# PSoC® 4 CY8CKIT-049 4xxx Prototyping Kit 使い方

日本サイプレス株式会社

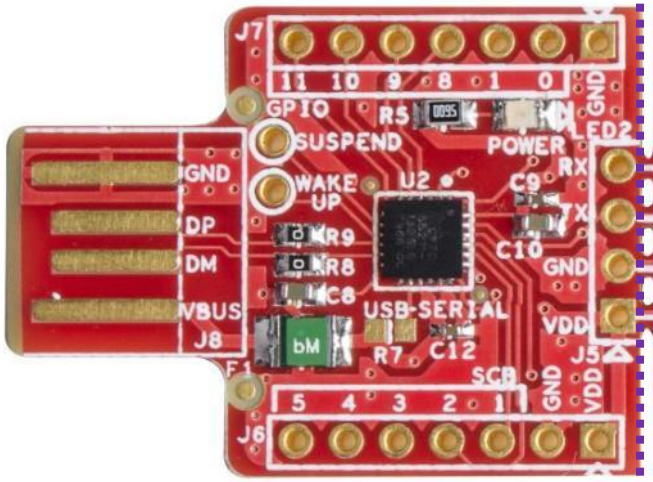
2014/04/07

# この資料の目的

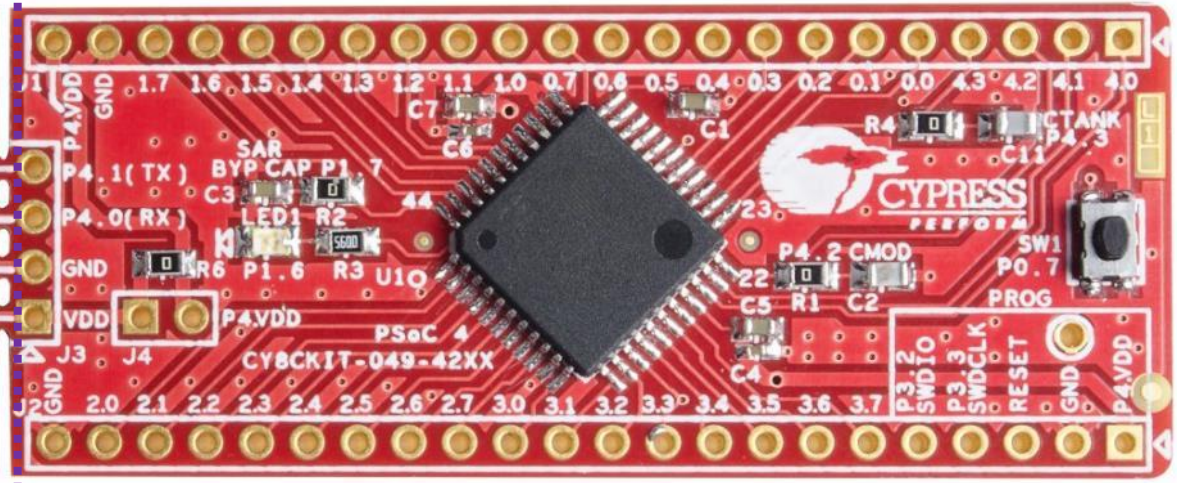
この資料はPSoC 4 Prototyping Kit Guide (001-90711) (http://www.cypress.com/?docID=48142) の抜粋です。
詳細は原文を参照してください。

このキットに関連する情報は以下のURLよりダウンロード出来ます。
http://japan.cypress.com/?rid=92146

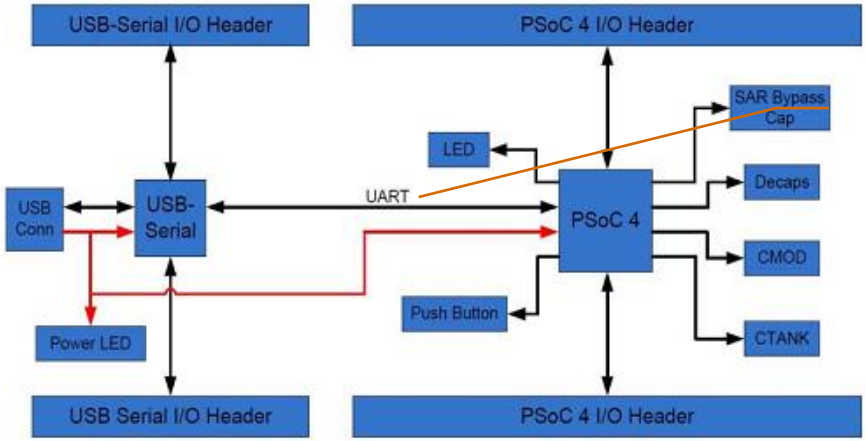
# CY8CKIT-049外観




USB シリアル(RS-232C) 変換




PSoC 4 評価ボード




二枚の基板はUARTで接続されています。
通常のPSoCでは書き換えはSWDプロトコルを用います。
このキットではBootloaderという仕組みを利用することで、UART経由での書き込みをサポートしています。

# 開発環境のセットアップ

# インストールが必要なツール

## PSoC Creator™ Ver. 3.0以降

PSoCの開発に使用します。

http://www.cypress.com/go/psoccreator

## USB-Serial Software Development Kit

これをインストールすることで、USBシリアル変換ICのドライバがPCにインストールされます。

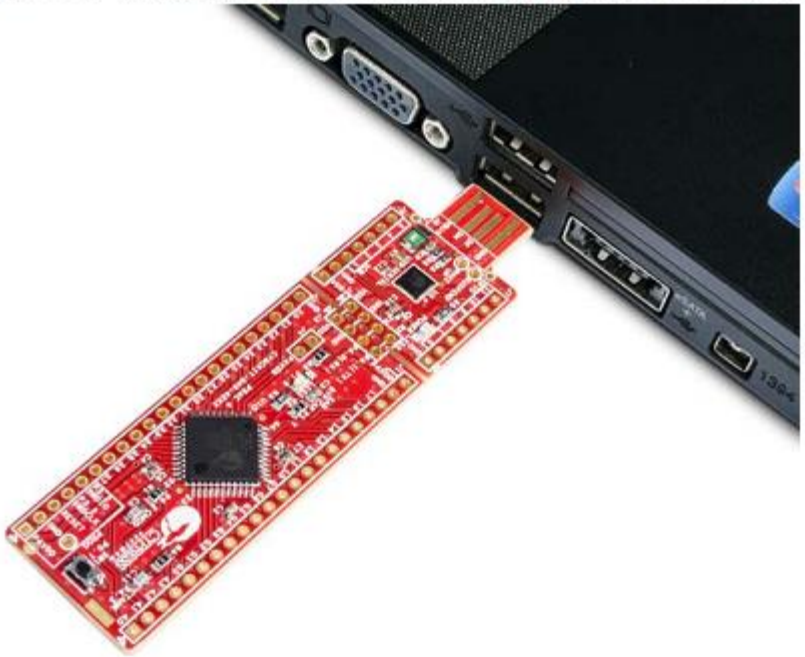
通常の、MiniProg3を用いたPSoC開発には、このソフトウェアは不要です。

