

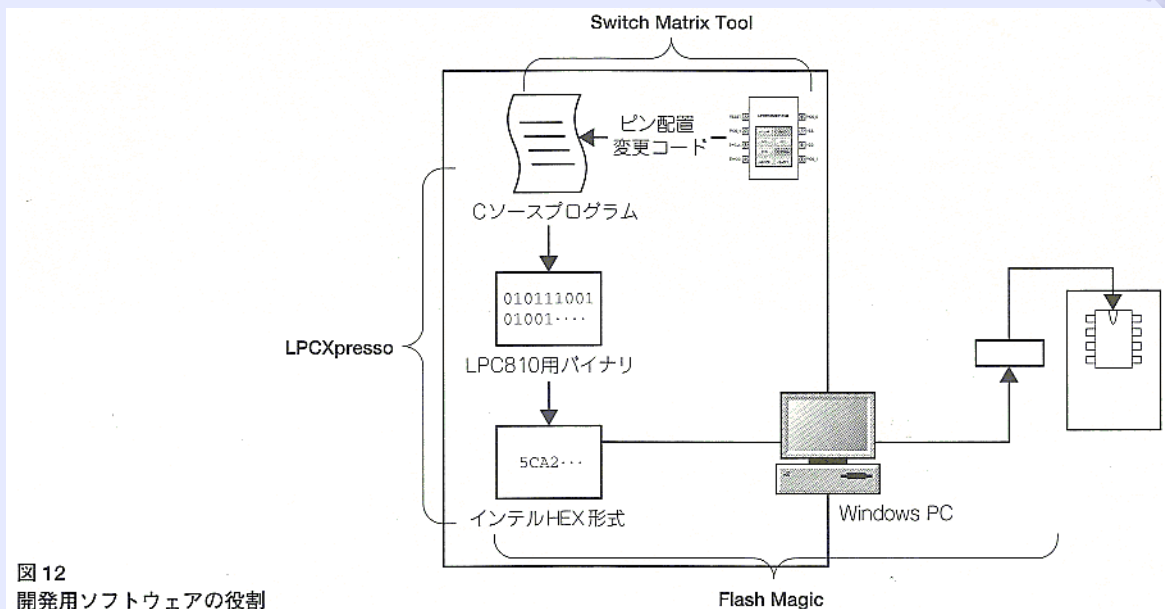
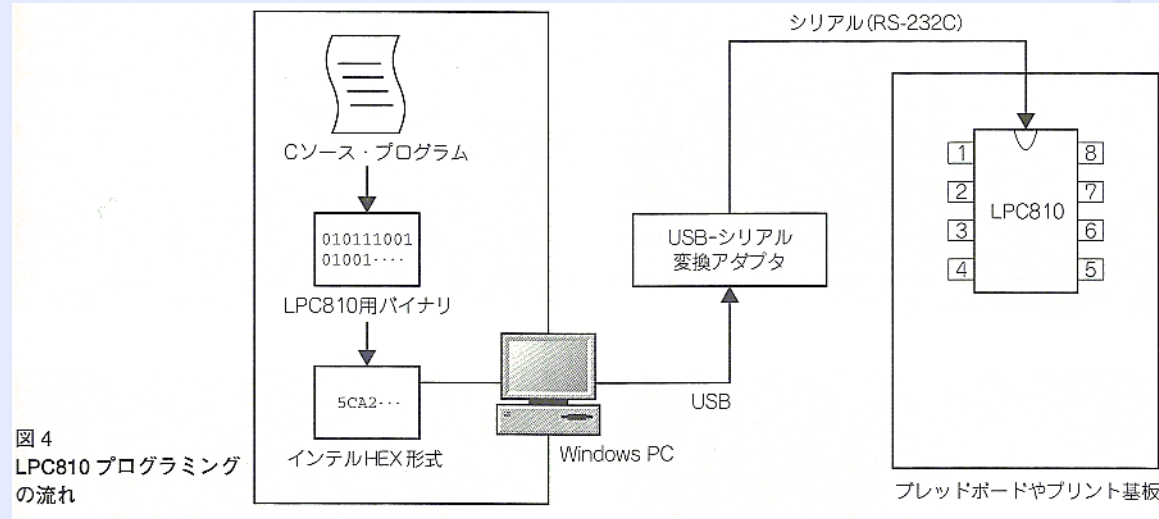
組込ソフトウェア工学

第5～8回

1. デザインウェーブマガジン編集部, ”組み込みソフトウェア開発スタートアップ
—ITエンジニアのための組み込み技術入門 (デザインウェーブムック)”, CQ出版, 2005.
2. 中村文隆, ”挿すだけ！ARM32ビット・マイコンのはじめ方”, CQ出版, 2015.

1. 開発環境(全体像)

この演習では、代表的なマイコンの一種であるARMの中の最もシンプルな製品を使用して、初歩的なLED点滅プログラムを作成する。



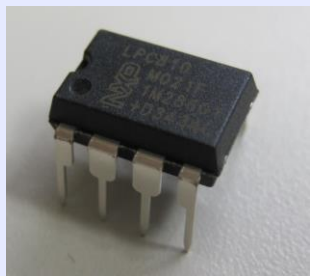
2. 開発環境(ハードウェア)

