

Eclipse & Blackfin Toolchain(gcc)

インストール&チュートリアル

第 1 版

ご注意：このドキュメントは旧版となりました。

下記 URL から最新版を参照ください。

<http://kaneko-sys.co.jp/support/>

金子システム株式会社

ご注意

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。
当社製品のご購入およびご使用にあたりましては、当社ホームページを通じて公開される情報を参照ください。
2. 当社から提供する情報の正確性と信頼性には万全を尽くしていますが、誤りがないことを保証するものではありません。当社はその使用に対する責任を一切負いません。その使用によって第三者の特許権、著作権その他知的財産が侵害された場合でも、同様に責任を負いません。
3. 本資料は、当社の書面による事前の明示同意がない限り、いかなる形式でも複製できません。

目次

1	はじめに.....	4
2	必要なハードウェア	4
3	ソフトウェアインストール	5
3.1	必要なソフトウェアパッケージ	5
3.2	Blackfin Toolchain のインストール	6
3.3	gnICE+ドライバのインストール.....	7
3.4	Blackfin Toolchain の設定ファイルの修正.....	7
3.5	Eclipse のインストール	10
3.6	Eclipse の起動	10
3.7	Blackfin Plugins のインストール.....	12
4	Blackfin アプリケーションの作成	16
4.1	プロジェクトの作成.....	17
4.2	gdb によるデバッグ	22
5	Silicon Revision について.....	26
6	更新履歴.....	27

1 はじめに

本書は、Eclipse と GNU Toolchain for the Blackfin Processor(以下 Blackfin Toolchain)を使ったアプリケーションの開発チュートリアルです。Windows を対象に、Blackfin Toolchain と Eclipse のインストール及び、簡単なサンプルプログラムの構築、デバッグまでの手順を解説します。

2 必要なハードウェア

ここでは、以下のハードウェアを使用します。

表 1 ハードウェア一覧

製品名	写真
金子システム ACB-BF592 以下よりお求めいただけます。 http://kaneko-sys.co.jp/shop/	 The image shows a blue printed circuit board (PCB) for the ACB-BF592-A. It features a central Blackfin processor (U1), a USB-to-JTAG interface (U3), and various passive components like resistors (R1, R16) and capacitors (C1, C2, C5). The board has a 40-pin header on the left and a 20-pin header on the right. The text 'KANEKO SYSTEM' is printed on the bottom right.
gnICE+ Digikey にて購入可能です Digikey 品番 : 190-2255-ND	 The image shows a small black integrated circuit (IC) package for the gnICE+ Hi-Speed USB JTAG ICE V1.1. It has a green PCB and a black plastic body. The text 'BLACKfin gnICE+' and 'Hi-Speed USB JTAG ICE V1.1' are visible on the package.

3 ソフトウェアインストール

3.1 必要なソフトウェアパッケージ

必要なパッケージは、以下の通りです。あらかじめ下記のファイルをダウンロードしてください。

- Blackfin Toolchain [blackfin-toolchain-win32-201R1.exe](http://blackfin.uclinux.org/gf/download/frsrelease/531/9541/blackfin-toolchain-win32-201R1.exe)
<http://blackfin.uclinux.org/gf/download/frsrelease/531/9541/blackfin-toolchain-win32-201R1.exe>
- Eclipse Helios Sr2 Packages [Eclipse IDE for C/C++ Developers](http://www.eclipse.org/downloads/packages/release/helios/sr2)
<http://www.eclipse.org/downloads/packages/release/helios/sr2>



Eclipse IDE for C/C++ Developers, (87 MB)
Downloaded 521,798 Times [Details](#)

Windows **32-bit 64-bit**
Mac Cocoa **32-bit 64-bit**
Linux **32-bit 64-bit**

※Windows 32bit版と 64bit版があります。ご使用の環境によってダウンロードするファイルを選択してください。

- Java Runtime
<http://java.com/ja/>

EclipseはJavaで作成されているため、動作にはJava Runtimeが必要です。ライセンスに同意のうえ、手順に沿ってインストールしてください。なお、javaのインストール手順はここでは解説しませんので、詳しくはjavaのサイトを参照してください。

3.2 Blackfin Toolchain のインストール

blackfin-toolchain-win32-2011R1.exe をダブルクリックし、インストールしてください。

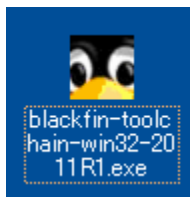


図 1 Blackfin Toolchain インストーラアイコン

インストールの条件を変更する必要はありません。

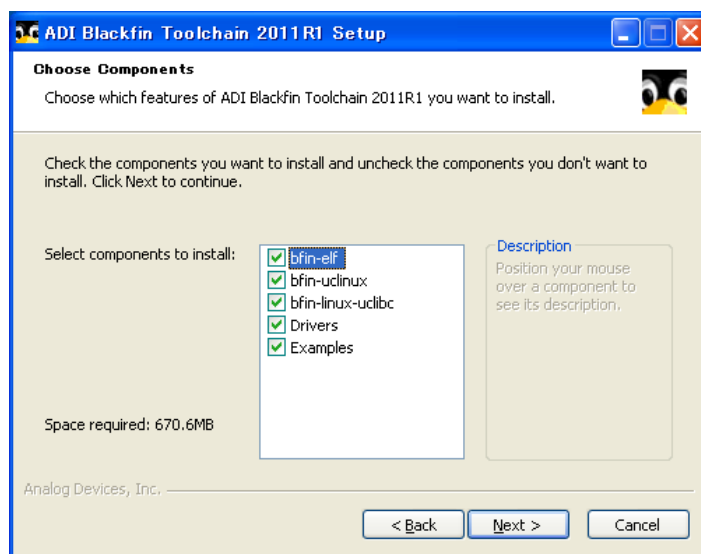


図 2 Blackfin Toolchain インストール画面（コンポーネント選択）

インストールが無事終われば、完了です。

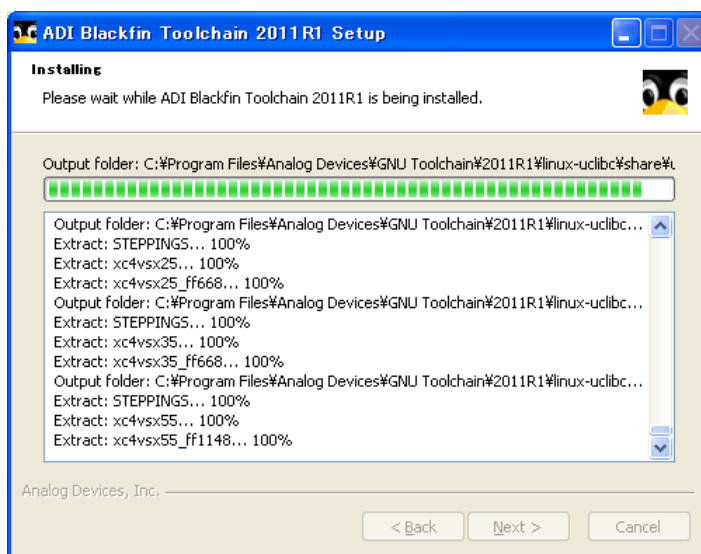


図 3 Blackfin Toolchain インストール画面（インストール中）

3.3 gnICE+ドライバのインストール

gnICE+を USB ケーブルで接続し、必要なドライバをインストールしてください。ドライバファイルは、標準構成で以下のフォルダに格納されています。

【32bit 版 Windows】

C:\Program Files\Analog Devices\GNU Toolchain\2011R1\gnICE-drivers

【64bit 版 Windows】

C:\Program Files (x86)\Analog Devices\GNU Toolchain\2011R1\gnICE-drivers

3.4 Blackfin Toolchain の設定ファイルの修正

ADSP-BF592 を gnICE+を使ってデバッグする場合、Blackfin Toolchain の設定ファイルのミスにより、JTAG にてプロセッサを正しく認識しないことが報告されています。gnICE+を ACB-BF592 に接続し、以下の作業で正しく認識することを確認してください。エラーが発生する場合、設定ファイルを書き換えれば認識します。この不具合は Blackfin Toolchain バージョン 2011R1 で確認しており、今後のバージョンでは修正されている可能性があります。

(※この作業は ADSP-BF592 のみが対象です。他のプロセッサでは必要ありません)

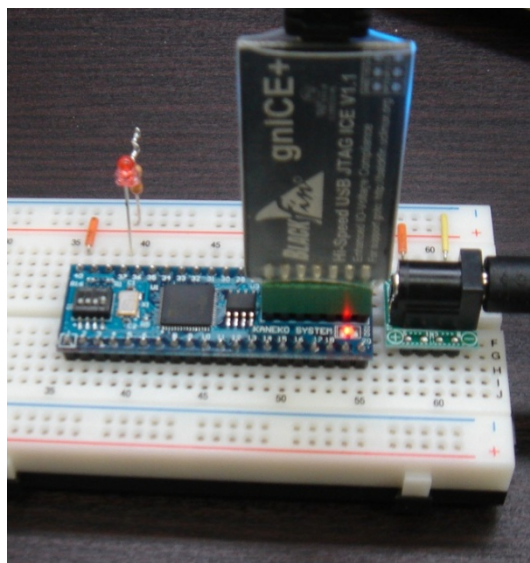


図 4 gnICE+接続例

コマンドプロンプトを起動し、**bf in-gdbproxy** と入力します。Blackfin Toolchain インストール時にパスが設定されるため、実行するフォルダはどこでもかまいません。