http://www.cypress.com/?rID=83110

# H/W Setup

1．以下の向きでPCのUSBポートに基板を接続

Figure 3-2. Connecting the PSoC 4 Prototyping Kit to a Computer



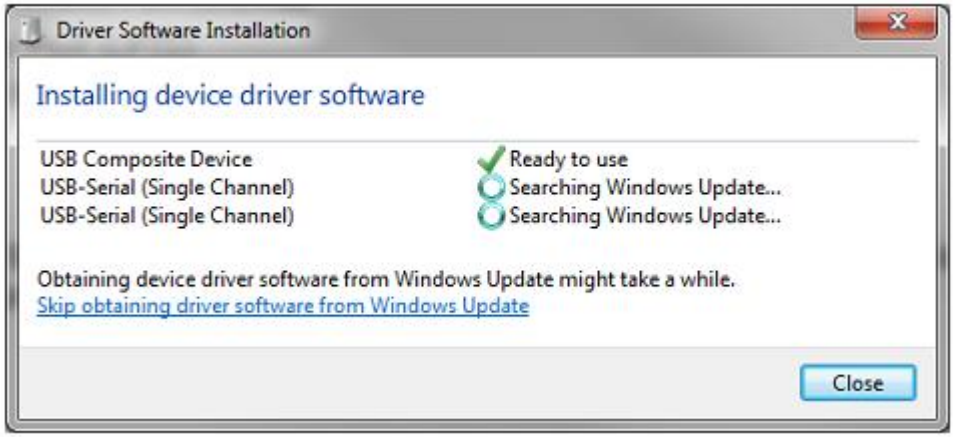
2．基板上の赤LEDが点灯、青LEDが点滅します

Figure 3-3. PSoC 4 Prototyping Kit Connected to the Computer

# USB シリアル変換チップドライバ

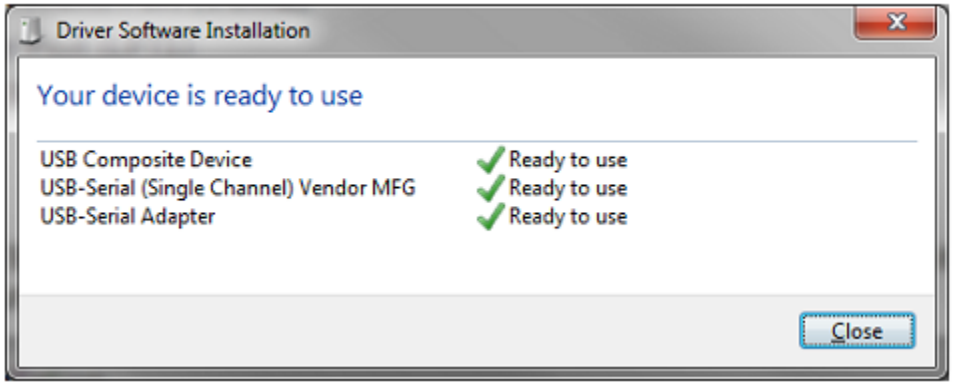
3. ドライバのインストールは自動的に開始されます

Figure 3-5. Automatic Driver Software Installation for CY8CKIT-049-4xxx




4. ドライバのインストール完了しますと以下のようになります

Figure 3-6. Driver Software Installation Complete

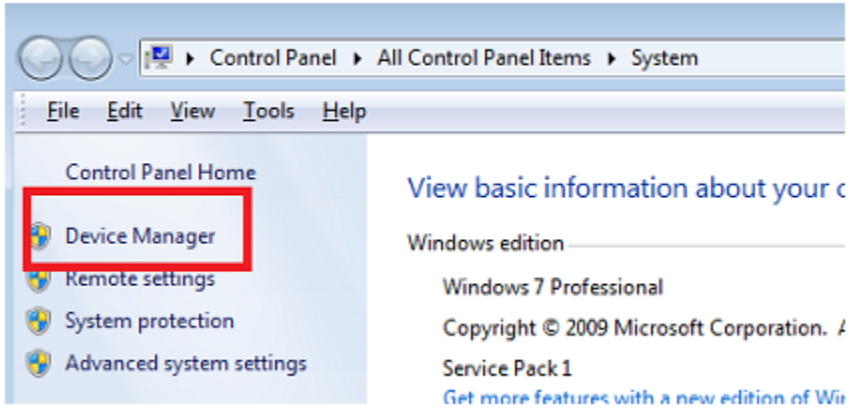


# COMポート番号の変更

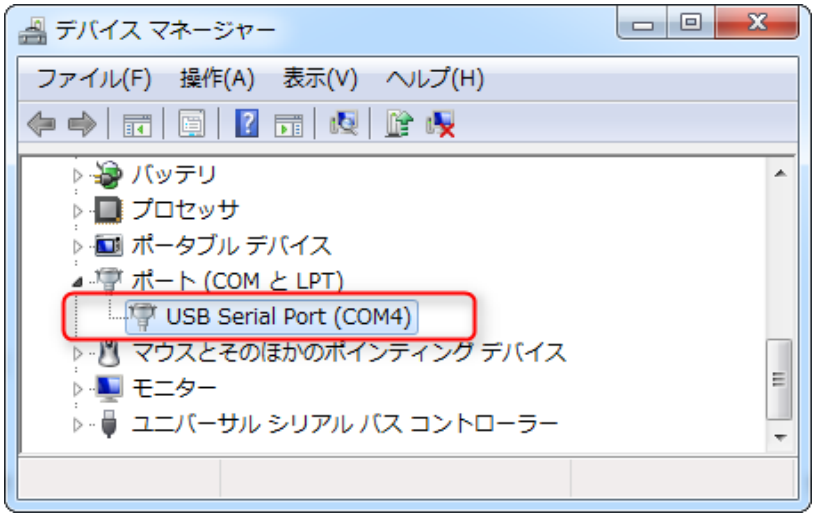
CY8CKIT-049において、USBシリアル変換チップに割り当てられるCOMポート番号は9以下である必要があります。これは、Bootloader Hostツールの制限によります。以下のステップに従って、割り当てられたCOMポート番号を確認し、10以上の場合は9以下となるように変更して下さい。