基盤の実装面や各部品の金属部分をできるだけ触らないこと。静電気と紛失に注意。

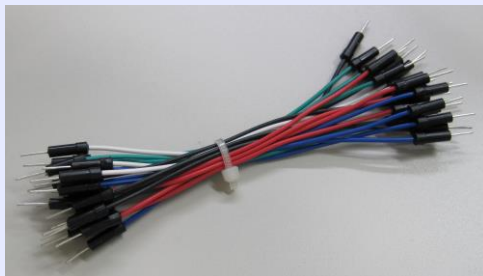
- A) マイコン **LPC810M021FN8 (1個)**: 半導体大手の**NXP Semiconductors**が製造するARM32ビット・8ピンDIPパッケージ。最大30MHz, 電源3.3V, Flashメモリ 4KB, SRAM 1KB.
- B) ICソケット **2227MC-08-03 (1個)**: マイコンに装着する。ブレッドボードへの抜き差しを繰り返すことによってマイコンの足が折れるのを防ぐ。
- C) ブレッドボード **BB-801 (1個)**: 一般的な導通パターンのブレッドボード。
- D) ジャンパーコード **(20本セット)**: ブレッドボード上の配線に使用する。
- E) **FT232RL搭載小型USB-シリアルアダプタ 3.3V (2名で1個)**: マイコンをPCに接続してプログラム書き込みと電源供給を行う。
- F) **USB2.0ケーブル A-miniBタイプ (2名で1本)**: 上記アダプタとPCを接続するケーブル。
- G) 精密ピンセット **TS-10 (2名で1個)**: 部品を扱う際に使用する。先端を保護するためのキャップを紛失しないように注意。
- H) 赤色LED **OSR5CA5B61P (1個)**: 配線の際は向きに注意。足の長いほうがアノード(+), 短いほうがカソード(-)。
- I) 黄色LED **OSY5EA5B61A-QR (1個)**: 同上。
- J) 炭素皮膜抵抗**1/4W 100Ω (2本)**: LEDに流れる電流を制限する。これがないと壊れる。

2. 開発環境(ハードウェア)

A)



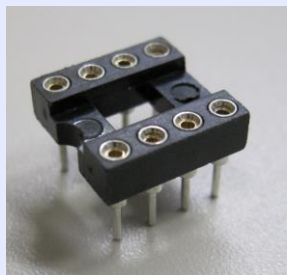
D)



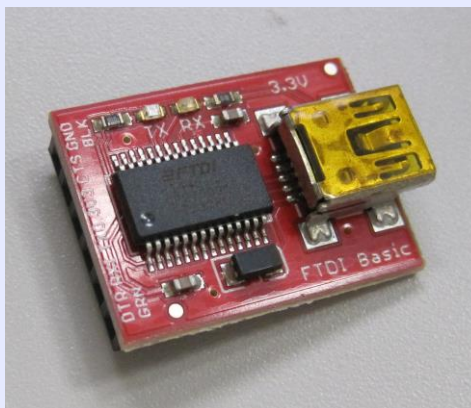
G)



B)



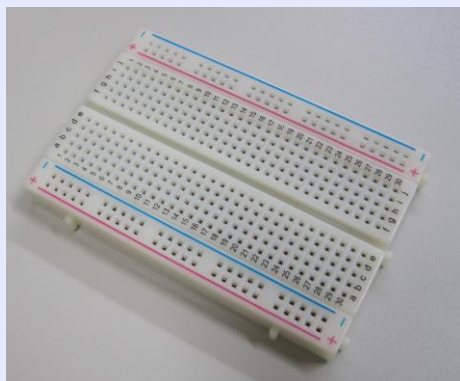
E)



H)



C)



F)



I)

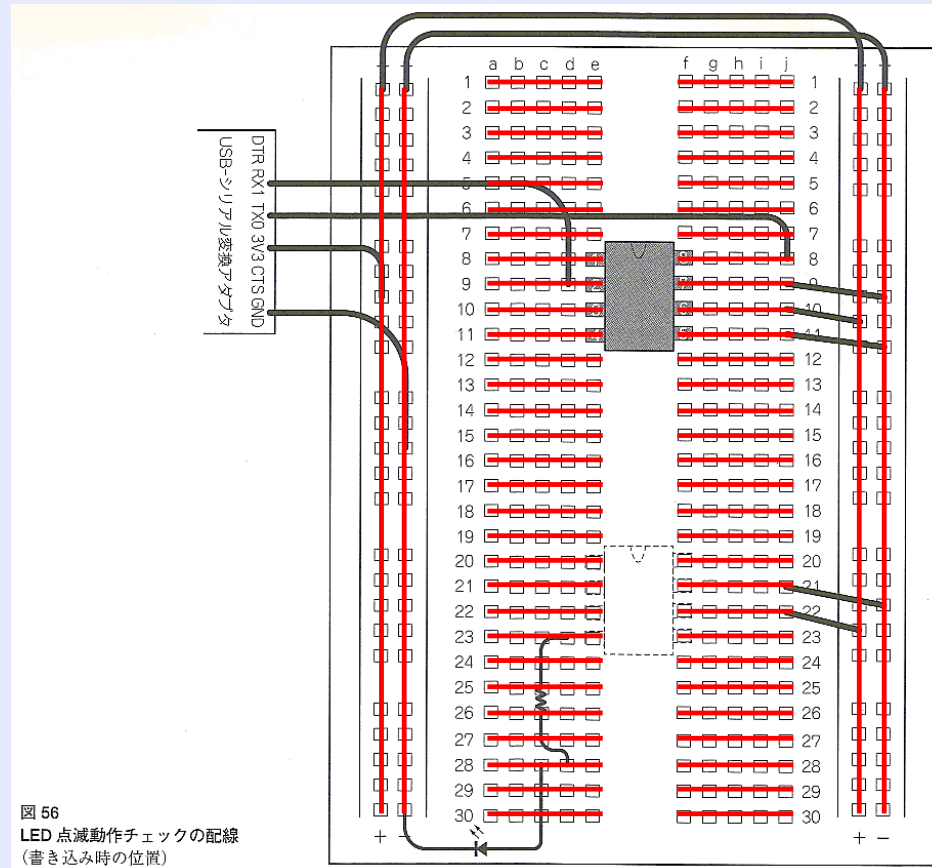


J)



2. 開発環境(ハードウェア)

- ◆ ブレッドボードは回路を試作する際に利用される(部品をいきなり基盤にハンダ付けしてしまうと、試行錯誤ができない).
- ◆ ブレッドボード内部は以下のように導通している.