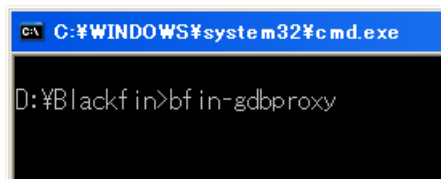


図 5 コマンドプロンプト

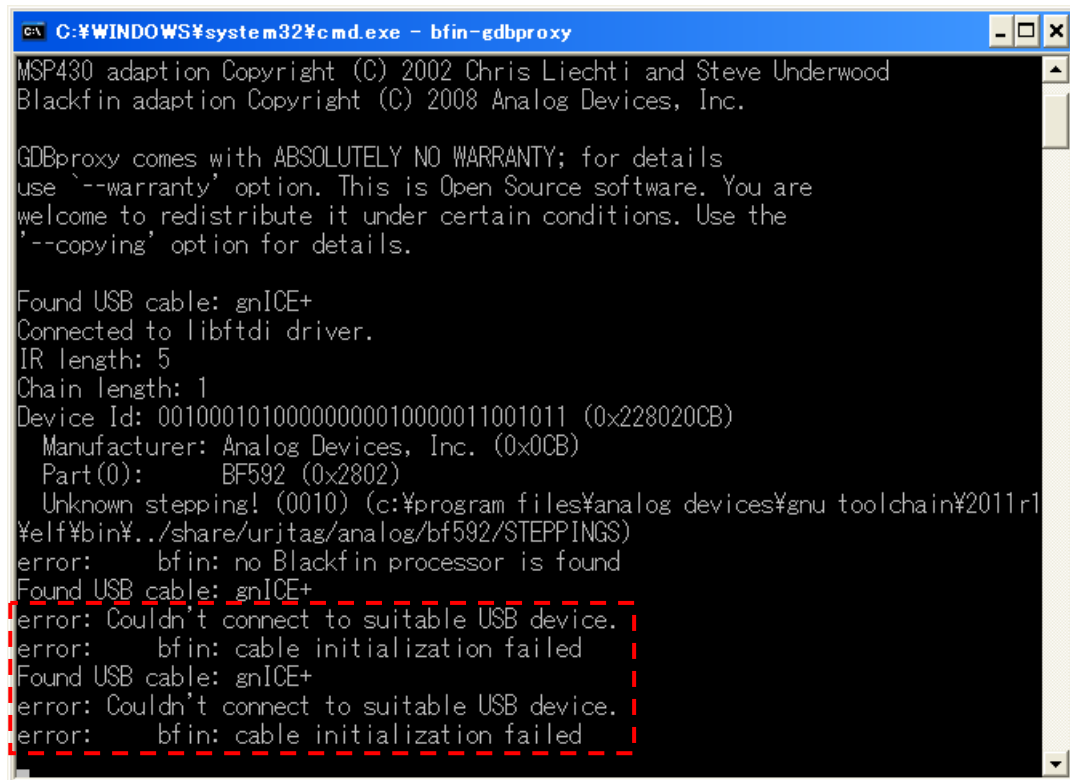


図 6 bfin-gdbproxy エラー画面

上記のエラーが出る場合、設定の修正が必要です。

エクスプローラから、以下のフォルダを開き、STEPPING ファイルをテキストエディタで編集します。

【32bit Windows 版】

C:\Program Files\Analog Devices\GNU Toolchain\2011R1\elf\share\urtag\analog\bf592

【64bit Windows 版】

C:\Program Files (x86)\Analog Devices\GNU Toolchain\2011R1\elf\share\urtag\analog\bf592