5. デバイスマネージャを開く
コントロールパネル > システム > デバイスマネージャ

Figure 3-7. Windows Device Manager

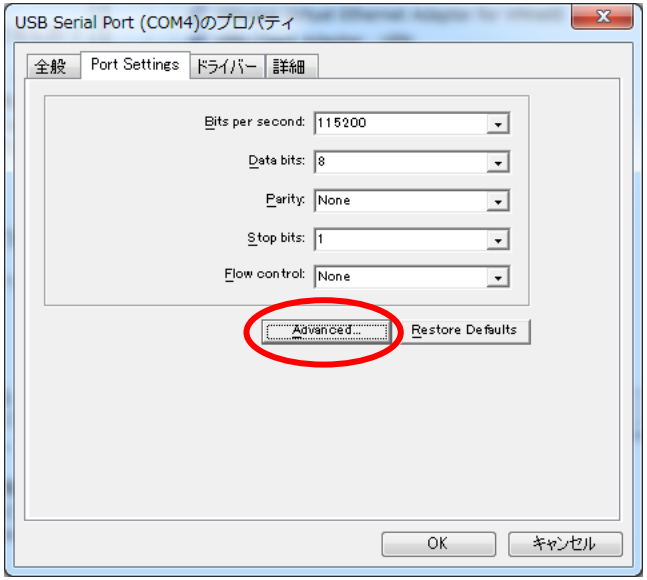



6. デバイスマネージャのポート(COMとLPT)でUSB Serial Port ( COMxx)を探す

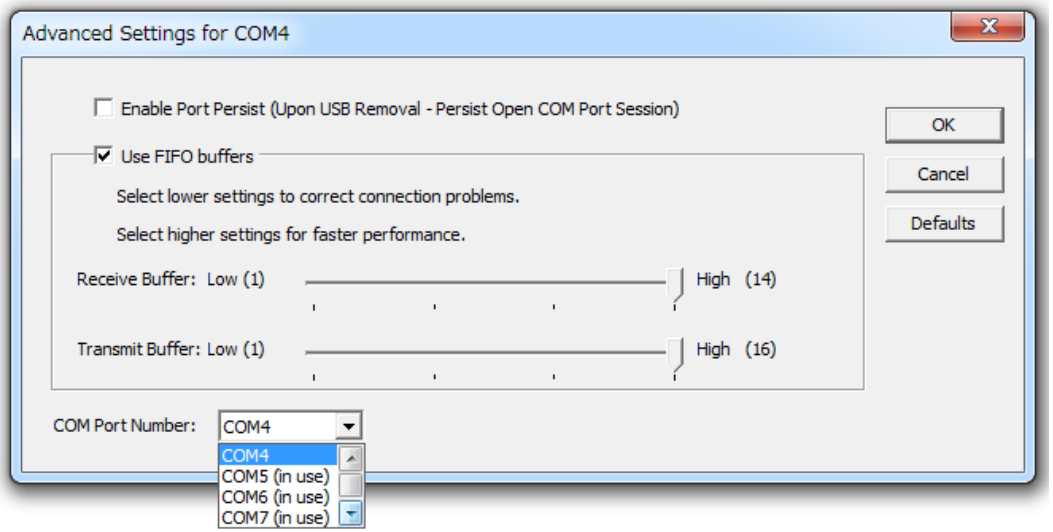


# COMポート番号の変更

7. COMポートのプロパティーの Port Settingsの
Advancedボタンをクリック




8. Com Port Numberが10以上の値の場合は、9
以下の値に設定して、OKをクリック



# キットの動作確認

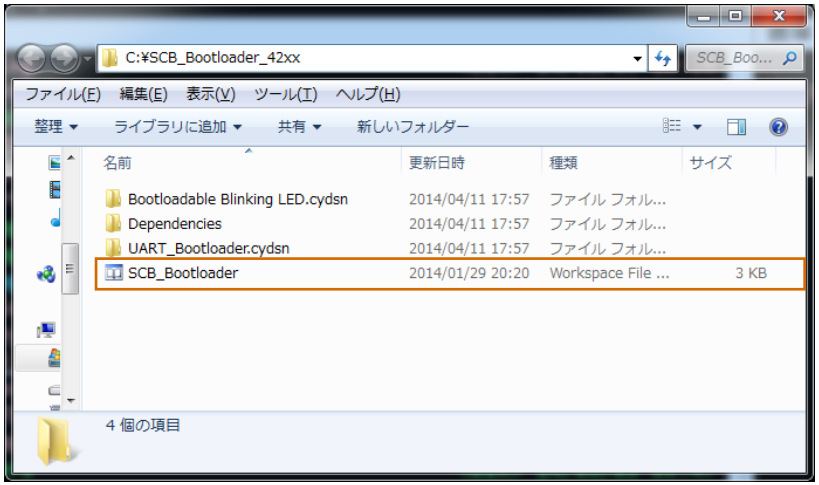
# サンプルプロジェクトのダウンロード

http://www.cypress.com/?docID=47827

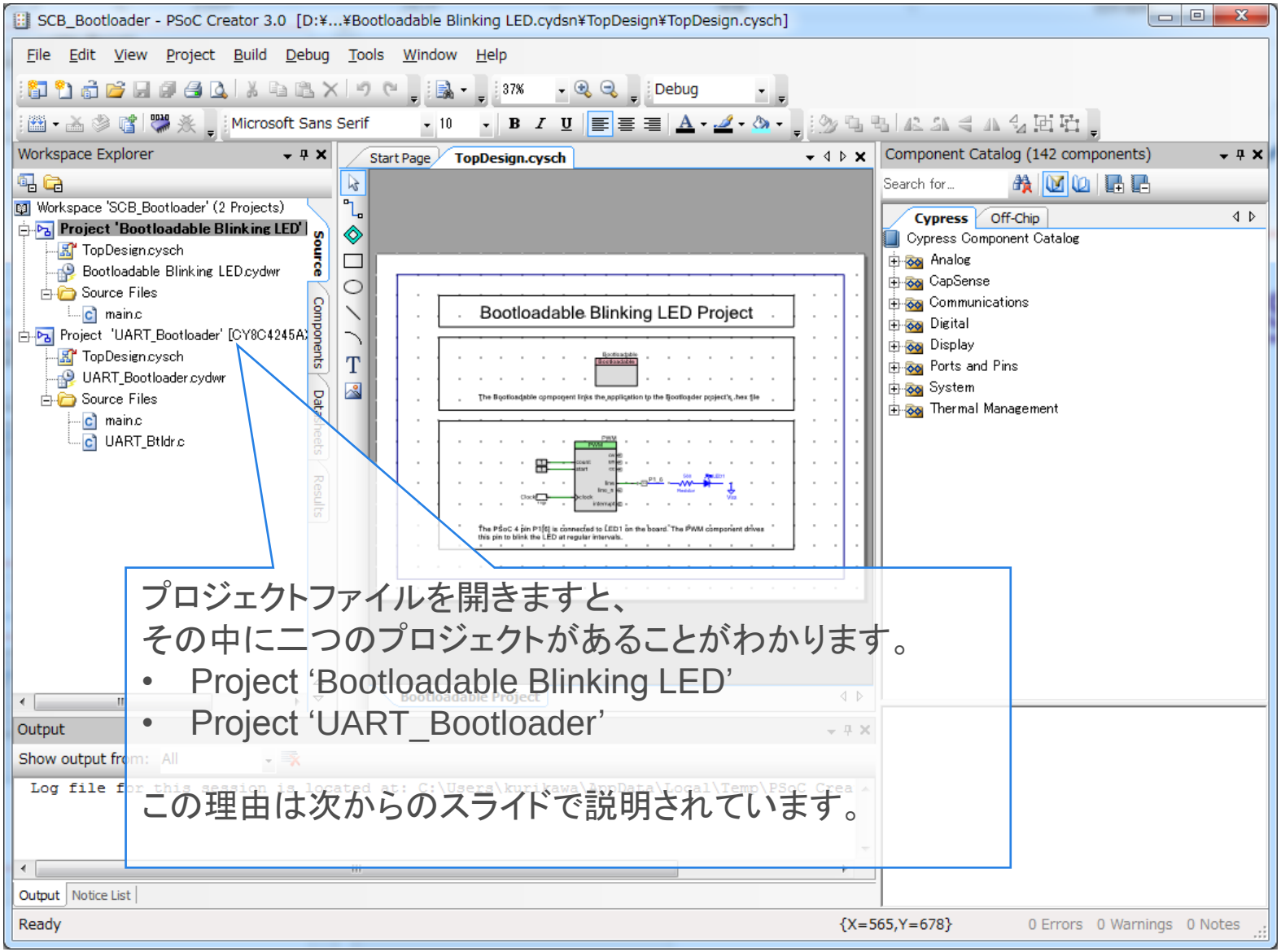
ここからダウンロード出来るZIPファイルを任意のフォルダ(2バイト文字を含まないファイルパス)に展開してください。

この資料では、以下の図のようにCドライブ直下に配置しています。

SCB_Bootloader.cywrk を開くと、PSoC Creatorが起動します。



# プロジェクト ファイルの構造




プロジェクトファイルを開きますと、
その中に二つのプロジェクトがあることがわかります。

- Project ‘Bootloadable Blinking LED’
- Project ‘UART_Bootloader’

この理由は次からのスライドで説明されています。

# PSoCへの書き込み手段 SWD

Flash ROM全体

ユーザーが定義する
アプリケーション

SWDプロトコル

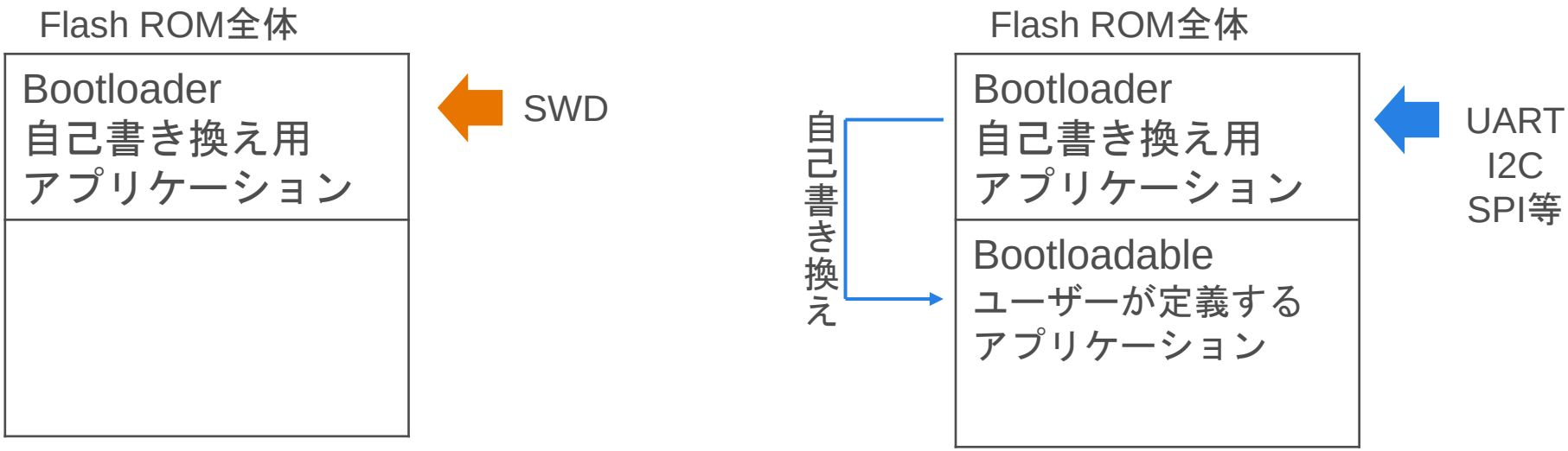
**通常の、PSoCのデザインでは、PSoC Creatorを使用して設計を行い、生成されたHEXファイルをSWDプロトコルでデバイスに書き込みます。**

**この方法では書き換えのたびにSWDホストが必要になります。**

**CypressはUSB-SWDブリッジとして**

**MiniProg3 ( CY8CKIT-002 ) を提供しています。**

# PSoCへの書き込み手段 SWD+Bootloader

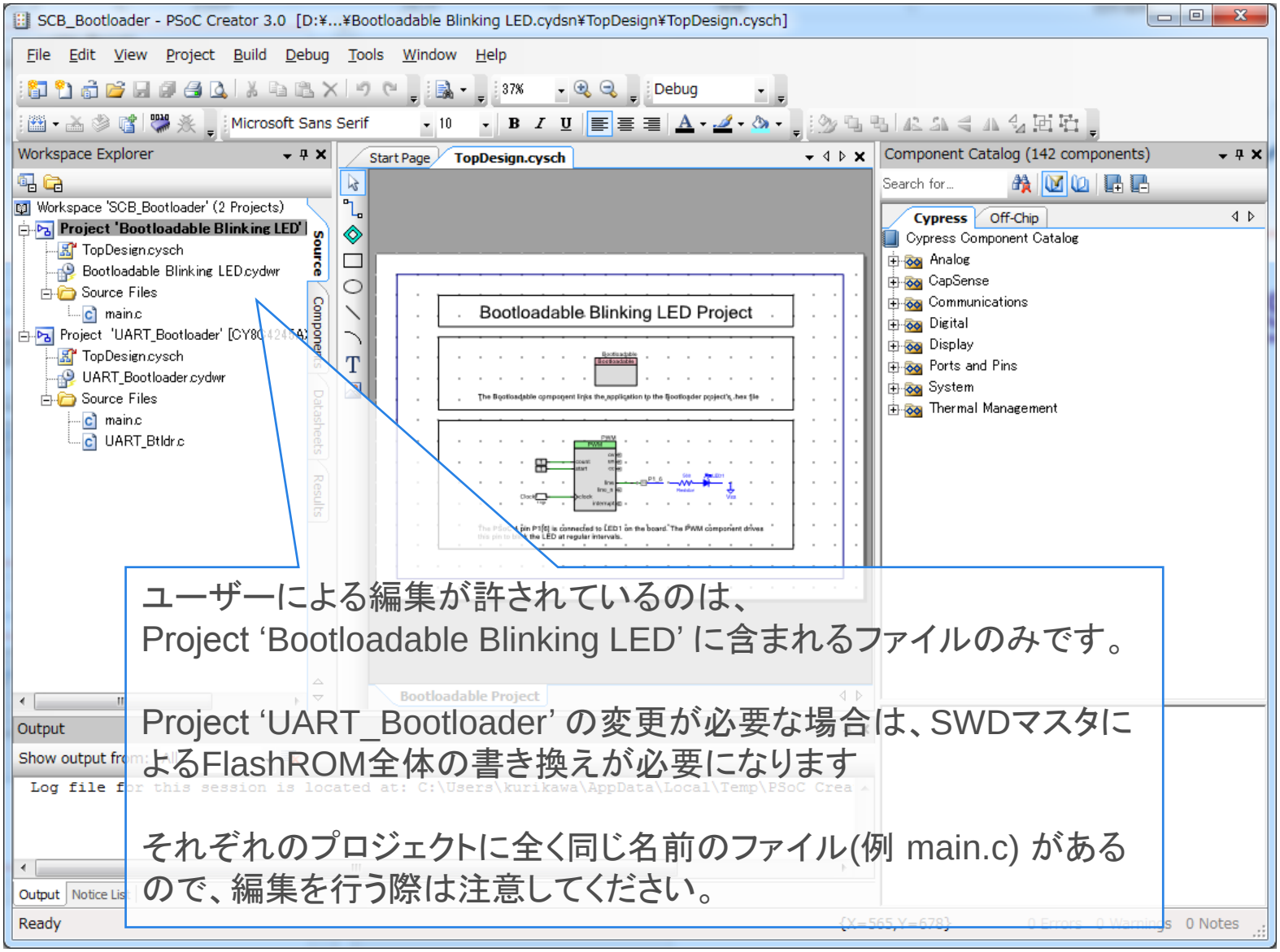



**Bootloaderを使用したデザインでは、PSoC Creatorを使用して設計を行い、生成されたBootloder HEXファイルをSWDプロトコルでデバイスに書き込みます。**

**一度Bootloaderを書き込んだ後は、Bootloaderに組み込まれた汎用プロトコル(例えば UART, I2C, SPIなど) で、Flash ROMの残りの領域 (Bootloadable領域) の書き換えが可能になります。**