2. 開発環境(ハードウェア)

- ◆ まず下図のように配線する. 左はプログラム書き込み時, 右は動作チェック時を表す. マイコンを挿し替えるだけで, 書き込みと動作チェックの切り替えが可能.
プログラム書き込み時と動作チェック時以外は電源を落としておくこと(GNDを外しておくこと). 特に, 電源が入ったままの状態ではマイコンを挿し替えないこと.
- ◆ LED 1個と抵抗1個はここでは使用しない(最後の演習で使用する).

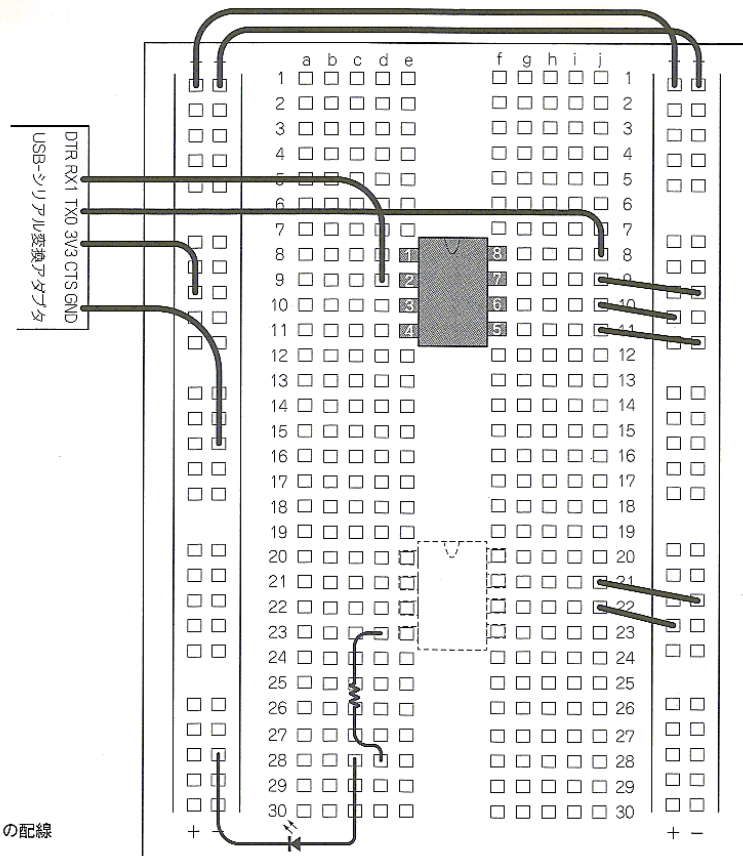


図 56
LED 点滅動作チェックの配線
(書き込み時の位置)

