R7 レジスタ
RAM	RAM 領域
Code Memory	ROM 領域
Disassembly	逆アセンブリ

“SFR's”から“Timers”を選択します。Timer に関連するレジスタの一覧が表示されます。

変更したいレジスタの値にカーソルを合わせ、キーボードで値を入力します。変更した値は即座に MCU に反映されます。



◆ ウォッチポイント

コードで定義した変数をモニタする機能です。

コード上で、モニタしたい変数にカーソルを合わせ、ダブルクリックします。

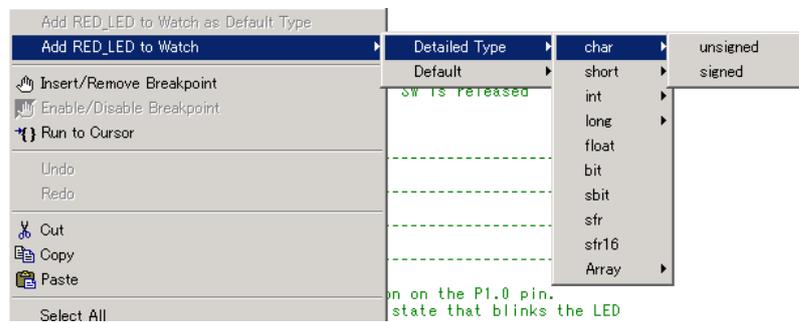
```

while (SW2 != 0);

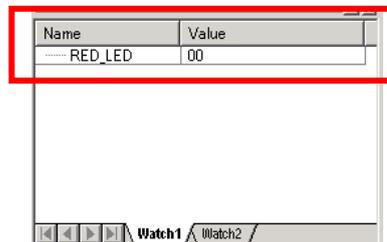
    ADOEN = !ADOEN;
    RED_LED = !RED_LED;

while (SW2 == 0);
for(i = 0; i < 10000; i++);
    
```

そのまま右クリックし、“Add xxx to Watch”を選択します。続いて、モニタしたい変数のデータ型を選択します。



ウォッチポイントウィンドウに変数が追加され、値をモニタできるようになります。値の変更も行えます。

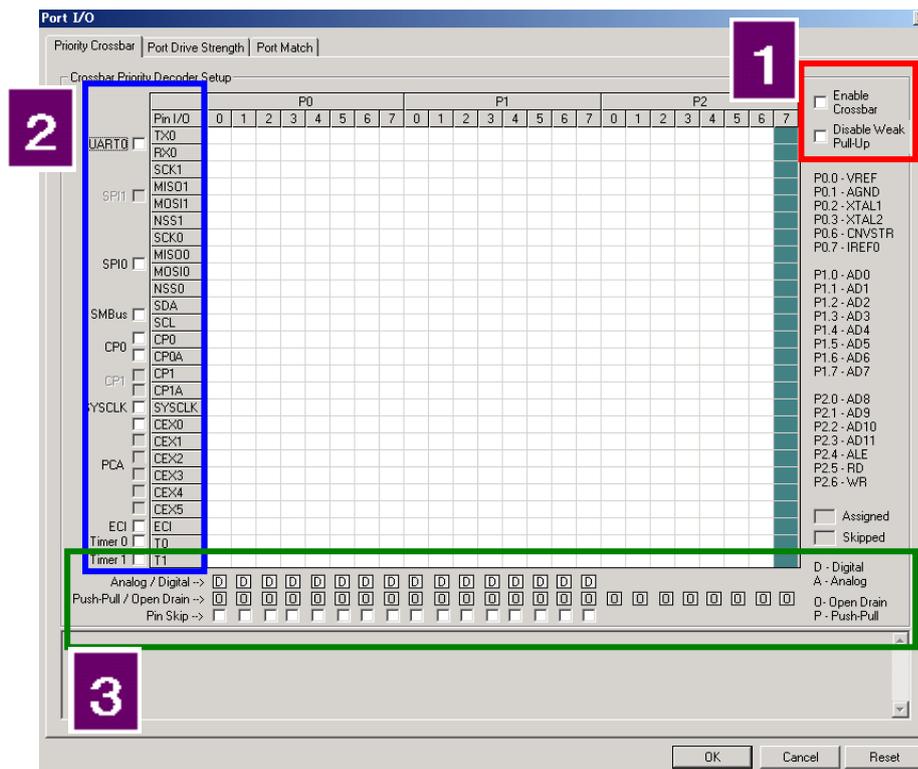


6-4 ピン設定やペリフェラル設定をしてみる (Configuration Wizard 2)

Configuration Wizard 2 を起動します。

◆ ピン設定

Peripherals メニューから Port I/O を選択すると、下図のようなウィンドウが表示されます。使用する MCU の機能やピン数によって、画面が異なります。