**このキットに搭載されているPSoC 4には既にUART Bootloaderが書き込まれているため、UART経由で書き換えが可能です。**

# プロジェクト ファイルの構造

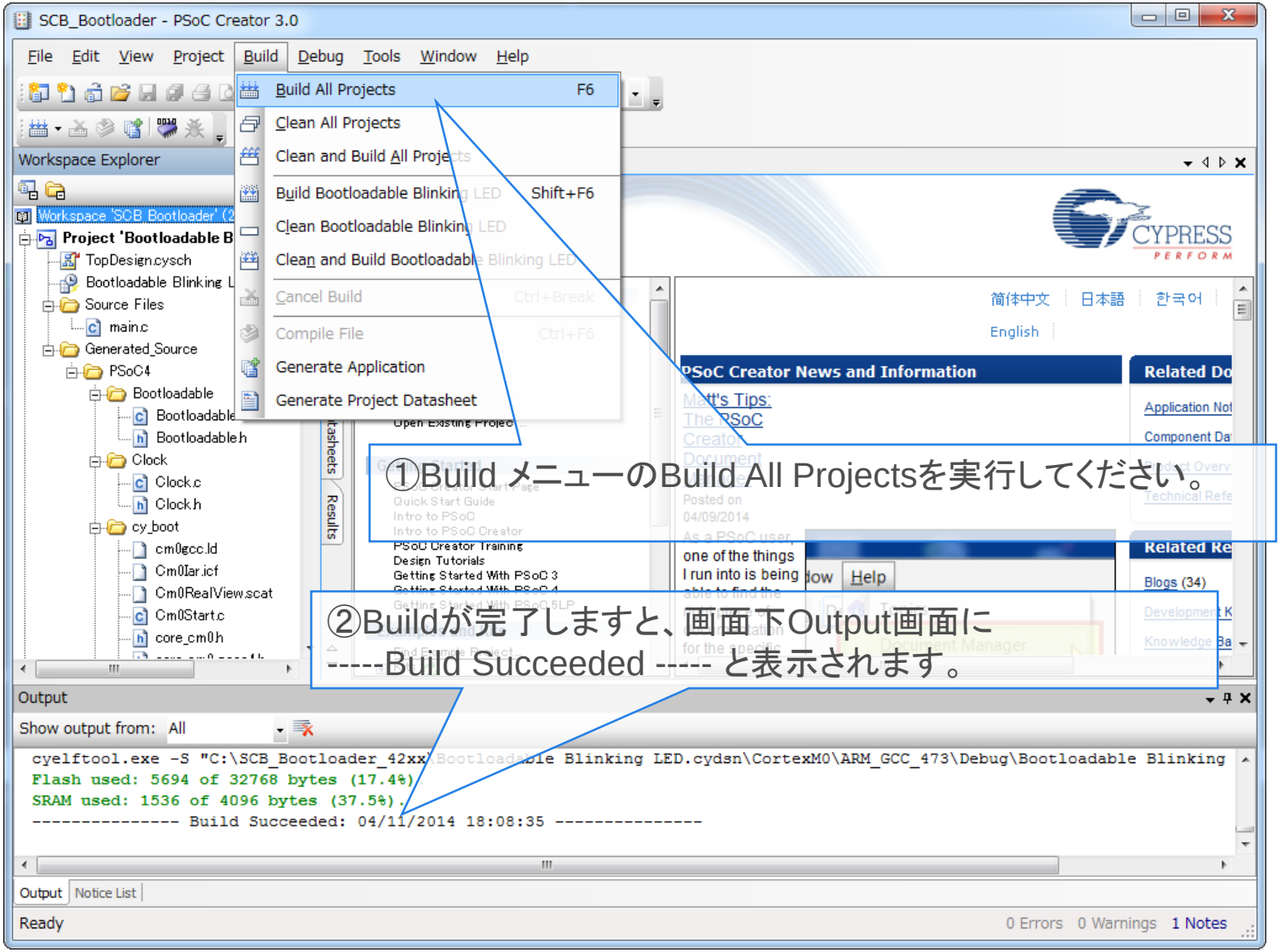



ユーザーによる編集が許されているのは、
Project ‘Bootloadable Blinking LED’ に含まれるファイルのみです。

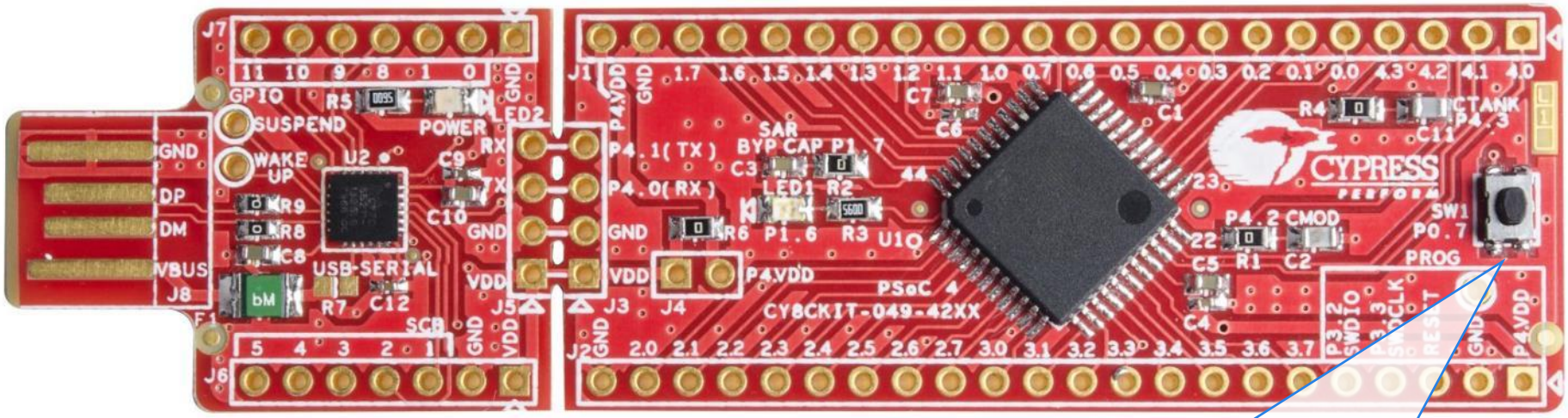
Project ‘UART_Bootloader’ の変更が必要な場合は、SWDマスタによるFlashROM全体の書き換えが必要になります

それぞれのプロジェクトに全く同じ名前のファイル(例 main.c) があるので、編集を行う際は注意してください。

# 動作確認


SCB_Bootloader - PSoC Creator 3.0
File Edit View Project Build Debug Tools Window Help
Build All Projects F6
Clean All Projects
Clean and Build All Projects
Build Bootloadable Blinking LED Shift+F6
Clean Bootloadable Blinking LED
Clean and Build Bootloadable Blinking LED
Cancel Build Ctrl+Break
Compile File Ctrl+F6
Generate Application
Generate Project Datasheet
Workspace Explorer
Workspace 'SCB_Bootloader' (2
Project 'Bootloadable B
TopDesign.cysch
Bootloadable Blinking L
Source Files
main.c
Generated_Source
PSoC4
Bootloadable
Bootloadable.h
Clock
Clock.c
Clock.h
cy_boot
cm0gcc.ld
Cm0Iar.icf
Cm0RealView.scat
Cm0Start.c
core_cm0.h
Output
Show output from: All
cyelftool.exe -S "C:\SCB_Bootloader_42xx\Bootloadable Blinking LED.cydsn\CortexM0\ARM_GCC_473\Debug\Bootloadable Blinking
Flash used: 5694 of 32768 bytes (17.4%).
SRAM used: 1536 of 4096 bytes (37.5%).
---------- Build Succeeded: 04/11/2014 18:08:35 ----------
Output Notice List
Ready
0 Errors 0 Warnings 1 Notes
CYPRESS PERFORM
简体中文 日本語 한국어
English
PSoC Creator News and Information
Related Do
Related Re
Blogs (34)
Help
①Build メニューのBuild All Projectsを実行してください。
②Buildが完了しますと、画面下Output画面に
-----Build Succeeded ----- と表示されます。