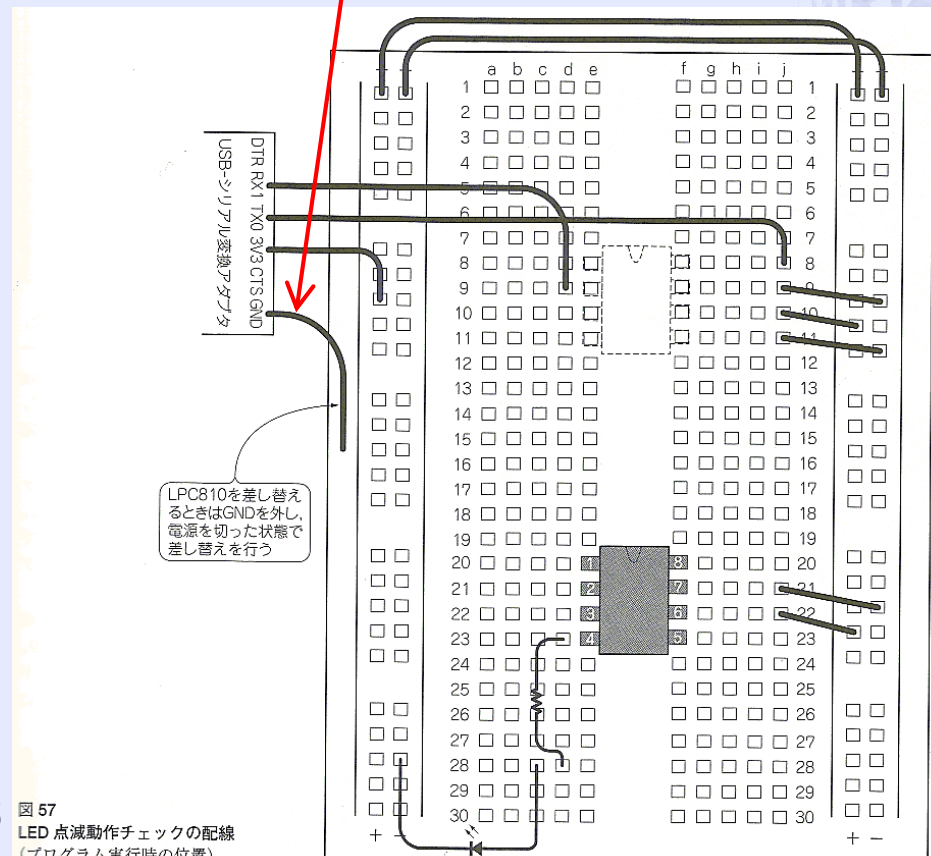


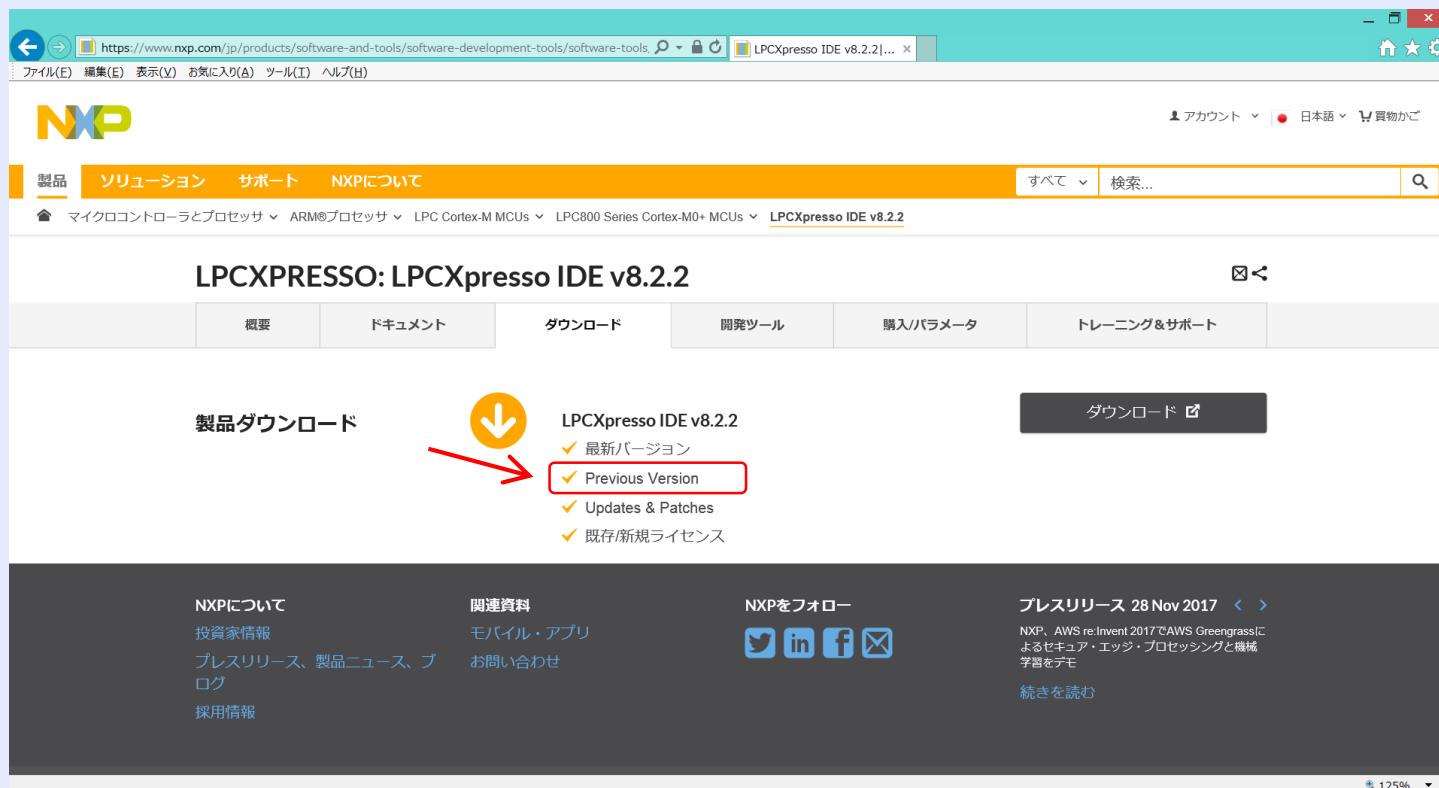
図 57
LED 点滅動作チェックの配線
(プログラム実行時の位置)

3. 開発環境(ソフトウェア)

- A) **LPCXpresso(Free Edition)**: Cソースファイルを作成し、最終的にインテルHEX形式ファイル(マイコン用の実行ファイル)を生成するために使用するEclipseベースのIDE. NXP Semiconductorsの公式Webサイト https://www.nxp.com/jp/products/software-and-tools/software-development-tools/software-tools/lpc-microcontroller-utilities/lpcxpresso-ide-v8.2.2:LPCXPRESSO?&tab=Design_Tools_Tab&lang=jp (次ページ参照)からダウンロードしてインストールすること. 本演習はv7.8.0の利用を想定しているが, 新しい版のv8.2.2の利用も可能. インストール設定はデフォルトでよい. アクティベーションはしなくてもよい.
- B) **Flash Magic**: 本演習では, LPC810にシリアル経由でインテルHEX形式ファイルを書き込む際に使用する. <http://www.flashmagictool.com/> からダウンロードしてインストールすること. 本演習はv9.2の利用を想定しているが, 新しい版の利用も可能. インストール設定はデフォルトでよい.
- C) **FT232RL用ドライバ**: USB-シリアル変換アダプタを使用するために必要. <http://www.ftdichip.com/Drivers/VCP.htm> からダウンロード後, 任意の場所で解凍し, アダプタをPCに接続すると自動でインストールされる. 自動でできない場合は, デバイスマネージャ中のアダプタのエントリ(!マーク付きのFT232あるいはUSB Serial Port)を右クリックして, ドライバ・ソフトウェアの更新を実行するか, トラブルシューティングを実行する. Flash Magicを使用する際に必要なので, USB Serial PortのCOM番号を確認しておくこと. これはインストール完了時のメッセージ, またはインストール後デバイスマネージャから確認可能.