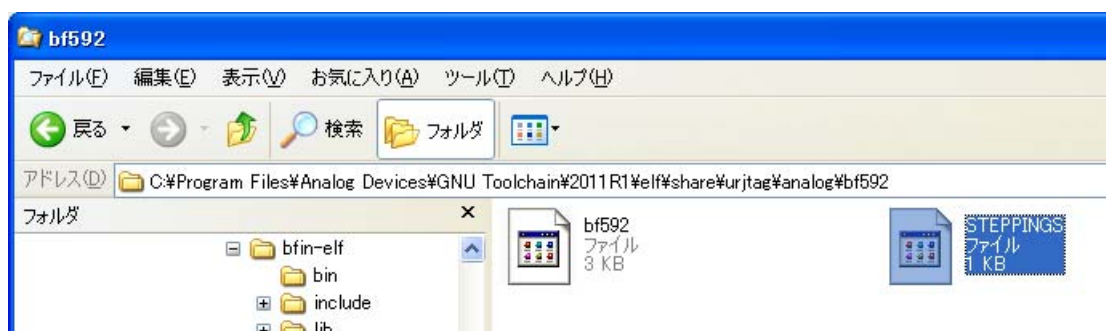


図 7 STEPPING ファイルの格納場所

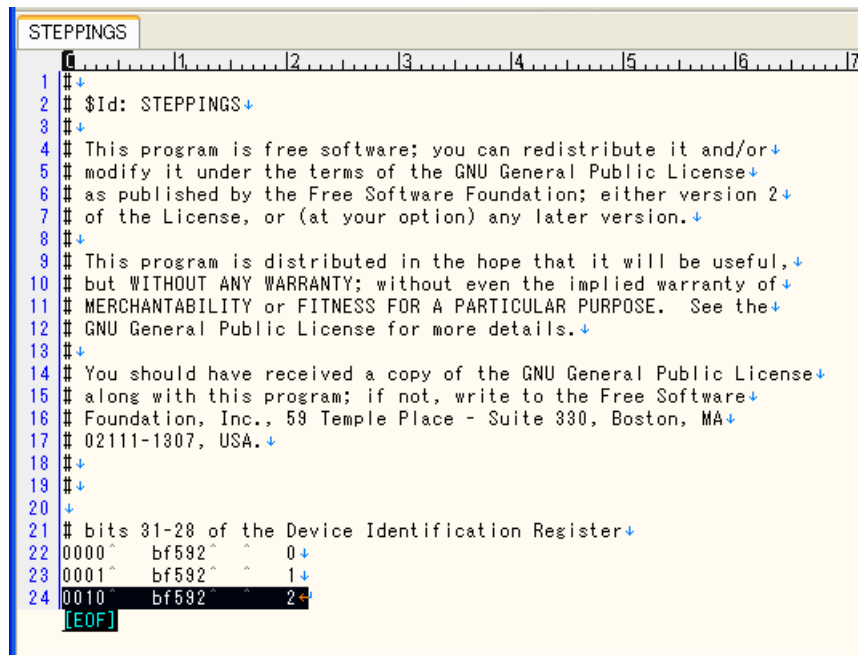


図 8 STEPPING ファイル修正箇所

STEPPING ファイルの再終行に、以下の 1 行を追加します。

```
0010 bf592 2
```

ファイルを保存し、再度コマンドプロンプトから、bfin-gdbproxy を実行し、エラーがないことを確認してください。

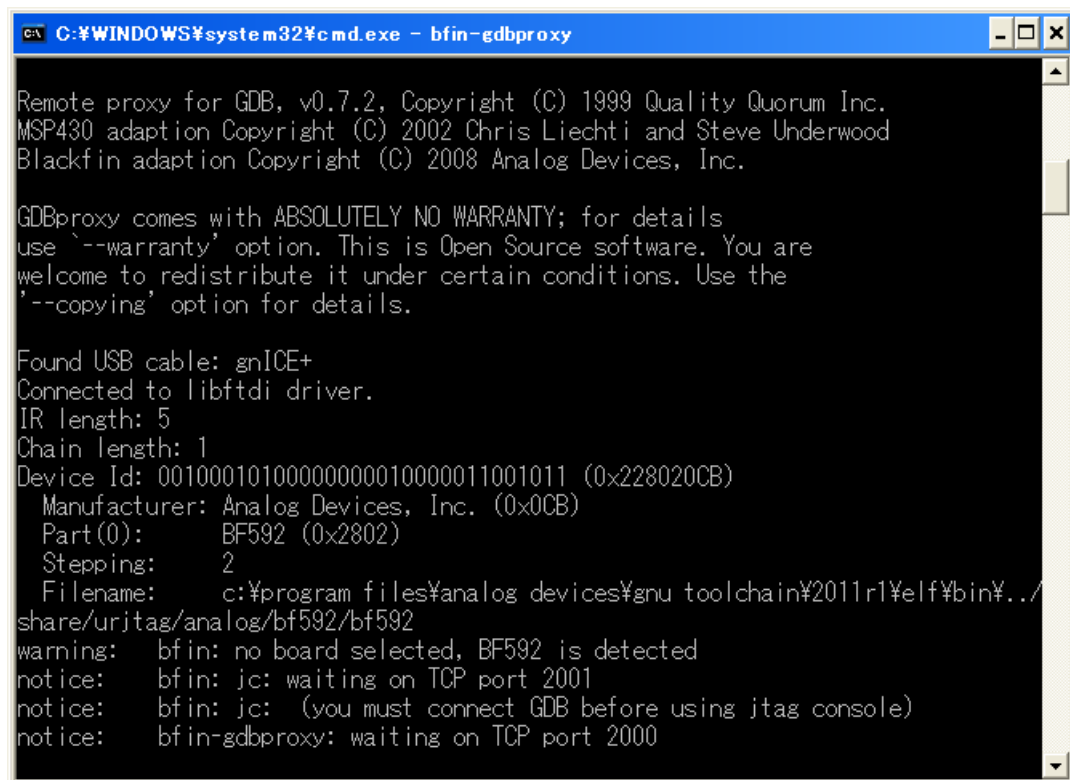


図 9 bfin-gdbproxy の動作画面

3.5 Eclipse のインストール

Eclipse はインストーラーがなく、zip ファイルを解凍すればインストールが完了します。作業するワークフォルダに展開してください。

ここでは、例として” C:\Program Files\Analog Devices\GNU Toolchain”に解凍します。

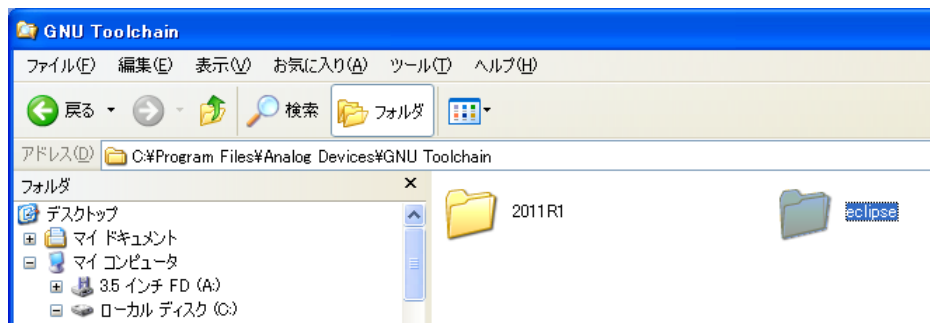


図 10 Eclipse の解凍先フォルダ例

3.6 Eclipse の起動

初回のみ、Eclipse の設定を行う必要があります。

最初に、解凍した eclipse フォルダにある、eclipse.exe を起動します。



図 11 Eclipse.exe アイコン

起動すると、以下のダイアログが表示されます。このフォルダは、ソースやプロジェクトを保存する作業フォルダ（ワークスペース）になりますので、ご使用の環境によって決めてください。以下では、D:\Blackfin フォルダを例に進めます。

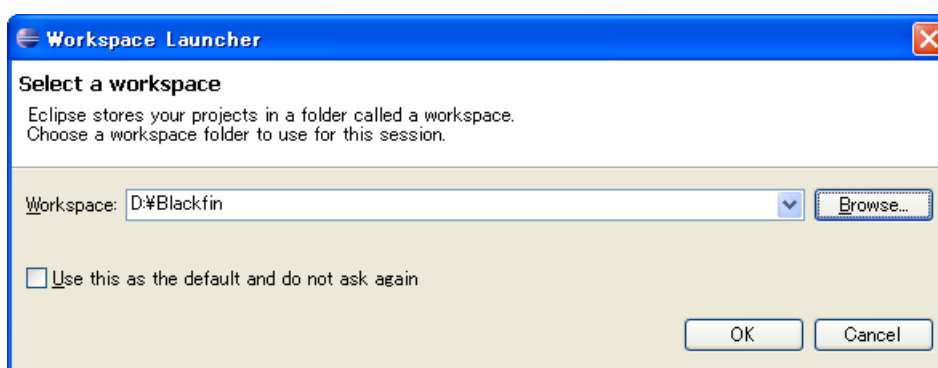


図 12 Eclipse 作業フォルダ指定

作業フォルダの指定が終わると、以下の画面が表示されます。

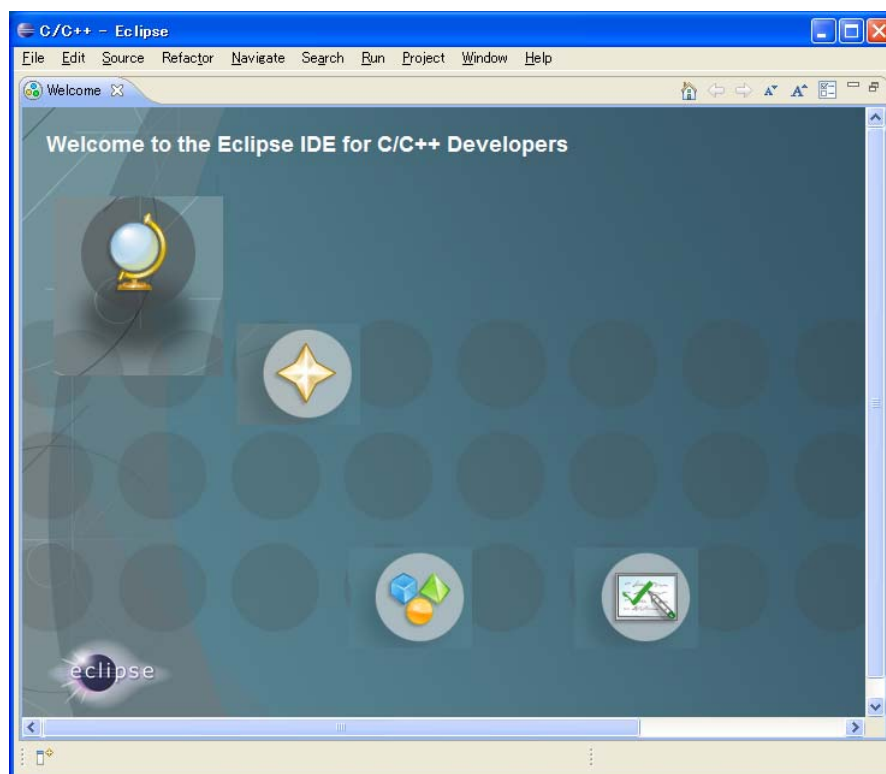


図 13 Eclipse 起動画面

3.7 Blackfin Plugins のインストール

Blackfin Plugins とは、Eclipse で Blackfin Toolchain を便利に使うための拡張機能の事です。このプラグインをインストールしておけば、コンパイラやデバッガ等の設定を手動で設定する必要がなくなり、プログラムと関係のない作業から解放され、簡単にプログラムの作成ができます。

Eclipse のメニュー Window → Preferences を選択します。

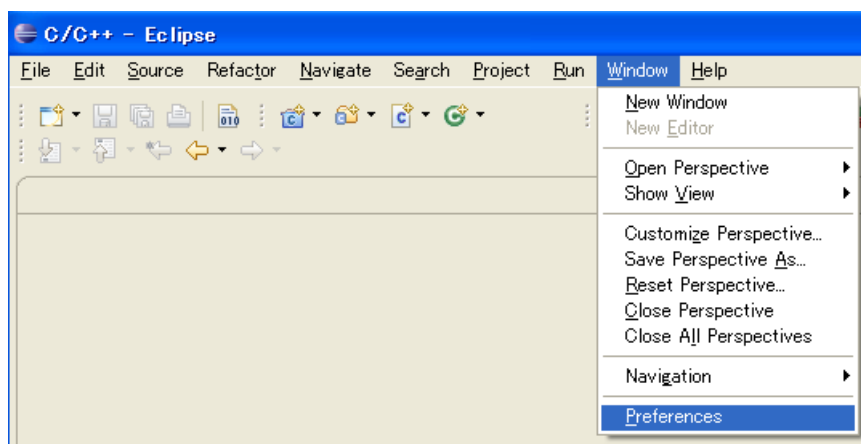


図 14 Eclipse Window メニュー

リストから Install/Update → Available Software Sites を選択し、テキストボックスに、<http://download.eclipse.org/tools/cdt/releases/helios>を入力して、URL のチェックボックスを選択し、OK ボタンを押してダイアログを閉じます。