# Bootloaderモードへの切り替え




Flash ROM全体

| |
|---|
| Bootloader<br>自己書き換え用App |
| Bootloadable<br>ユーザーが定義するApp |

SW1を押しながら、PCのUSBポートに差し込んでください。

電源投入時に、SW1が押されていると、Bootloader ( 自己書き換え用App )が起動します。

電源投入時に、SW1が押されていないと、Bootloadable ( ユーザーが定義するApp )が起動します。

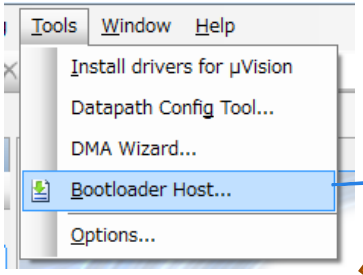
# Bootloaderモードで起動したことを確認

青いLEDの点滅周期が早くなったら、PSoCはBootloderモードで起動しています。

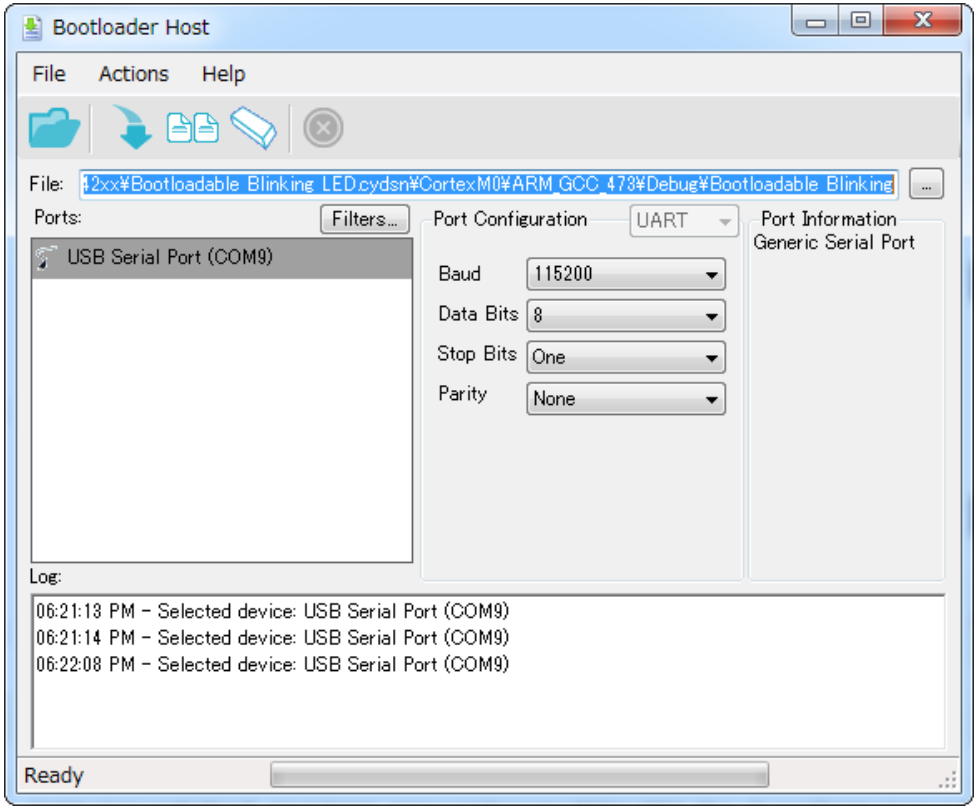
電源投入時に、SW1が押されていると、Bootloader ( 自己書き換え用App )が起動します。このとき、青色LEDは1秒間に2回の頻度で点滅します。

電源投入時に、SW1が押されていないと、Bootloadable ( ユーザーが定義するApp )が起動します。
このとき、青色LEDはBootloadable内で定義された周期で点滅します。キット開封時は1秒間に1回の頻度で点滅します。

# Bootloader Host Toolの起動




Toolsメニューの Bootloader Host を選択してください。
Bootloader Hostツールが起動します。



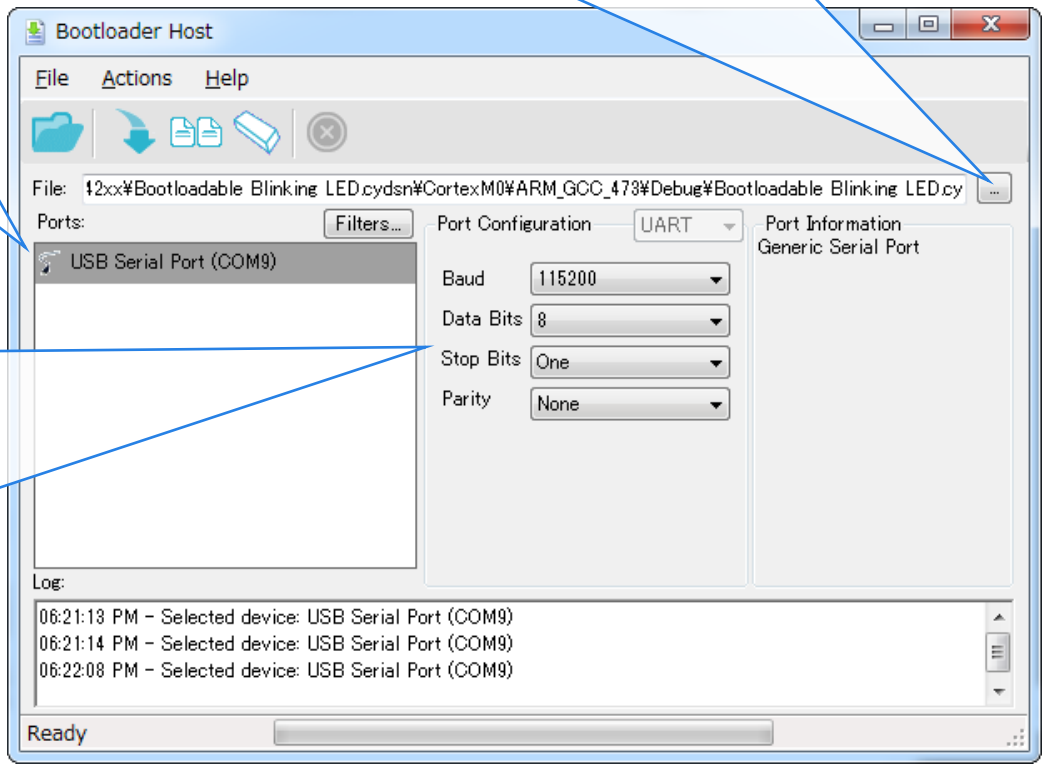
# Bootloader Host ツールの設定

① File: ボックスの右側をクリックして、以下のファイルパスを選択してください。
C:¥SCB_Bootloader_42xx¥Bootloadable Blinking LED.cydsn¥CortexM0¥ARM_GCC_473¥Debug¥Bootloadable Blinking LED.cyacd

② Ports:でUSB Serial Port(COM*)を選択してください。

③ Port Configuration は以下のように設定してください。
Baud : 115200
Data Bits : 8
Stop Bits : One
Parity : None

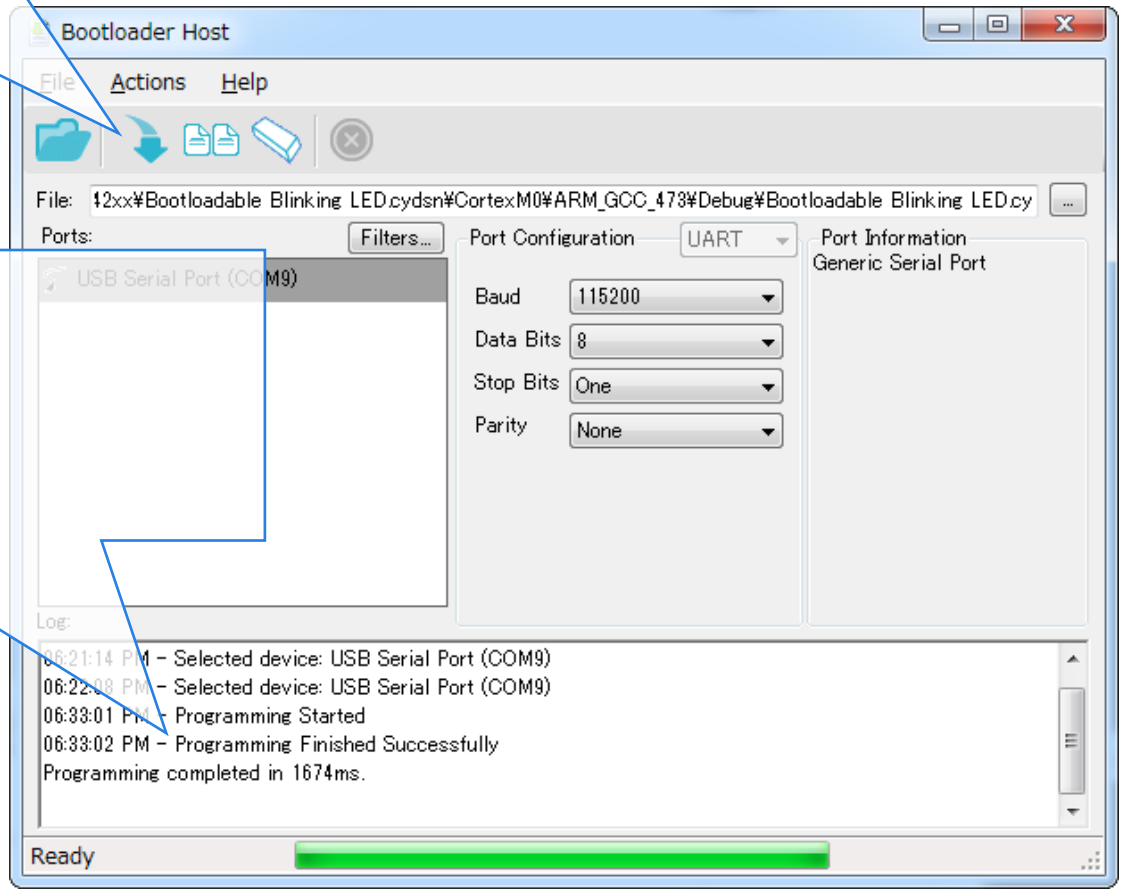
Bootloader Host ツールはポート番号が9以下のCOMポートのみ利用出来ます。



# デバイスへの書き込み

①プログラムボタンをクリックするか、
Actions メニューのProgramを選択しますと、
デバイスへのプログラムが開始されます。

②書き込みに成功しますと、
Log:ボックスに
Programming Finished Successfully
Programming Completed
と表示されます。