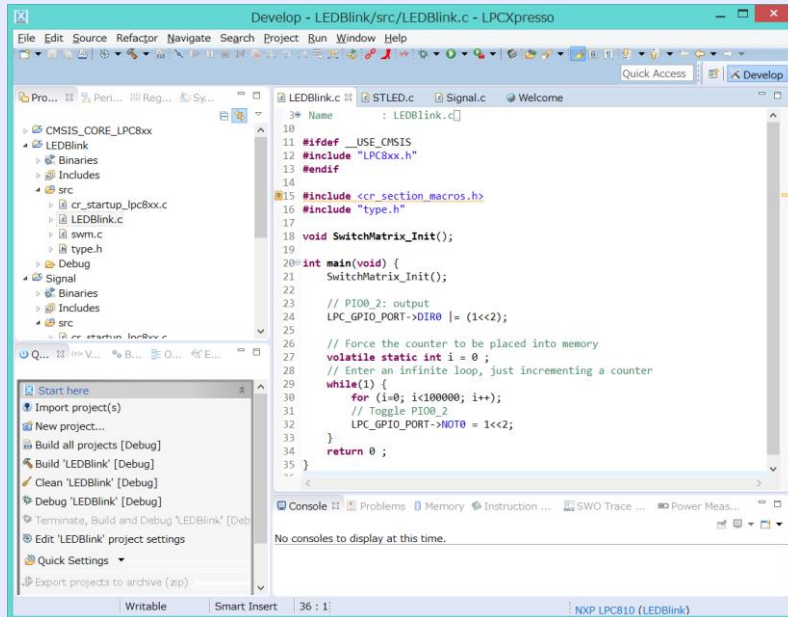
3. 開発環境(ソフトウェア)

NXP Semiconductorsの公式Webサイト(下図)中の「Previous Version」を選択すると、ログイン画面に遷移する。ログインアカウントを作成するとダウンロード画面に遷移するので、利用規約を確認後、v7.8.0のインストールファイルを選択して入手する。

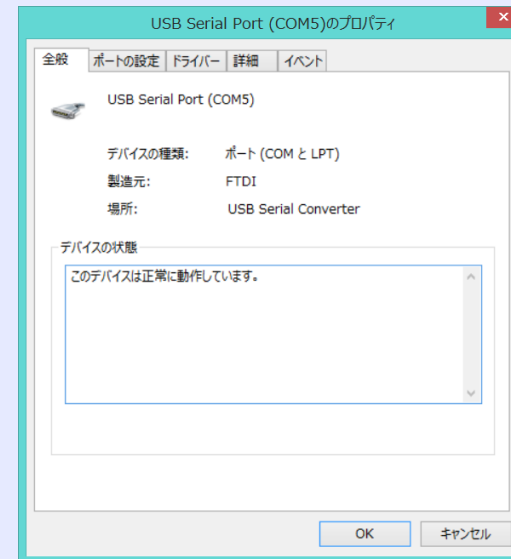


3. 開発環境 (ソフトウェア)

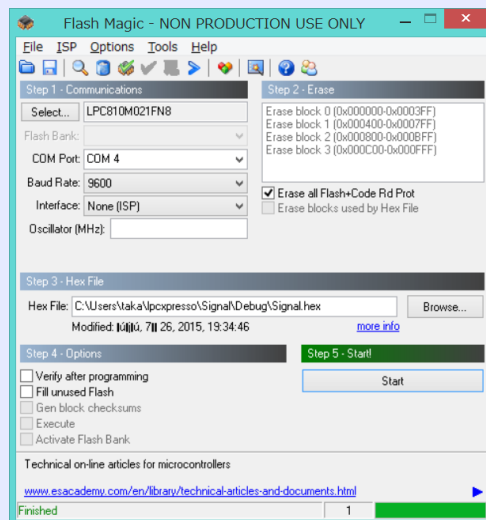
A)



C)

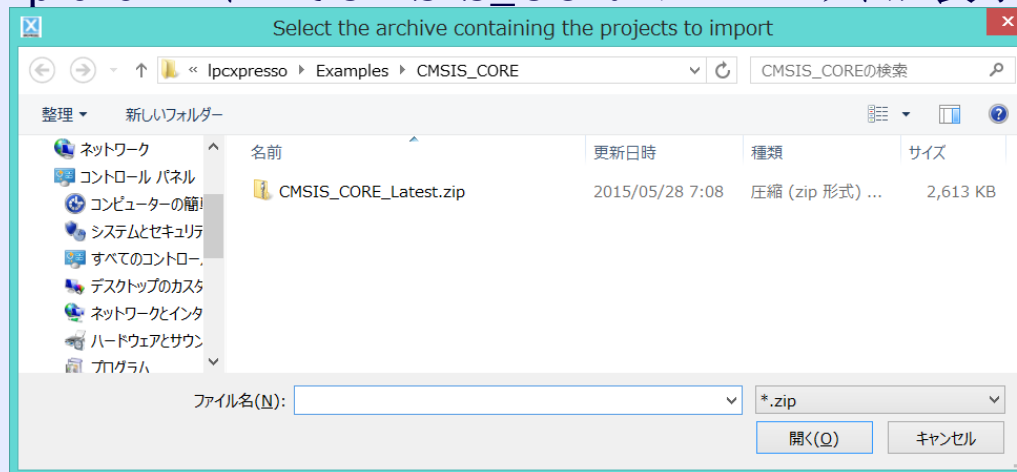


B)



4. 必要なライブラリのインポート

1. LPCXpressoを起動し、ワークスペースを設定する。ワークスペースは、全角文字やスペースが含まれていないパスに作成すること。
2. Start hereペイン中のImport project(s)をクリックする。
3. Project archive (zip)のBrowse...ボタンをクリックし、デフォルトで表示されるExampleフォルダからCMSIS_COREまたはLegacyを開き(下図を参照), CMSIS_CORE_latest.zipを選択する。開くボタンをクリック。
4. Next>ボタンをクリック。
5. 機種選択画面でCMSIS_CORE_LPC8xxのみにチェックを入れる。
6. Finishをクリック。
7. Project ExplorerペインでCMSIS_COREプロジェクトが表示されていればOK。