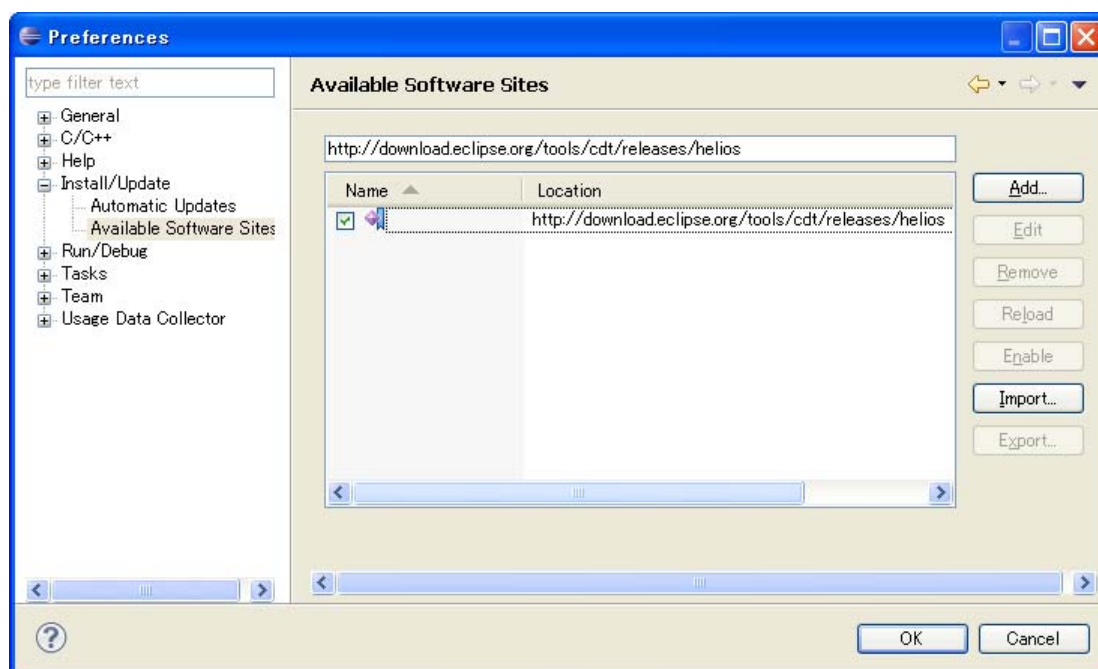


図 15 Eclipse Preference ダイアログ

次に、メニューの Help → Install New Software を選択します。

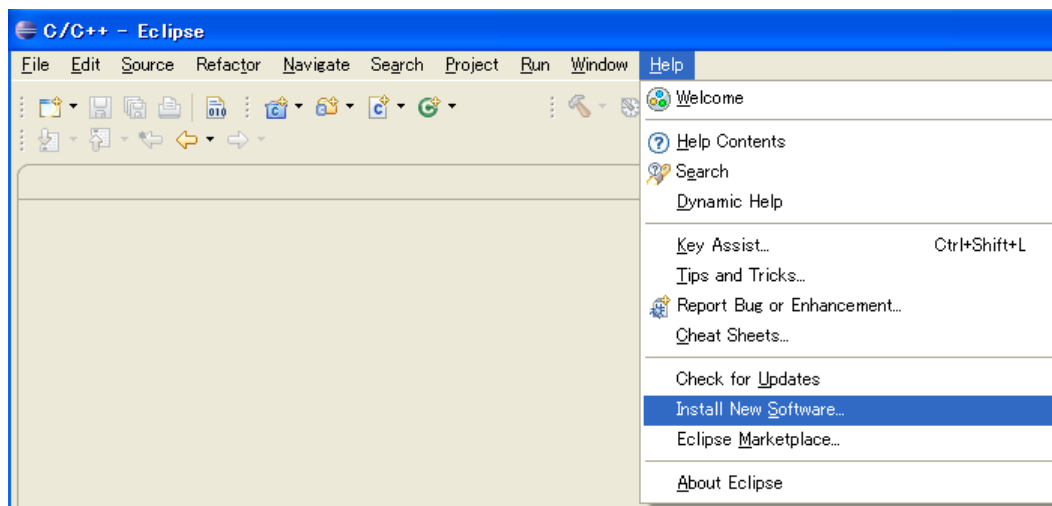


図 16 Eclipse Help メニュー

Work with のリストボックスに、

<http://download.eclipse.org/tools/cdt/releases/helios>

と入力し、CDT Optional Features - C/C++ GDB Hardware Debugging にチェックを入れ、Next ボタンをクリックし、ライセンス条項を確認して Finish ボタンをクリックします。インストール後、Eclipse を終了し、再起動してください。

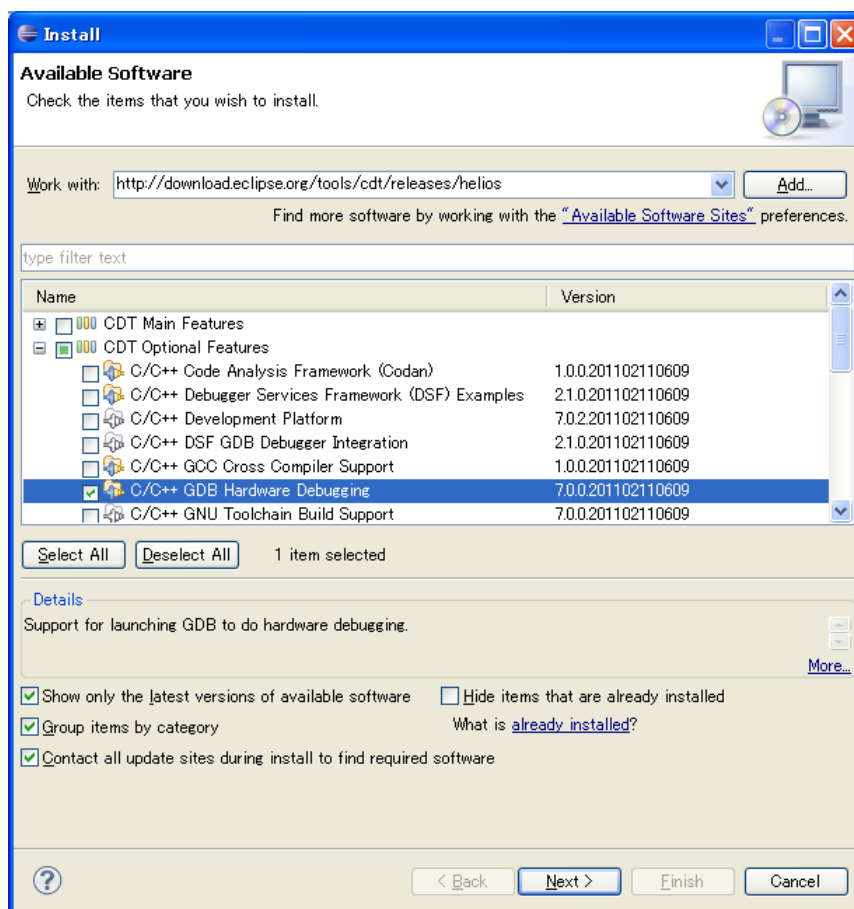


図 17 Eclipse Install ダイアログ

再度、メニューの Help → Install New Software を選択します。

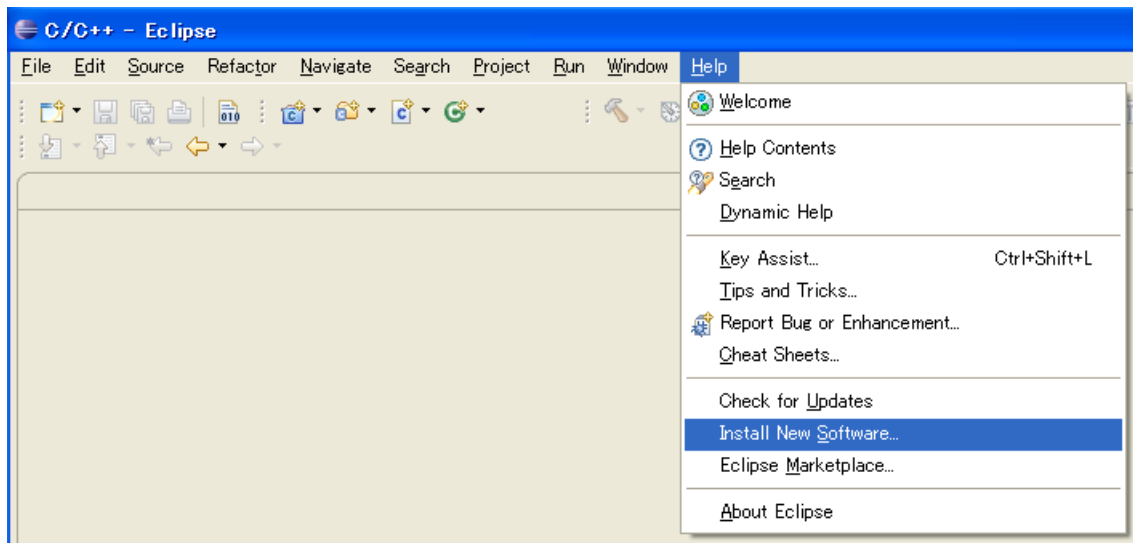


図 18 Eclipse Help メニュー

Install ダイアログの Add ボタンを押して、以下のように入力します。

Name: Blackfin Linux

Location: <http://blackfin.uclinux.org/eclipse/>

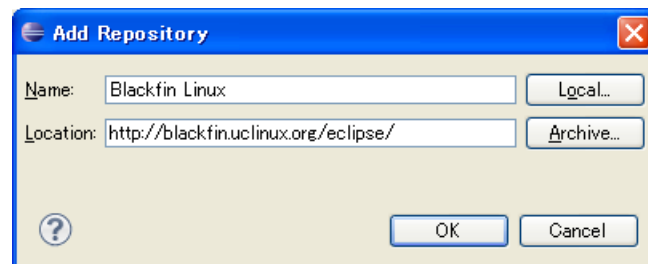


図 19 Add Repository

<http://blackfin.uclinux.org/eclipse/>から正しくデータが取得できると、以下のようにリストが更新されます。Blackfin DebugとBlackfin GNU Toolchainにチェックを入れて、Nextボタンをクリックし、ライセンス条項を確認した後、Finishボタンをクリックします。インストール後、Eclipseを終了し、再起動してください。