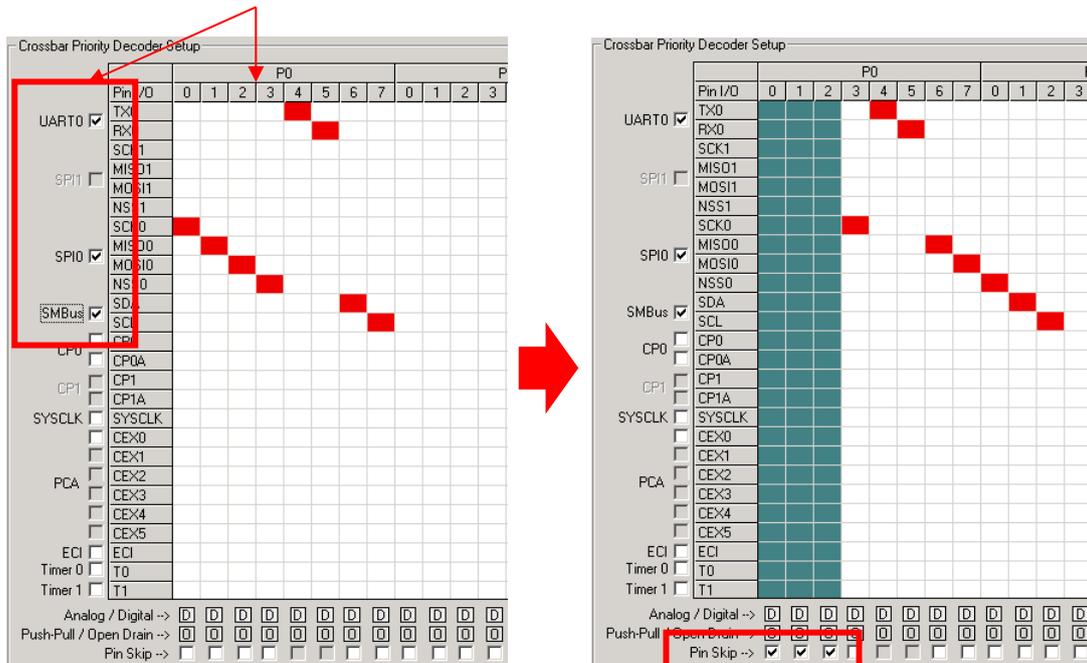
番号	概要
1	Crossbar の有効・無効 (必ず有効にしてください) 内蔵 pull-up の有効・無効
2	使用するペリフェラルの選択
3	アナログピン、デジタルピンの切り替え 出力モード(push-pull, open-drain)の切り替え ピンスキップの設定

I/O を使用するペリフェラル(UART, SPI など)を使用する場合、ポートはそれらペリフェラルに優先的に割り振られます。特定のポートを GPIO として使用したい場合には、ピンスキップの設定を行うことで優先的に確保できます。

下図(左)は、UART, SPI0, SMBus を使用する場合のポート設定です。P0.0~P0.7 が、これらペリフェラルに割り当てられています。

それに対し、下図(右)は、P0.0~P0.2 をピンスキップ設定した場合です。ペリフェラルのポート割り当てが P0.3~P1.2 へ変更されたことが判ります。

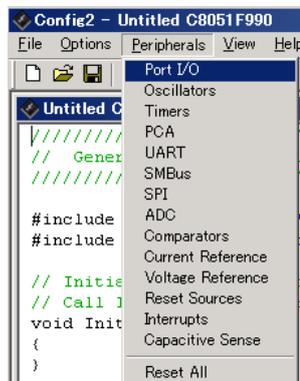
UART, SPI0, SMBus を選択。 P0.0~P0.7 が優先的に占有された。



ピンスキップ設定

◆ ペリフェラル設定

各ペリフェラルの設定は、Peripherals メニューを使用して行っていきます。一つ設定が終わるたびに、Cコードのヘッダーファイルが更新されます。このヘッダーファイルには、設定に準じたペリフェラル初期化ルーチンが記述されていますので、ソフトウェア設計の際にご使用頂けます。



改版履歴

Version	改定日	改定内容
1.0	2011年07月	・新規作成
1.3	2015年09月	・マクニカオンラインで公開

参考文献

- Silicon Labs 社 各種ドキュメント
- Silicon Labs 社 ナレッジベース、コミュニティフォーラム

免責、及び、ご利用上の注意

弊社より資料を入手されましたお客様におかれましては、下記の使用上の注意を一読いただいた上でご使用ください。

1. 本資料は非売品です。許可無く転売することや無断複製することを禁じます。
2. 本資料は予告なく変更することがあります。
3. 本資料の作成には万全を期していますが、万一ご不審な点や誤り、記載漏れなどお気づきの点がありましたら、弊社までご一報いただければ幸いです。
4. 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については、責任を負いかねますのであらかじめご了承ください。
5. 本資料は製品を利用する際の補助的なものとしてかかれたものです。製品をご使用になる場合は、メーカーリリースの資料もあわせてご利用ください。

本社

〒222-8561 横浜市港北区新横浜 1-6-3 TEL 045-470-9841 FAX 045-470-9844