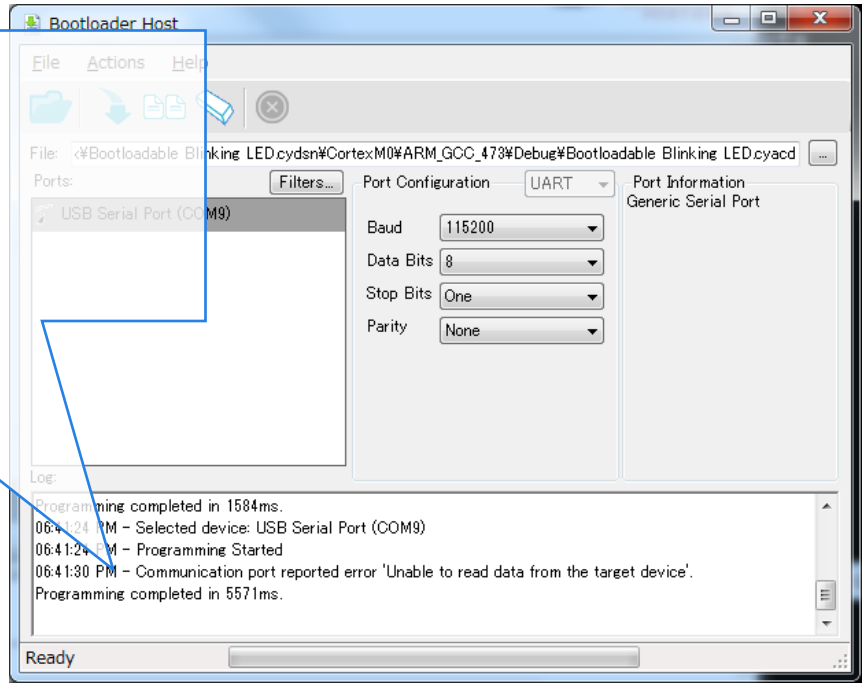
# 正しく書き換えられた場合

Bootloader 経由での書き換えが完了しますと、デバイスはBootloadable ( ユーザーが定義するApp ) に自動的に切り替わります。

青いLEDの点滅周期が遅くなったら(1秒間に1回の頻度)、PSoCはBootloder モードで起動しています。

# 書き込みに失敗した場合

書き込みに失敗しますと、
Log:ボックスに
エラーメッセージ
Programming Completed
と表示されます。

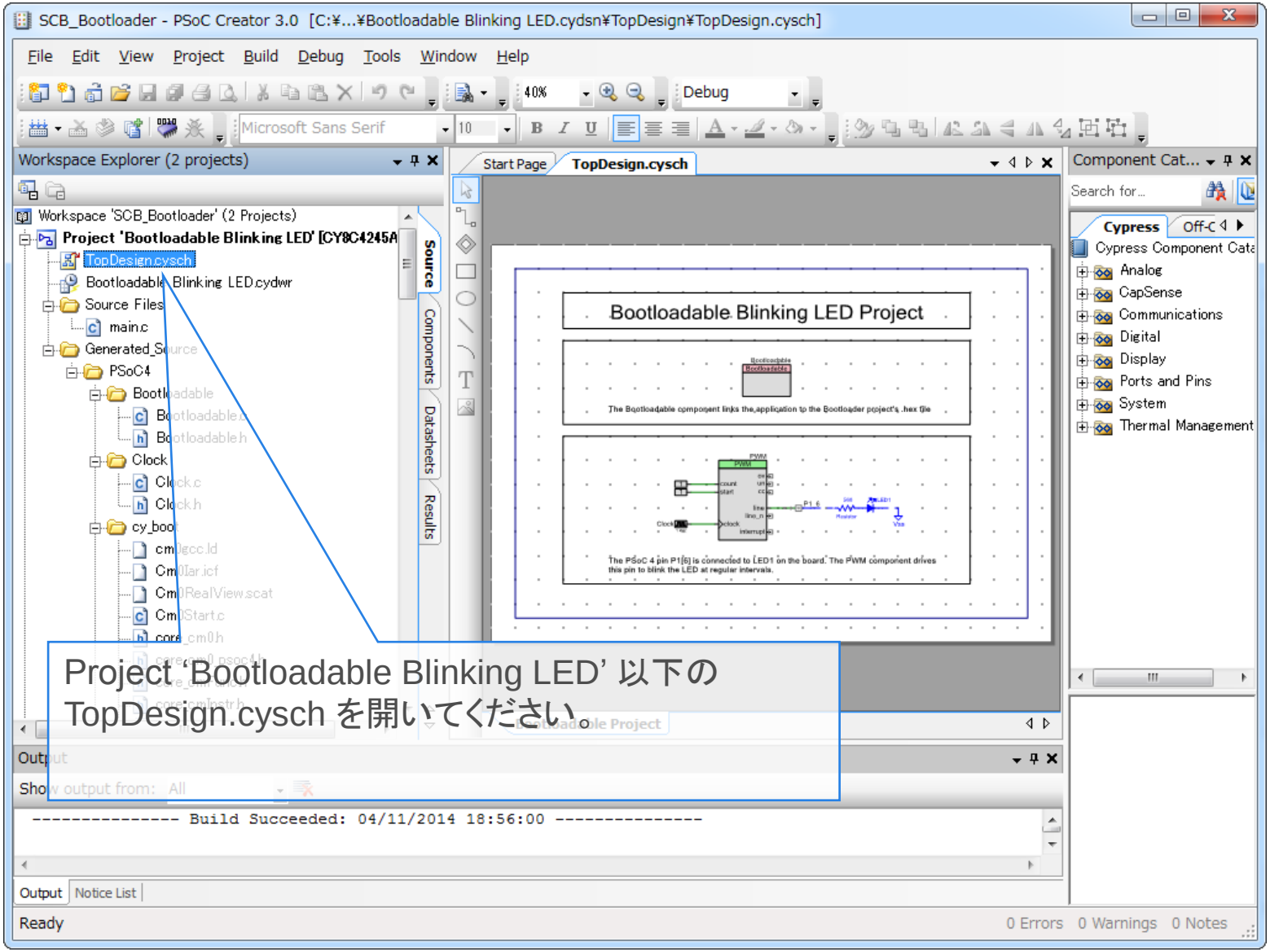



最も頻繁におきる失敗の理由は、PC接続時にSW1を押し忘れたことにより、
「デバイスがBootloader Modeで起動していない」です。
このとき、上図のようなエラーメッセージが出力されます。