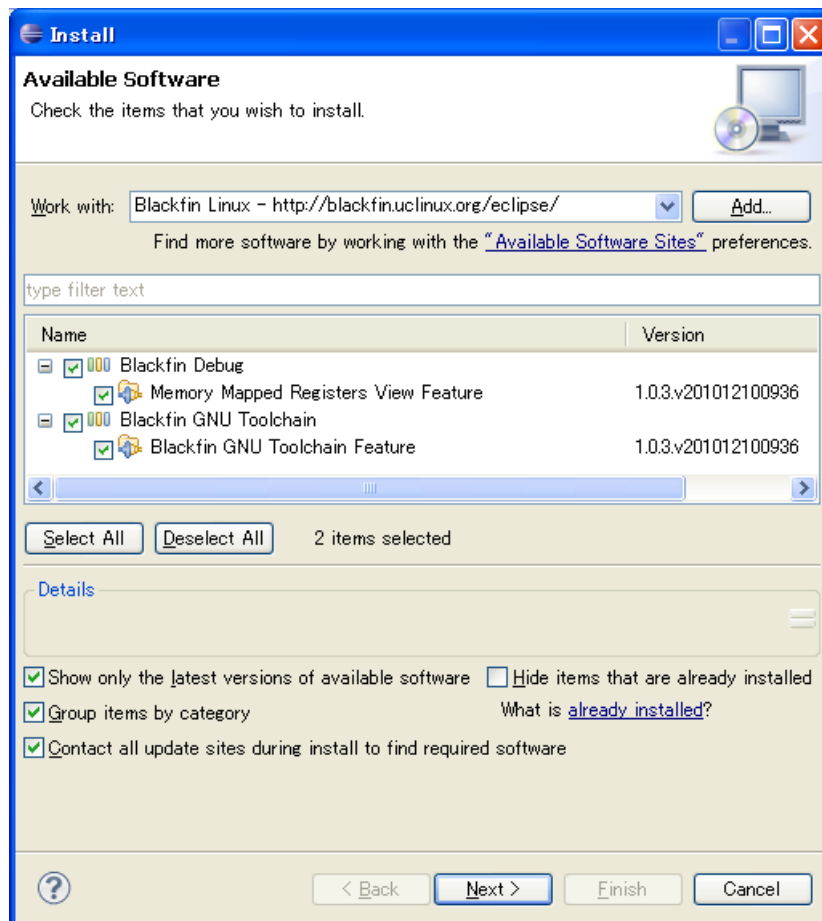


図 20 Eclipse Install ダイアログ

以上でインストールは完了です。

4 Blackfin アプリケーションの作成

ここでは、Blackfin アプリケーションの作成からデバッグまでの手順を解説します。

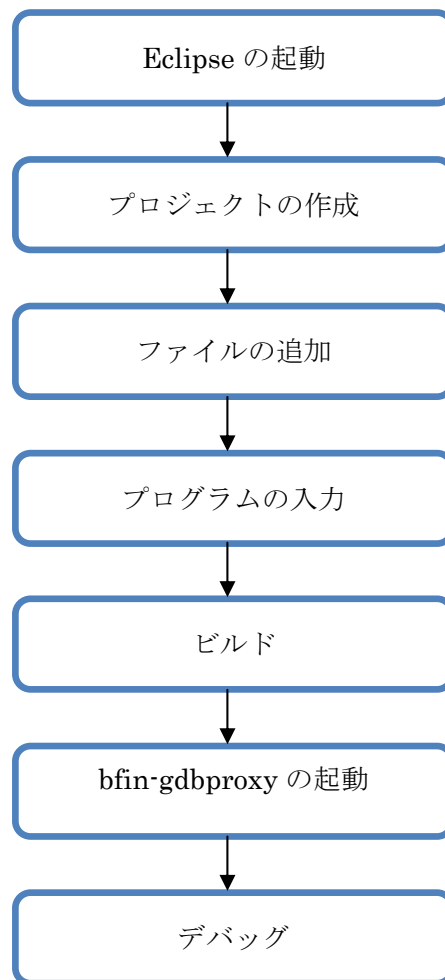


図 21 Eclipse での開発の流れ

4.1 プロジェクトの作成

プロジェクトを作成します。

Eclipse を起動し、New → C Project を選択します。

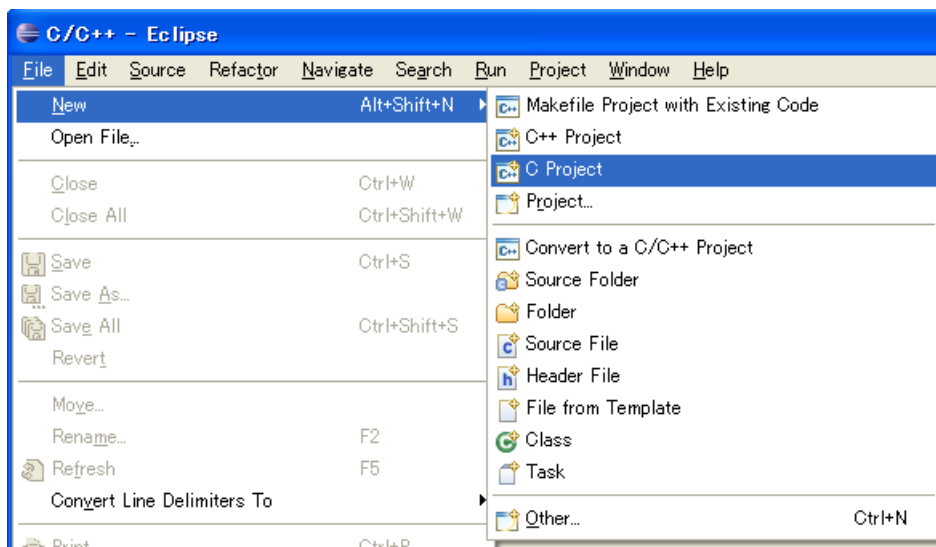


図 22 Eclipse New C Project

プロジェクト名に、led_blink と入力します（この名前はアプリケーションによって変更してください）。Project Type には Empty Project、Toolchains には Blackfin Bare Metal (bfin-elf) を選択し、Next ボタンをクリックします。

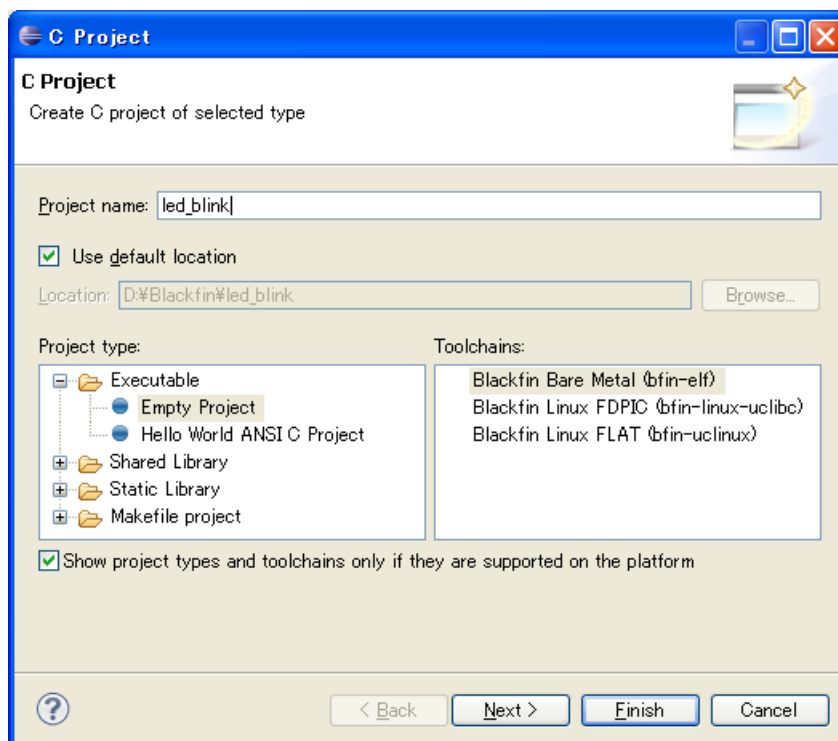


図 23 C Project ダイアログ

Select Configurations 画面では、そのまま Next ボタンをクリックします。

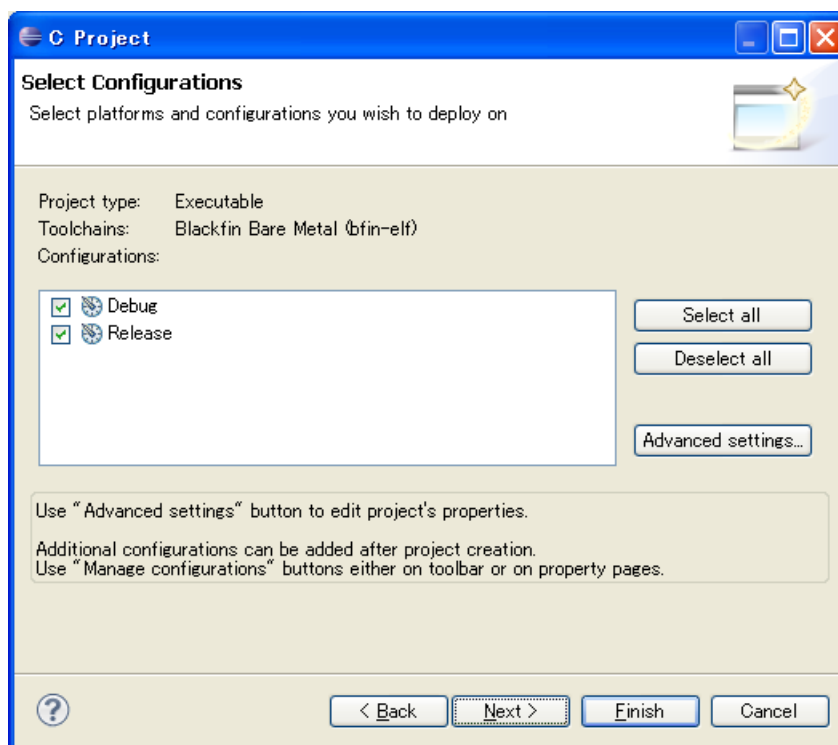


図 24 C Project ダイアログ

Select Processor画面では、使用するBlackfinプロセッサを選択します。ここではBF592 を選択します（プロジェクト作成後でも変更可能です）。Silicon Revisionはanyにします。Silicon Revisionについては、「5 Silicon Revisionについて」を参照してください。