5. 新規プロジェクトの作成

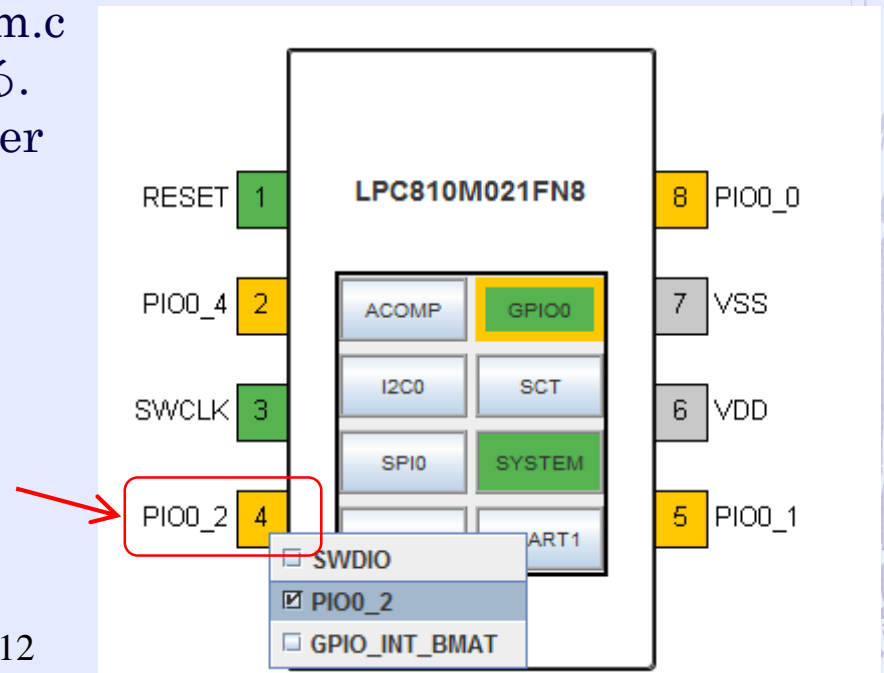
1. Start hereペイン中のNew project...をクリックする.
2. プロジェクトの種類としてLPC81xのC Projectを選択する.
3. プロジェクト名(半角英数字, たとえばLEDBlink)を入力する.
4. ターゲットMCUとしてLPC810を選択する.
5. CMSIS Core LibraryとしてCMSIS_CORE_LPC8xxを指定する.
6. Micro Trace Buffer Enableのチェックを外す.
7. Code Read Protect (CRP) のチェックを外す.
8. Finishをクリックし, ウィザードを完了する.
9. Project Explorerペインを使用して, 作成したプロジェクト中のcrp.cとmtb.cを削除する.
10. HEXファイルを生成するように設定を変更する. まず, 対象プロジェクトを選択した状態で, 上部メニューからProject->Propertiesを選択して, ダイアログ中のC/C++ BuildからSettingsを選択する. そしてBuild StepsページのPost-build stepsのコマンドを以下に修正する.

```
arm-none-eabi-size "${BuildArtifactFileName}"  
arm-none-eabi-objcopy -v -O ihex "${BuildArtifactFileName}" "${BuildArtifactFileName}.hex"
```

6. ピン設定コードの準備

LPC810のピン4にLEDを接続するためのピン設定コードをすでに用意してあるので、これを各自のプロジェクトにインポートする。なお、ピンの番号と名前は下図の通り。
(ピン4にPIO0_2という名前が付いていることに注目)

1. swm.cとtype.hを以下からダウンロードし、プロジェクトのsrcフォルダに保存する。
[¥¥stfile.eng.kagawa-u.ac.jp¥Material¥ENIE¥takagi¥emb_soft_eng](http://stfile.eng.kagawa-u.ac.jp/Material/ENIE/takagi/emb_soft_eng)
2. LPCXpressoでF5キーを押下してswm.cとtype.hをプロジェクトにインポートする。
インポートされたことをProject Explorerペインで確認する。



7. プログラムの作成

- ◆ ”プロジェクト名”.c というCソースファイルがsrcフォルダ中に生成されているので、LPCXpressoのエディタを使用して以下のように修正する。

```
#ifdef __USE_CMSIS
#include "LPC8xx.h"
#endif
```

```
#include <cr_section_macros.h>
#include "type.h"
```

```
void SwitchMatrix_Init();
```

```
int main(void) {
    SwitchMatrix_Init();
```

```
    // PIO0_2: output
```

```
    LPC_GPIO_PORT->DIR0 |= (1<<2);
```

```
    // Force the counter to be placed into memory
```

```
    volatile static int i = 0 ;
```

```
    // Enter an infinite loop, just incrementing a counter
```

```
    while(1) {
```

```
        for (i=0; i<1000000; i++);
```

```
        // Toggle PIO0_2
```

```
        LPC_GPIO_PORT->NOT0 = 1<<2;
```

```
    }
```

```
    return 0 ;
```

```
}
```