**(注意) プログラムボタンを押した後は、**
**Programming Completed と表示されるまでは、**
**絶対にKITをUSBポートから抜かないでください！**

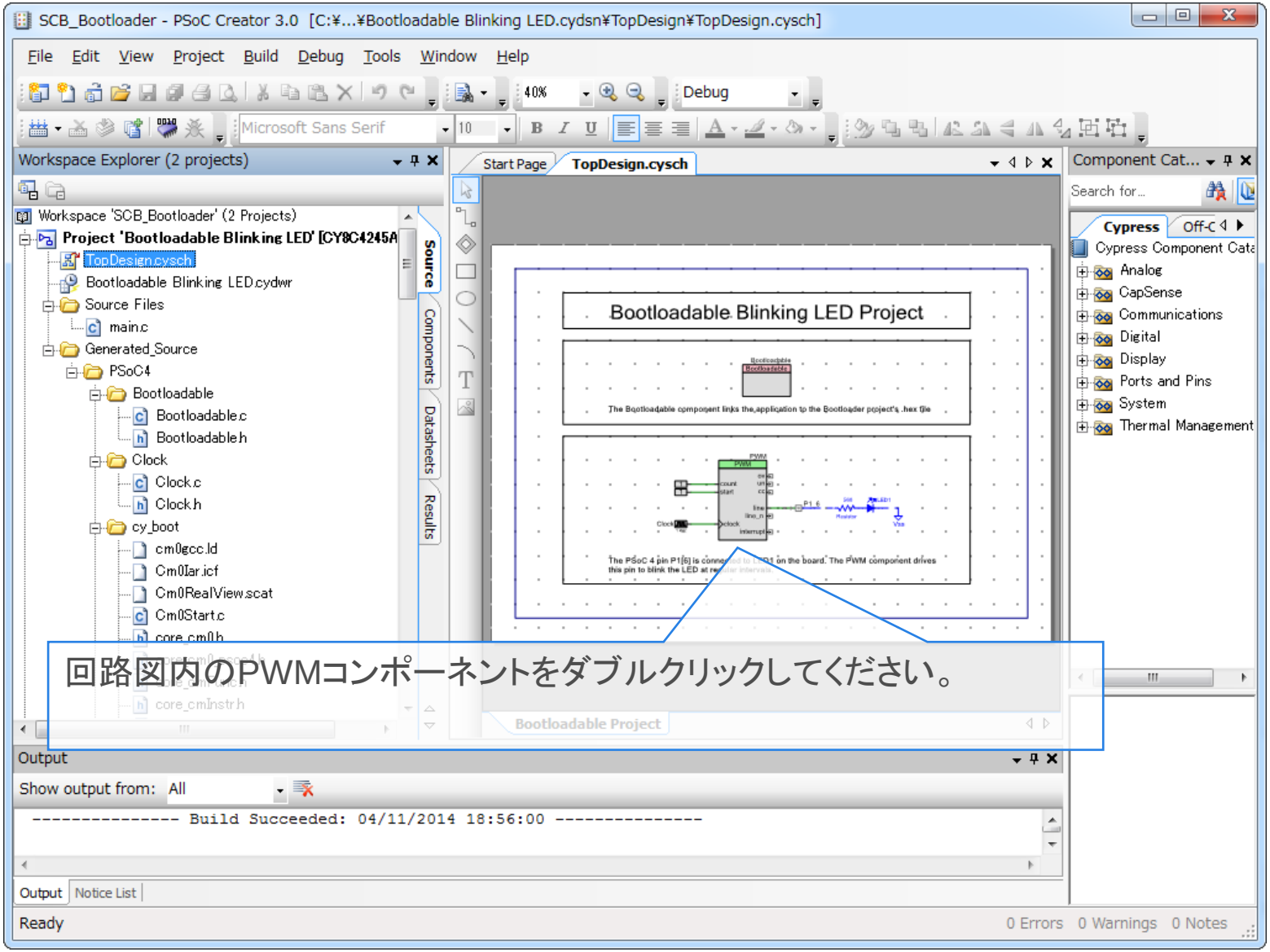
# LAB: PWMのデューティ比の変更

# ①回路図を開く




Project ‘Bootloadable Blinking LED’ 以下の
TopDesign.cysch を開いてください。

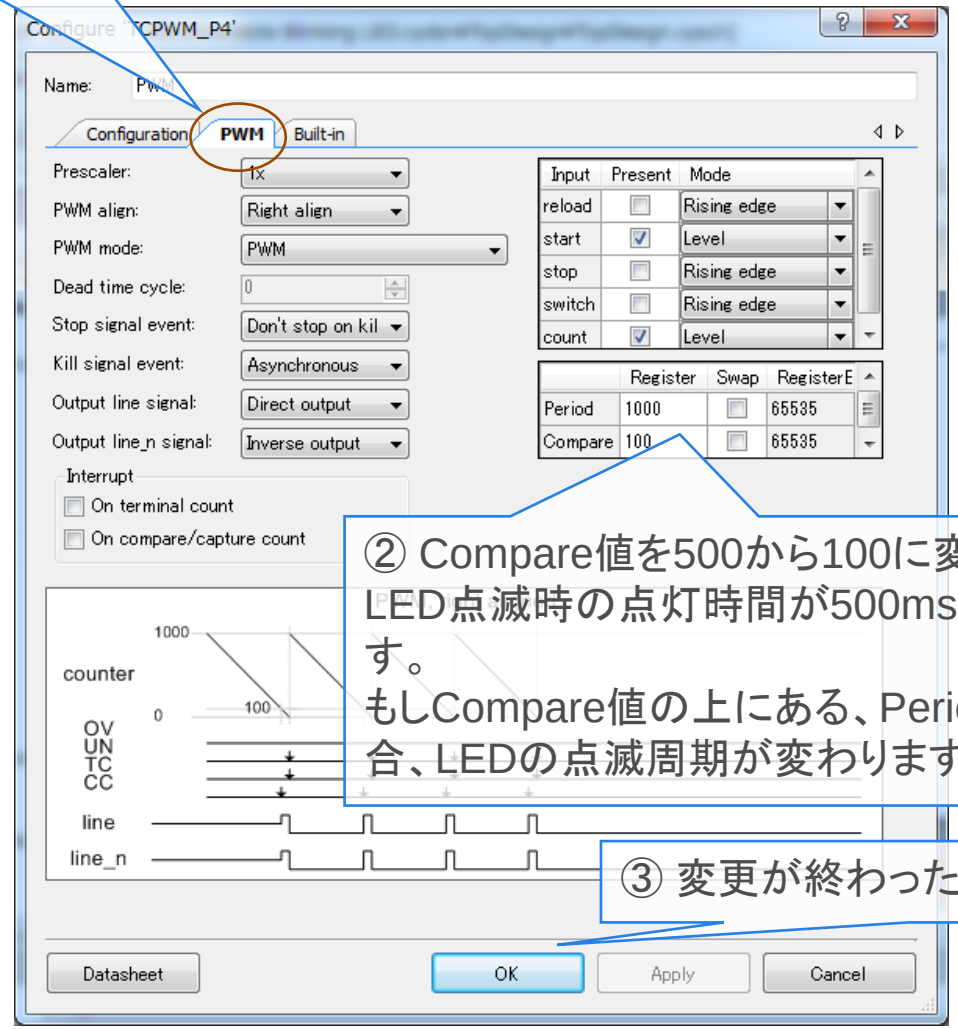
# ②PWM Configuration画面を開く




回路図内のPWMコンポーネントをダブルクリックしてください。

# ③ デューティ比の変更

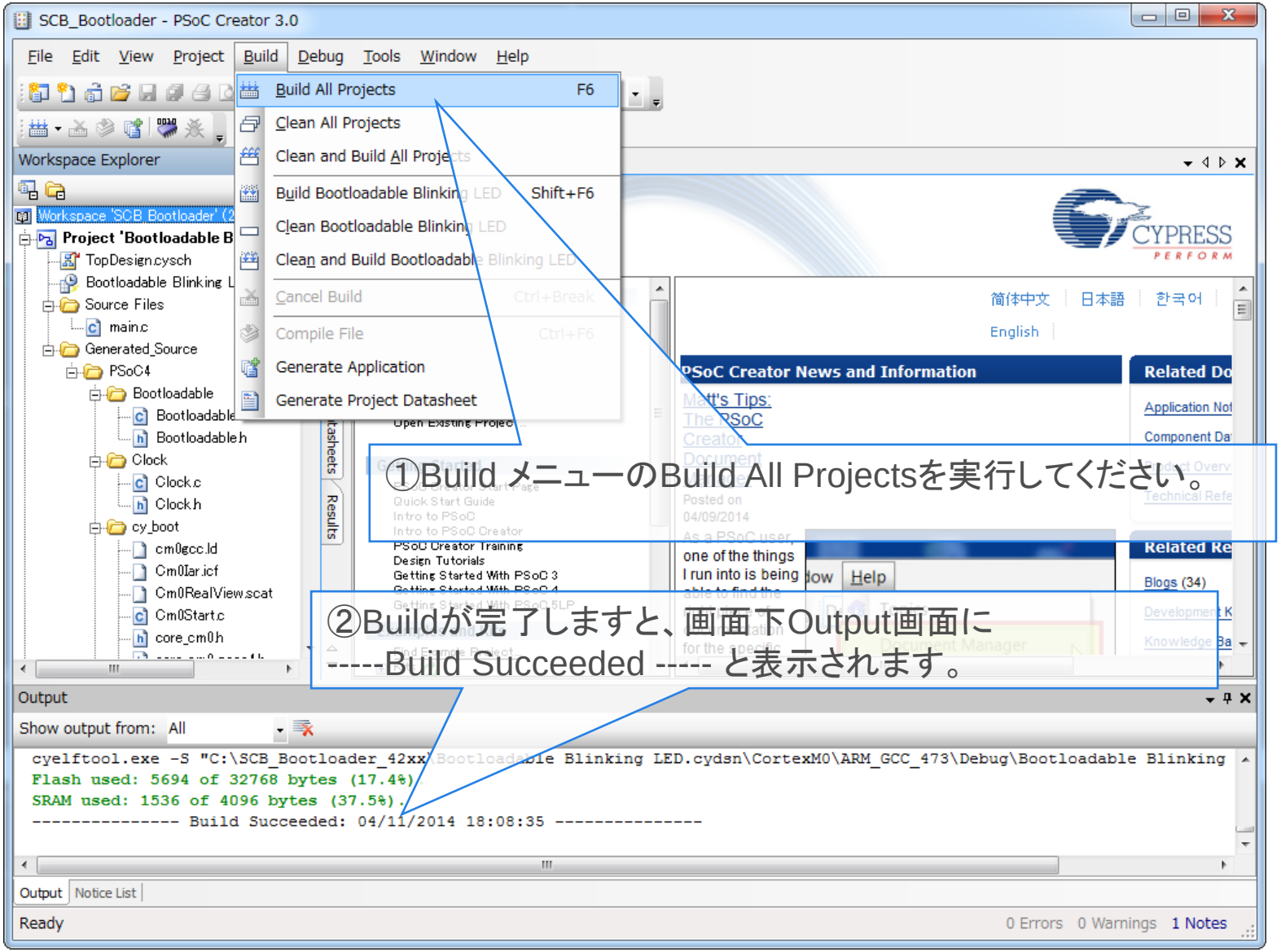
① PWMタブを開いてください。




② Compare値を500から100に変更してください。
LED点滅時の点灯時間が500ms => 100msになります。
もしCompare値の上にある、Period値を変更した場合、LEDの点滅周期が変わります。

③ 変更が終わったらOKをクリック。

# 動作確認


SCB_Bootloader - PSoC Creator 3.0
File Edit View Project Build Debug Tools Window Help
Build All Projects F6
Clean All Projects
Clean and Build All Projects
Build Bootloadable Blinking LED Shift+F6
Clean Bootloadable Blinking LED
Clean and Build Bootloadable Blinking LED
Cancel Build Ctrl+Break
Compile File Ctrl+F6
Generate Application
Generate Project Datasheet
Workspace Explorer
Workspace 'SCB_Bootloader' (2
Project 'Bootloadable B
TopDesign.cysch
Bootloadable Blinking L
Source Files
main.c
Generated_Source
PSoC4
Bootloadable
Bootloadable.h
Clock
Clock.c
Clock.h
cy_boot
cm0gcc.ld
Cm0Iar.icf
Cm0RealView.scat
Cm0Start.c
core_cm0.h
Output
Show output from: All
cyelftool.exe -S "C:\SCB_Bootloader_42xx\Bootloadable Blinking LED.cydsn\CortexM0\ARM_GCC_473\Debug\Bootloadable Blinking
Flash used: 5694 of 32768 bytes (17.4%).
SRAM used: 1536 of 4096 bytes (37.5%).
---------- Build Succeeded: 04/11/2014 18:08:35 ----------
Output Notice List
Ready
0 Errors 0 Warnings 1 Notes
CYPRESS PERFORM
简体中文 日本語 한국어
English
PSoC Creator News and Information
Related Do
Related Re
Blogs (34)
Help
①Build メニューのBuild All Projectsを実行してください。
②Buildが完了しますと、画面下Output画面に
-----Build Succeeded ----- と表示されます。