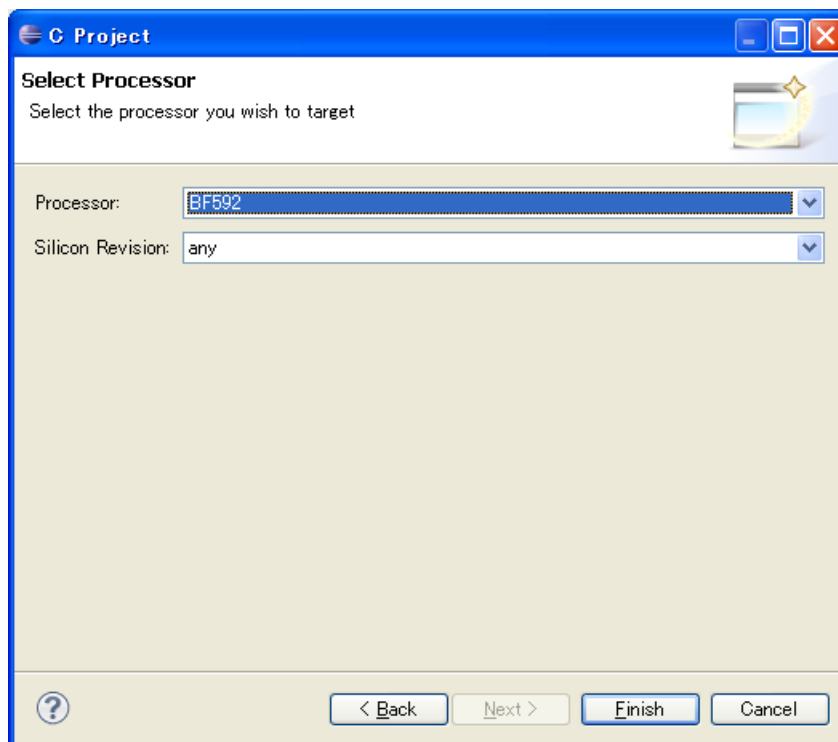


図 25 C Project ダイアログ

プロジェクトを作成すると、Project Explorer にプロジェクトが追加されます。

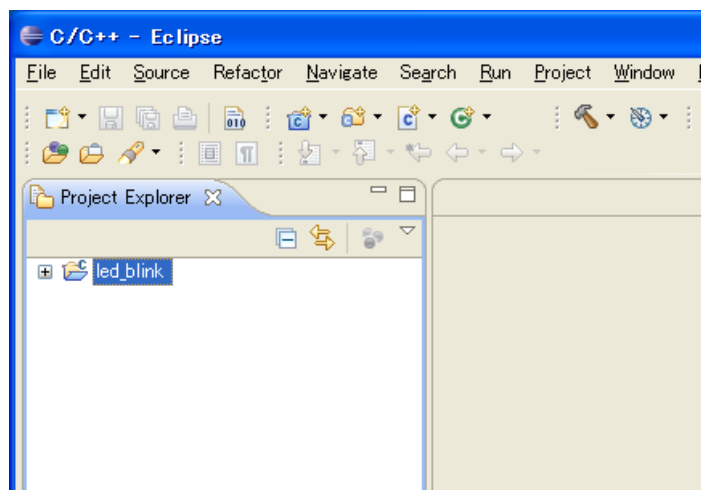


図 26 Project Explorer

プロジェクトには何もありませんので、ソースファイルを追加します。プロジェクトを右クリックしてメニューを開き、New → Source File をクリックします。

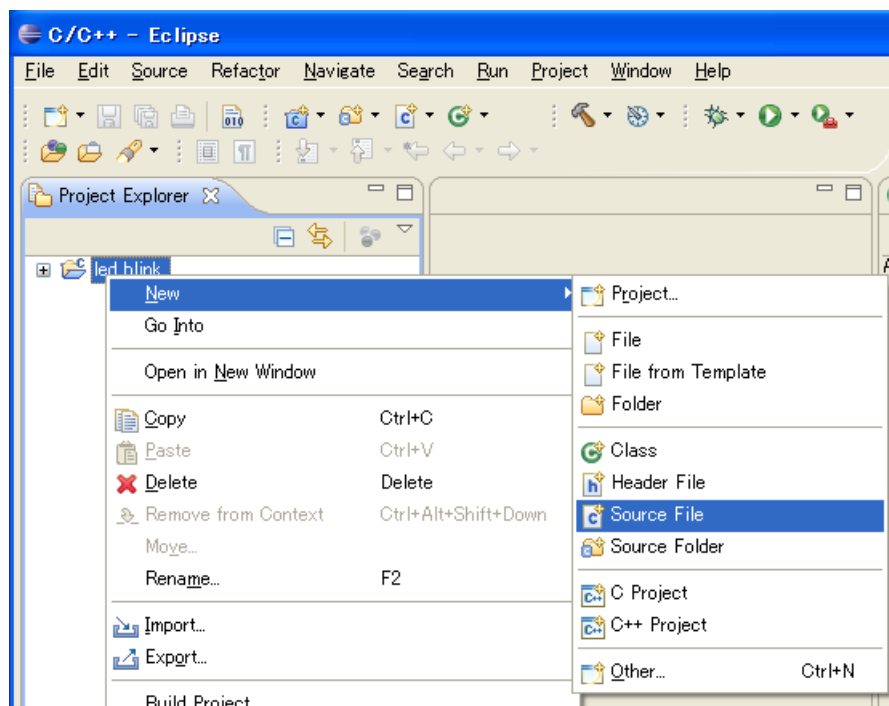


図 27 Source File の追加

追加するファイル名を Source file テキストボックスに入力します。ここでは、main.c とし、Finish ボタンを押します。

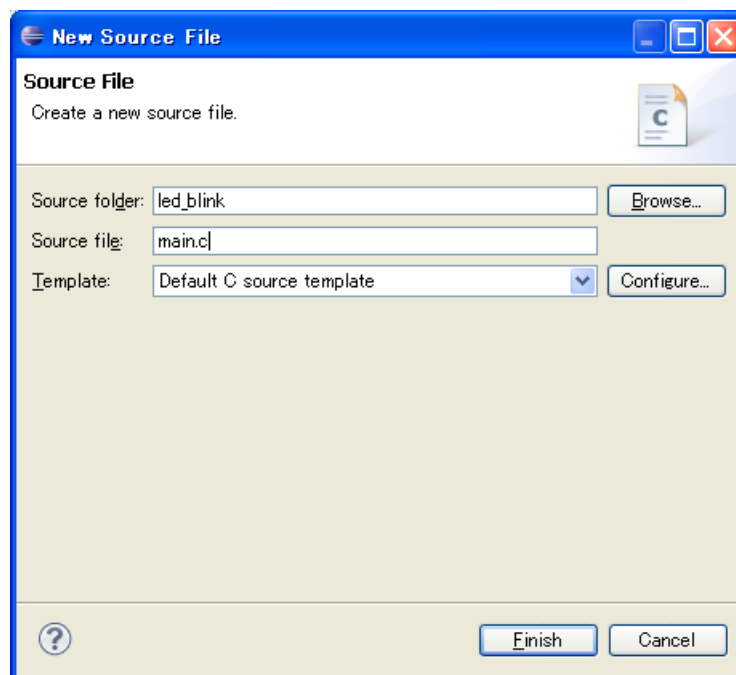


図 28 Source File の追加

プロジェクトに main.c が追加されます。

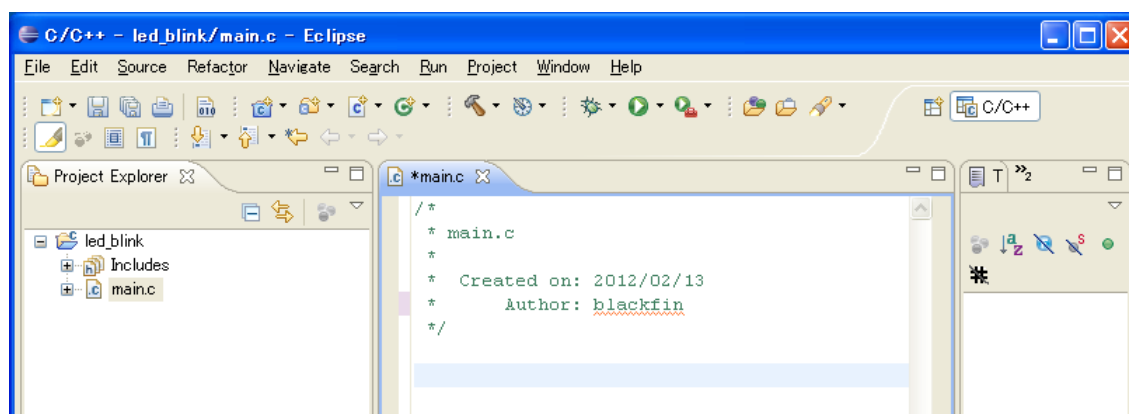


図 29 main.c の追加

これで、プログラムの入力準備が整いました。実際にプログラムを入力してみます。ここでは、ACB-BF592 の 37 番ピンに接続された LED を点滅させるプログラム例です。