swm.cで定義される, ピン設定のための関数

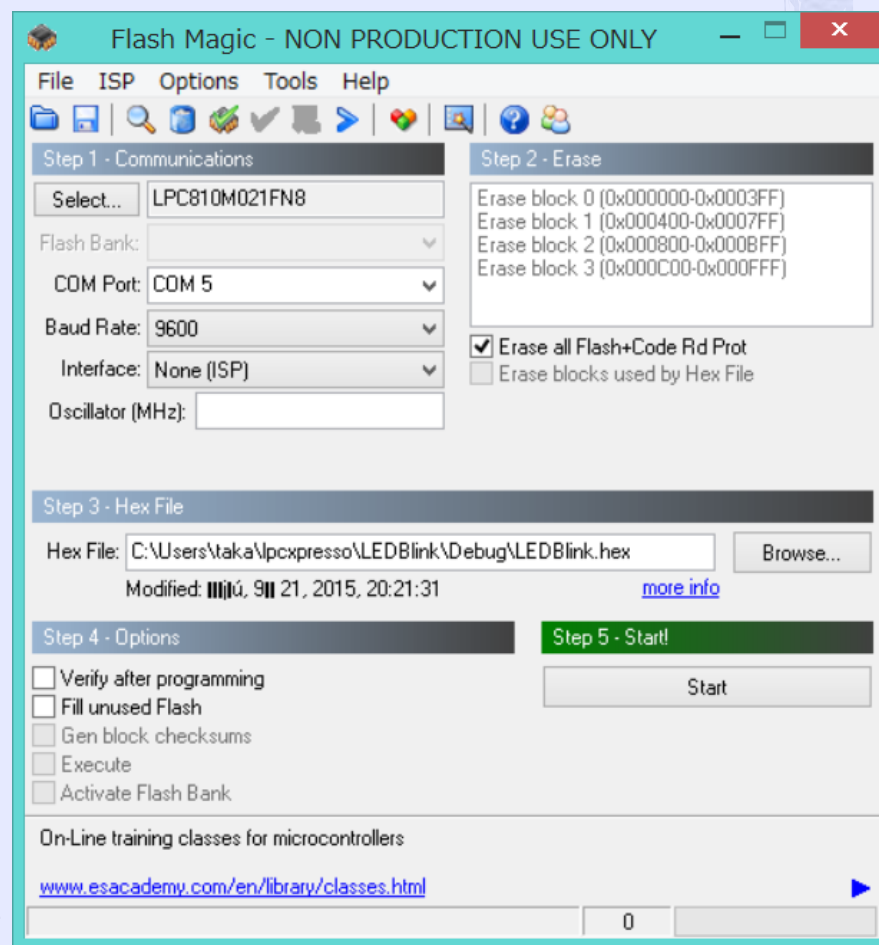
DIR0はPIO0の入出力を設定するレジスタ(n ビット目はPIO0_ n に対応. 1を2ビット左シフトしorをとる(つまりDIR0の2ビット目を1にする)ことでPIO0_2を出力モードに設定.

待ち

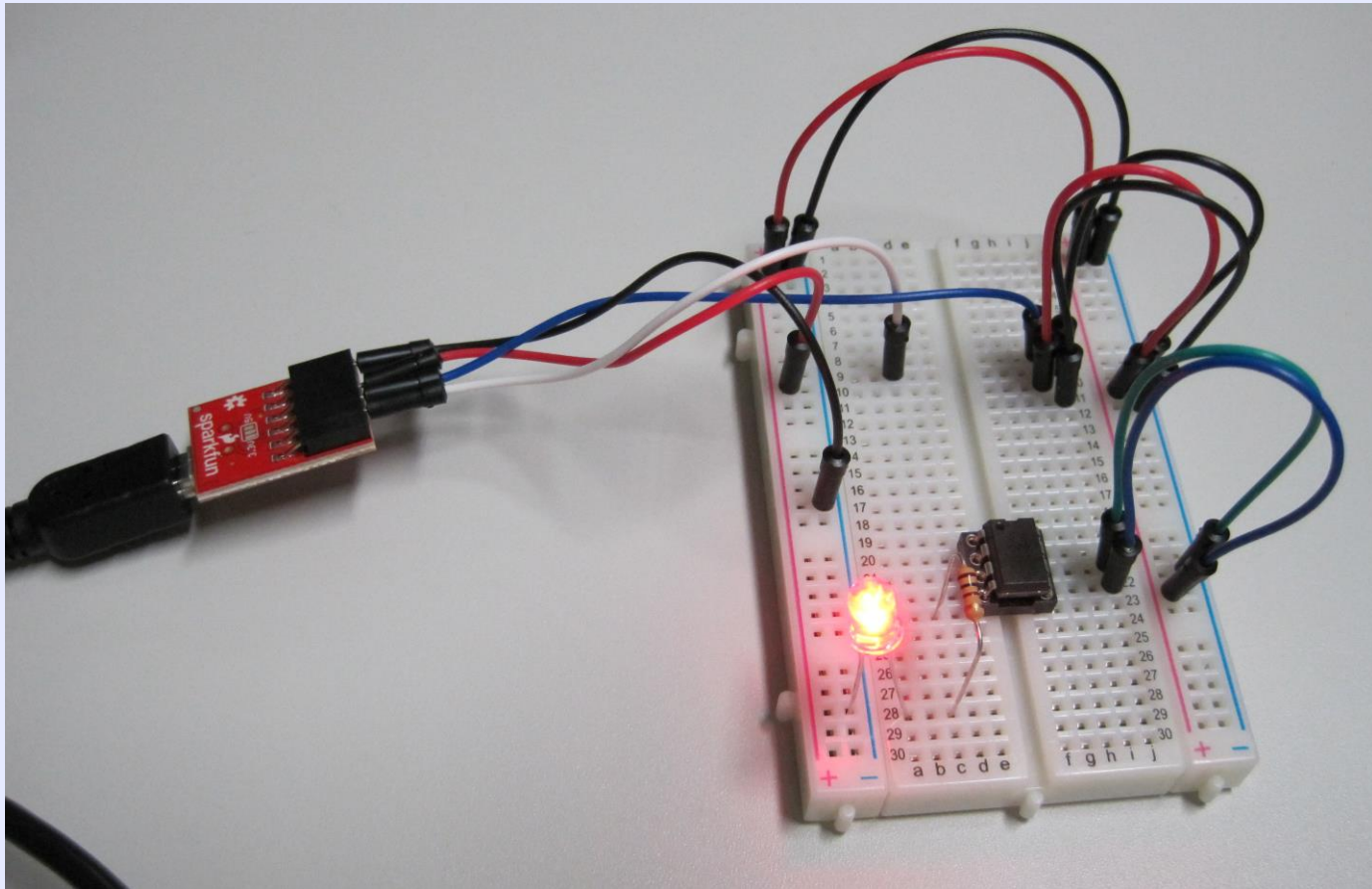
PIO0の指定したポート(PIO0_2)を反転

8. HEXファイルの生成, 書き込み, 実行

1. LPCXpressoにおいて, ビルド対象プロジェクトを選択した状態で, 上部メニュー Project->Build Projectをクリックしてビルドを実行. 成功すれば, Debugフォルダ内にHEXファイルが出力される.
2. PCとマイコンを接続する(図56). GNDを繋げるのを忘れないこと.
3. Flash Magicを起動し, 右図のようにパラメータを指定する. ただし, COM Port:は各自の環境に基づいて選択すること(デバイスマネージャで確認). Hex File:には, 先ほど生成したHEXファイルを指定する.
4. Startをクリックする. ステータスバーに Finishedと表示されればOK.
5. 動作チェックを行う. GNDを外してマイコンを挿し替える(図57). そしてGNDを再度つなぎ, LEDが点滅すればOK.