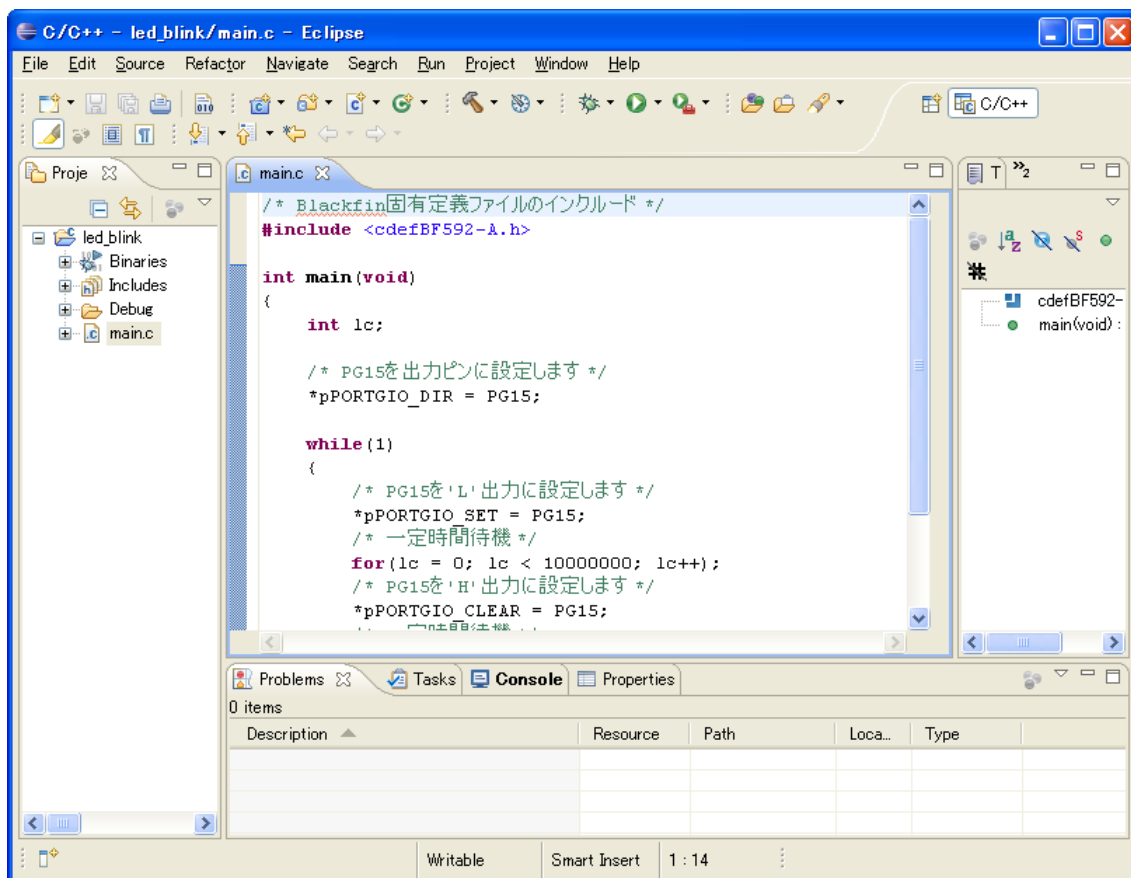


図 30 プログラム入力例

```
/* Blackfin固有定義ファイルのインクルード */
#include <cdefBF592-A.h>

int main(void)
{
    int lc;

    /* PG15を出力ピンに設定します */
    *pPORTGIO_DIR = PG15;

    while(1)
    {
        /* PG15を'L'出力に設定します */
        *pPORTGIO_SET = PG15;
        /* 一定時間待機 */
        for(lc = 0; lc < 100000000; lc++);
        /* PG15を'H'出力に設定します */
        *pPORTGIO_CLEAR = PG15;
    }
}
```

```

        /* 一定時間待機 */
        for(lc = 0; lc < 100000000; lc++);
    }
    return 0;
}

```

プロジェクトをビルドするには、Ctrl+B を押すか、メニュー Project → Build All を選択します。

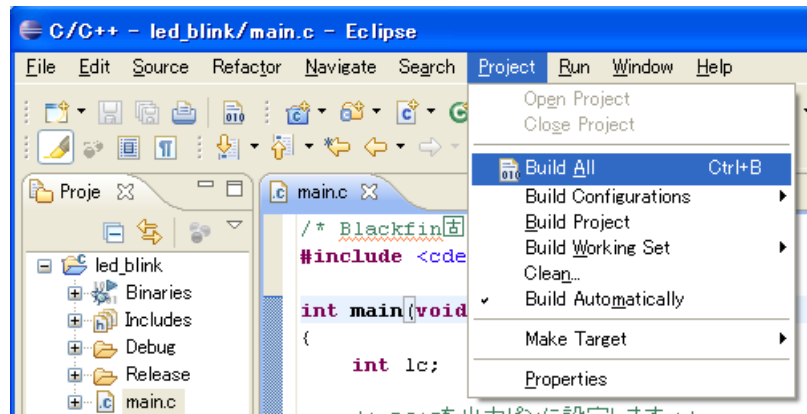


図 31 ビルド

下部の Problem タブのリストに、エラーが表示されなければビルド完了です。

4.2 gdb によるデバッグ

ここでは gnICE+を使って、Eclipse からデバッグを行う方法を説明します。ACB-BF592 に gnICE+を接続し、電源を入れてください。

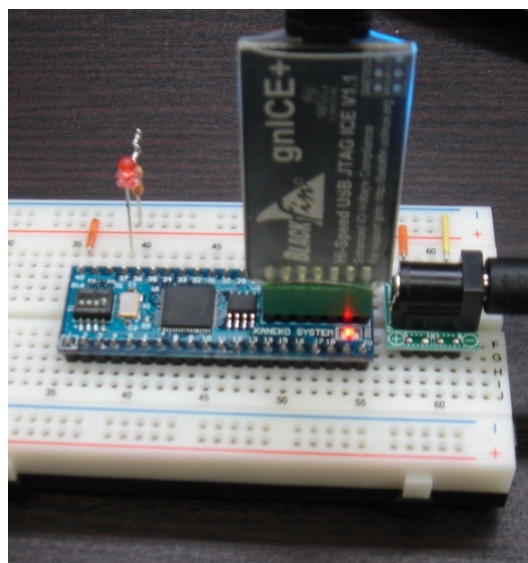


図 32 gnICE+接続例

最初に bfin-gdbproxy を起動する必要があります。

コマンドプロンプトを起動し、**bfin-gdbproxy** と入力します。

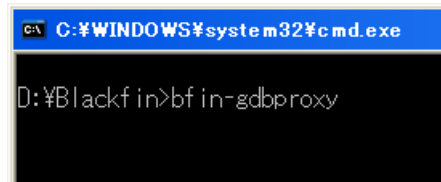


図 33 コマンドプロンプト

※bfin-gdbproxyを実行してエラーが出る場合は、「3.4 Blackfin Toolchainの設定ファイルの修正」を参照してください。

Eclipse のデバッグの設定を行います。虫アイコンの横にある▼をクリックしてメニュー表示させ、Debug Configuration を選択します。

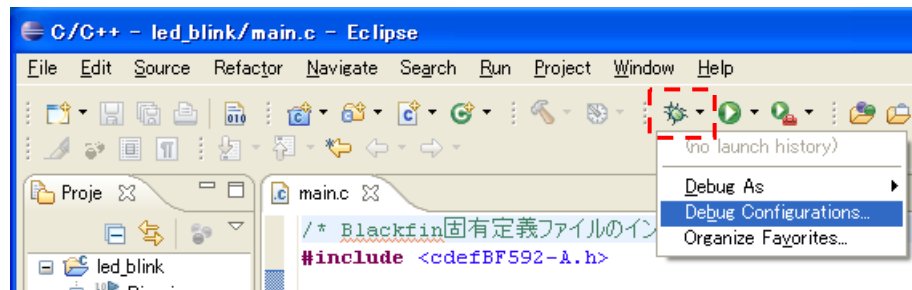


図 34 Debug Configuration メニュー

リストの Blackfin Bare Metal Application をダブルクリックし、led_blink Debug を作成します。

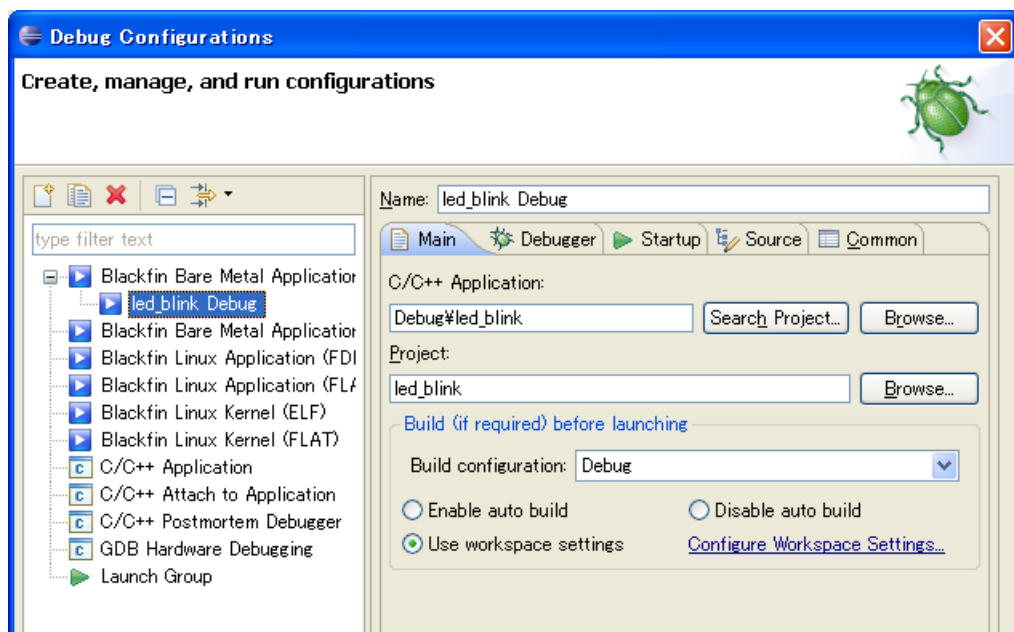


図 35 Debug Configuration メニュー

Debugger タブを選択し、Remote Target の Port Number を 2000 に変更します。
Apply を押して設定を保存し、Debug ボタンを押します。