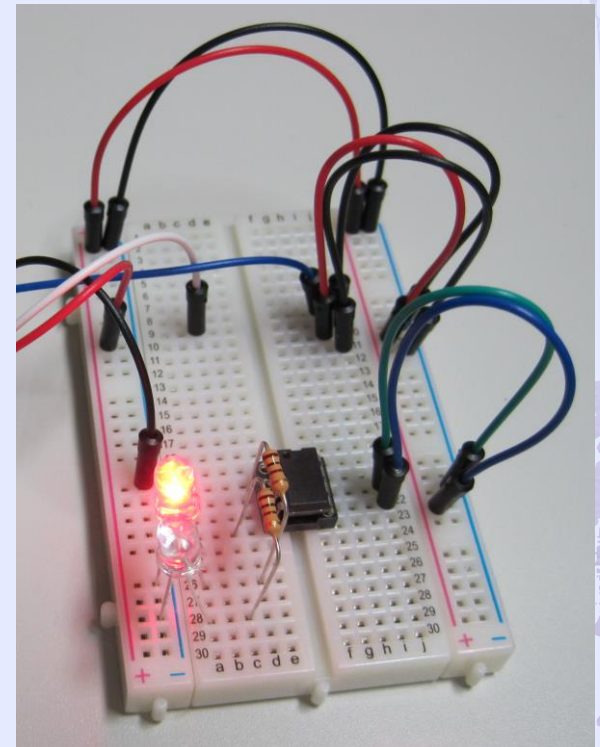
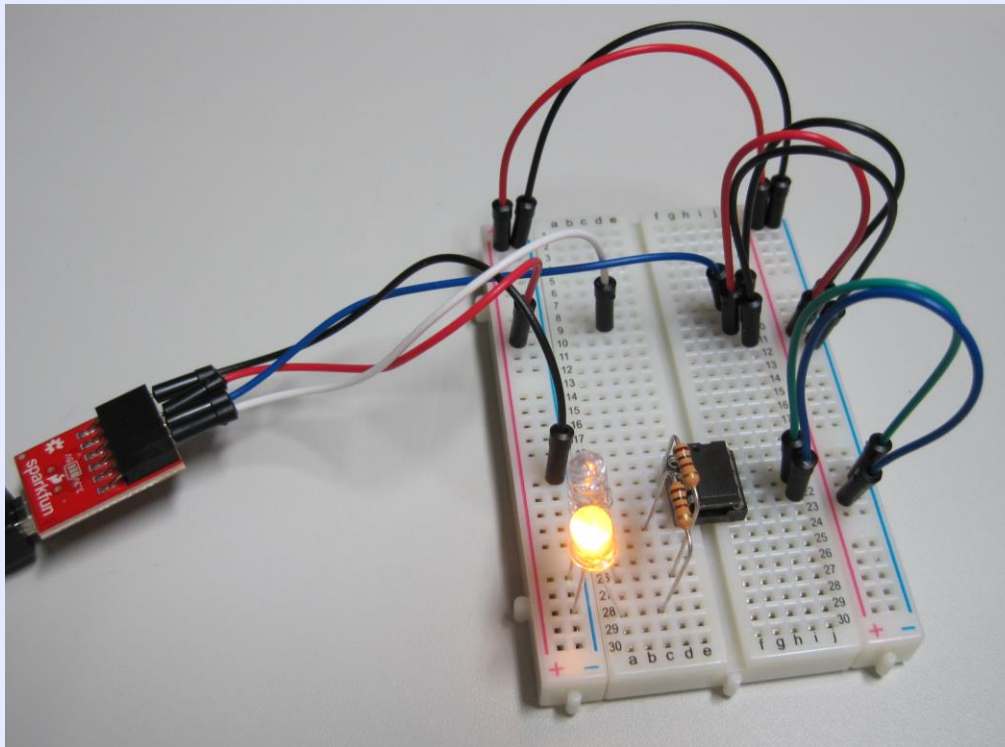


8. HEXファイルの生成, 書き込み, 実行



9. 演習

- ◆ 2人一組で協力して、歩行者用信号機の点灯パターンを実装せよ. 青信号は黄色のLEDで代用する. 図57を参考に, ピン4に黄色のLEDを, ピン2に赤色のLEDを接続すること(無論, 抵抗を入れるのを忘れないこと. なお, ピン2はデフォルトでGPIOの機能がアサインされている(PIO0_4)ので, ピン設定コードを作り直す必要はない.



9. 演習(つづき)

- ◆ レポートの書式： ページのサイズはA4とする. 1ページ目にタイトル, 提出日, 学籍番号, 氏名を記すこと. (1) 問題文, (2) プログラムのソースコードとその説明文, (3) 配線や動作が分かる写真(と可能なら動画), (4) 感想が必要である. 説明文は, 他の受講生が読んで理解できるレベルの文章を記述すること.
- ◆ 提出先 : `¥¥stfile.eng.kagawa-u.ac.jp¥Report¥ENIE¥takagi¥emb_soft_eng¥200213` に各自の学籍番号名のpdfファイル(と動画ファイル)をアップロードすること.
- ◆ 提出期限 : 2月16日 23:59 (厳守)
- ◆ その他 : 解答の丸写しが発覚した場合は単位を認定しない.
自前で別の開発環境および電子部品(たとえば, PICやArduinoなど)を用意して, 同等のものを作成しても良い.