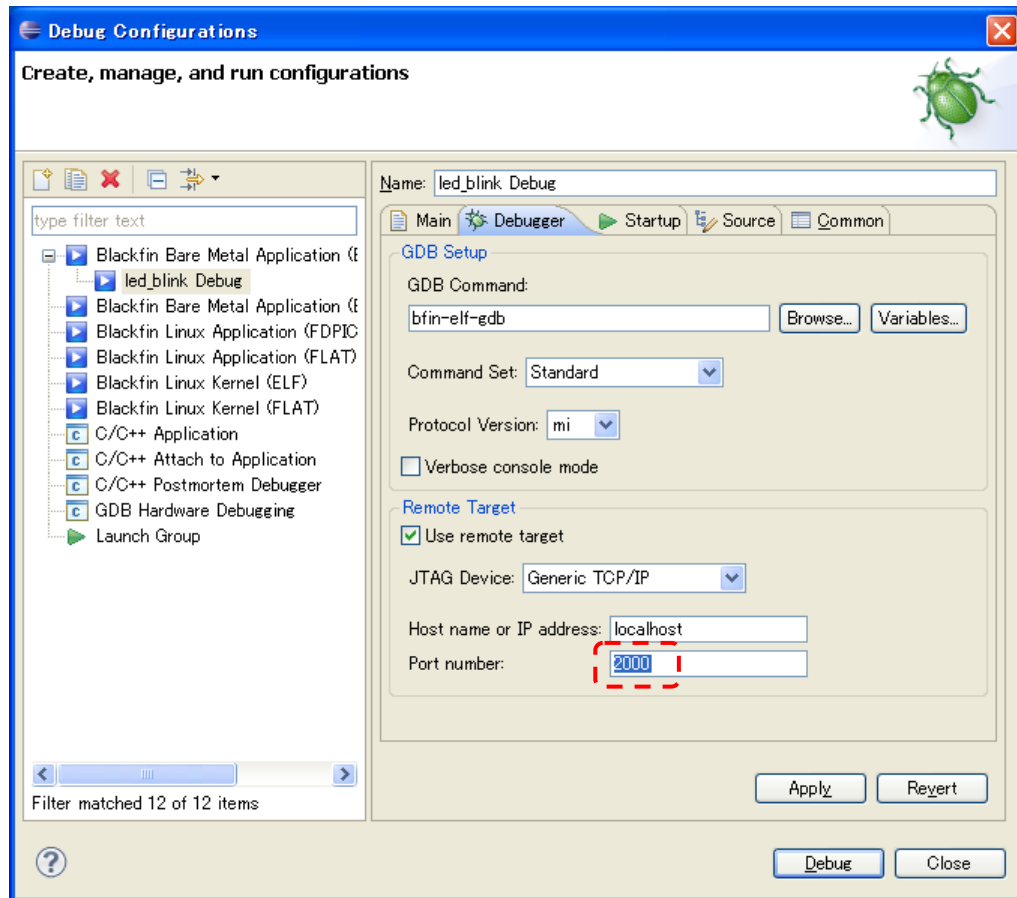


図 36 Remote Target の Port Number

Eclipse のデバッグ画面に変更するかどうか聞かれますので、Yes を押します。

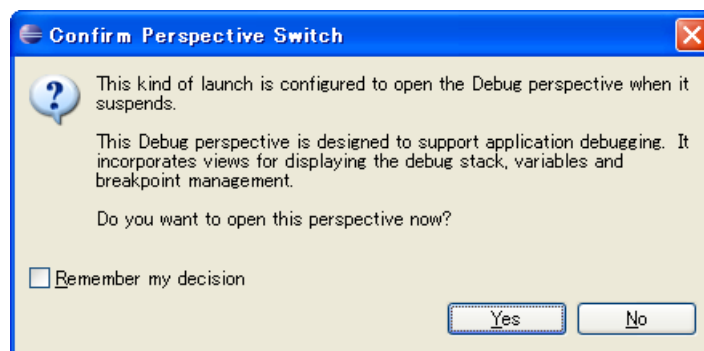


図 37 Confirm Perspective Switch ダイアログ

問題がなければ、main の最初のコードで停止します。

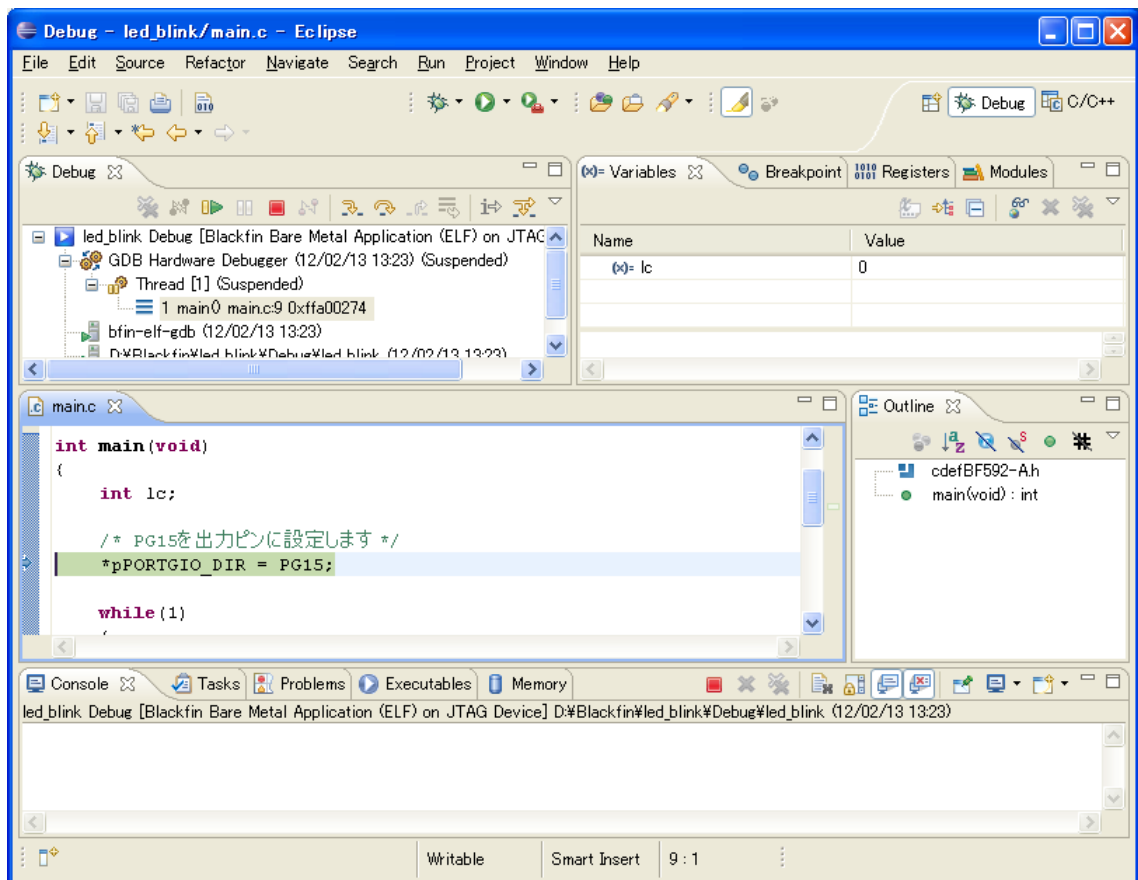


図 38 ステップ実行画面

この状態でステップ実行が可能です。そのまま実行させると、ACB-BF592 の 37 番ピンに接続された LED が点滅します。

5 Silicon Revision について

シリコンリビジョンとは、Blackfin チップのバージョンの事です。0.0 から始まり、新しいほど数字が大きくなります。最新の ADSP-BF592 では特に目立った問題はありませんが、ADSP-BF53x や ADSP-BF51x などの古いリビジョンでは、DMA の優先順位がコア優先から変更できない、内蔵 SRAM のメモリ配置にミスがあるなどの問題があります。具体的な内容は、Analog Devices 社のサイトに IC アノマリとして公開されており、一度確認されることをお勧めします。

<http://www.analog.com/jp/processors-dsp/Blackfin/processors/ic-anomalies/resources/index.html>

最近では Blackfin プロセッサのバグもかなり修正されており、どの Blackfin プロセッサでも問題ないのですが、特に理由がなければ、新しいリビジョンのチップを使用してください。

なお、コンパイラに Silicon Revision を指定するのは、これらのバグを回避する対策を指示するためで、any を指定すると全リビジョンに対応しますが、コード又は、データサイズが若干増加するケースがあります。

6 更新履歴

版	更新日	更新内容
第 1 版	2011/02/13	初版発行