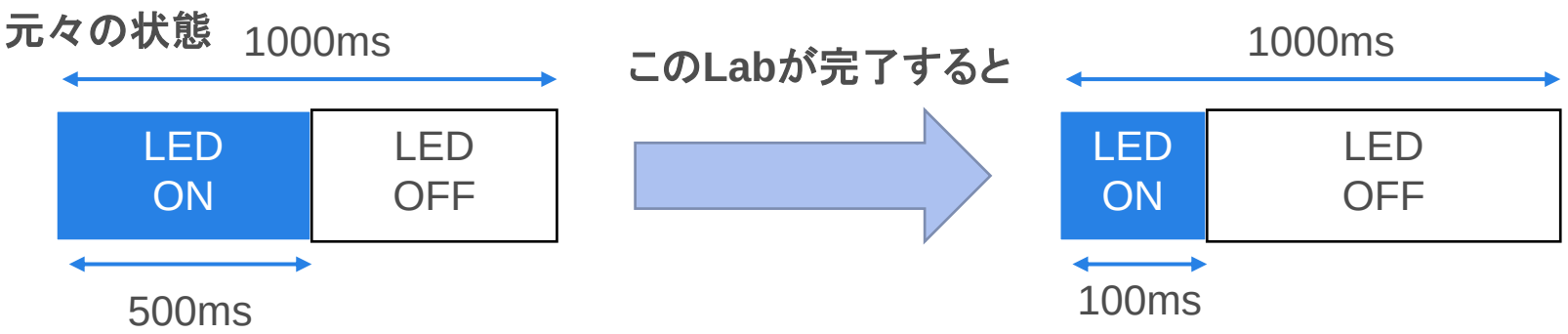
# デバイスへの書き込み

**本資料 P.17～21を参照してください。**

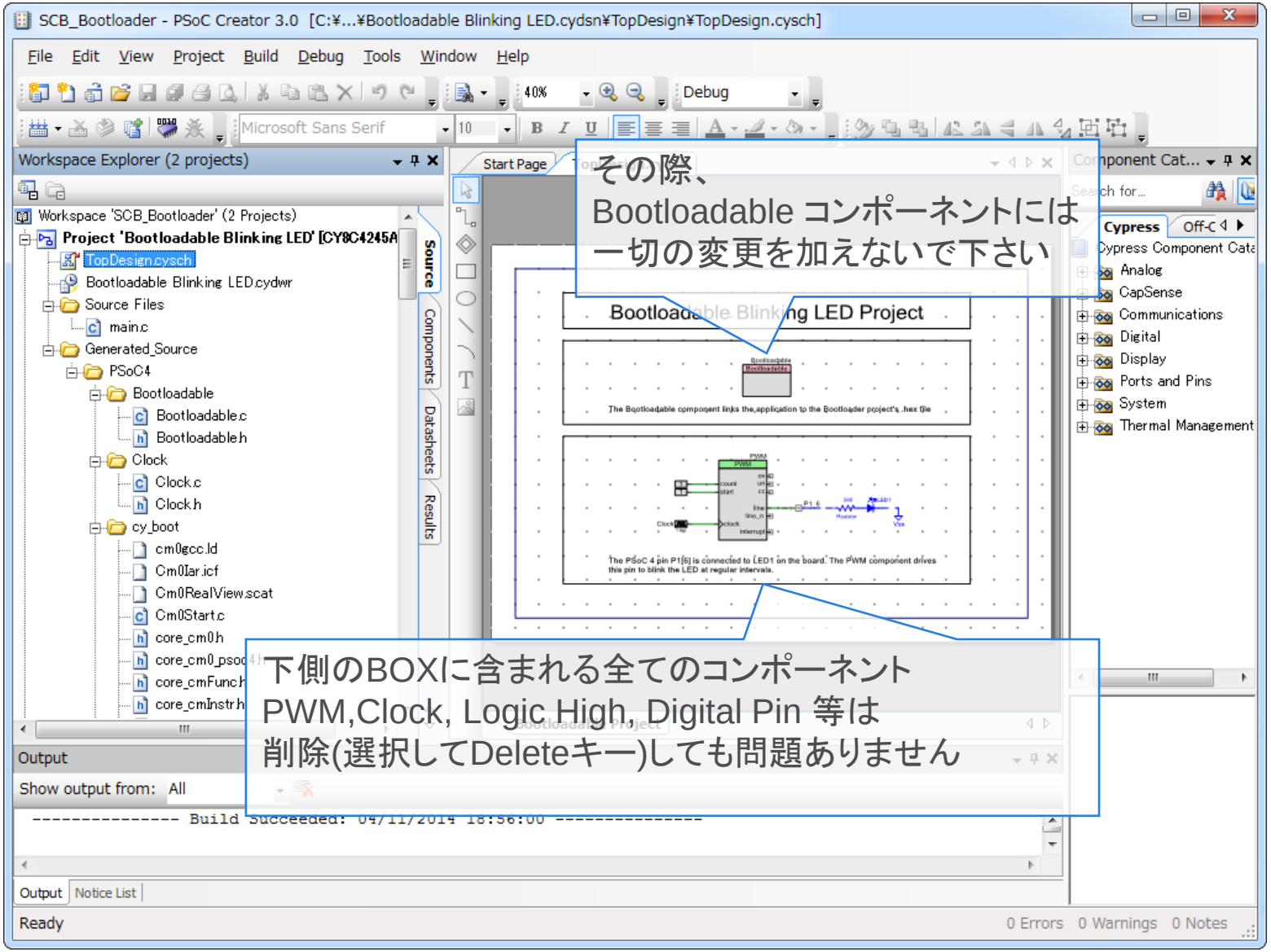
# 正しく書き換えられた場合

Bootloader 経由での書き換えが完了しますと、デバイスはBootloadable ( ユーザーが定義するApp )に自動的に切り替わります。

青いLEDの点灯時間が短くなったことを確認してください。



# サンプルプロジェクトを編集することで、様々な機能の評価を行えます




その際、
Bootloadable コンポーネントには
一切の変更を加えないで下さい

下側のBOXに含まれる全てのコンポーネント
PWM,Clock, Logic High, Digital Pin 等は
削除(選択してDeleteキー)しても問題ありません

# テクニカル サポートは

本開発キットの使用方法などに関するご質問は、本社がWebにて対応しておりますテクニカル サポート(英語)をご利用ください。日本サイプレスでは対応ができません。あらかじめご了承ください。

Webでのテクニカル サポートのご利用は:

japan.cypress.comのどのページからでも、添付図の[サポート]-[テクニカル サポート]-[MyCaseを作成]よりアクセスし、ご質問内容をCaseとして作成し送信してください。

※貴社ご担当の販売代理店さんがいらっしゃる場合は、その代理店さんにご相談されることをお勧めいたします。

# Appendix

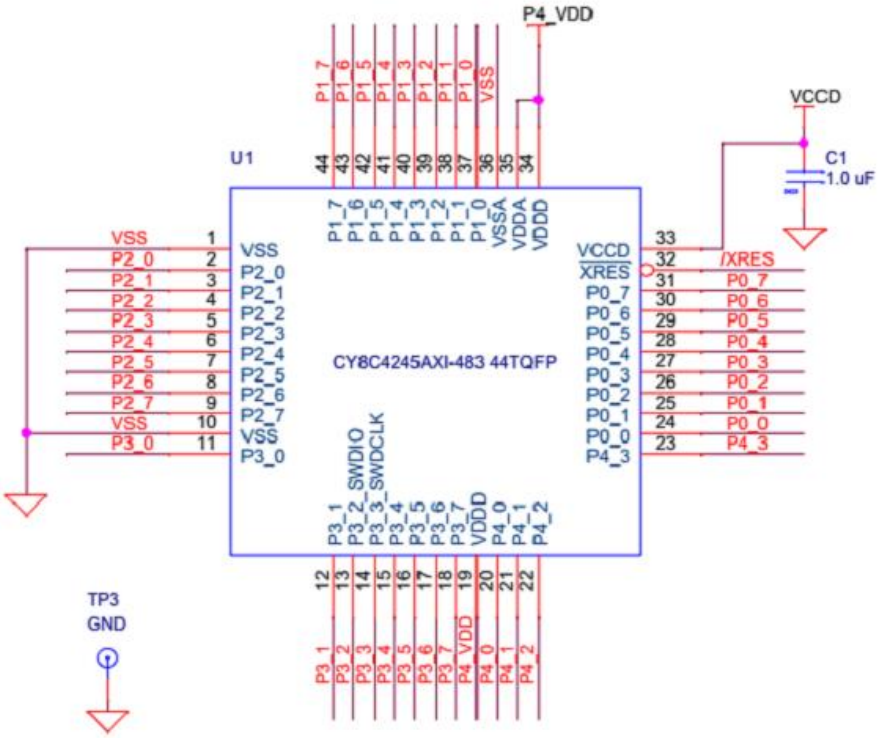
# MiniProg3での書き換え方法

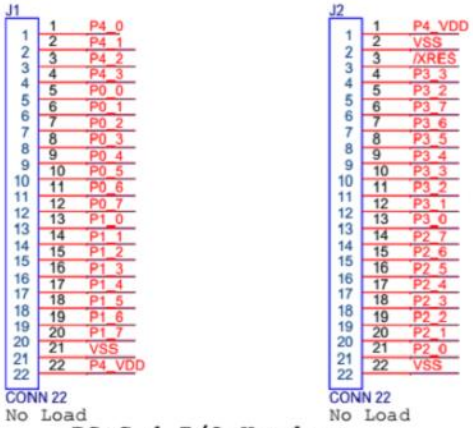
**CY8CKIT-049に**
**ピンヘッダを付けることで、**
**MiniProg3でも書き換える事が出来るようになります**

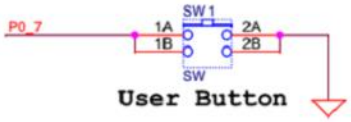
CY8CKIT-002 MiniProg3
http://jp.cypress.com/?rid=38154

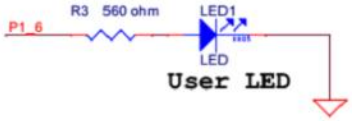
# CY8CKIT-049 回路図 ( PSoC側)

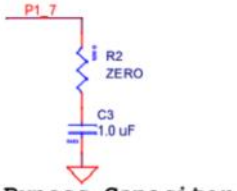
Figure A-1. CY8C4245AXI-483 (CY8CKIT-049-42XX only)

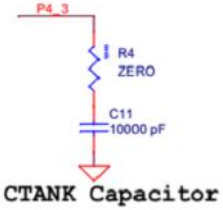


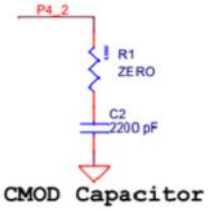





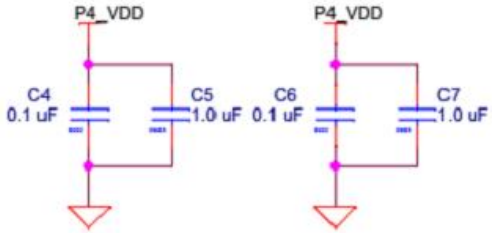



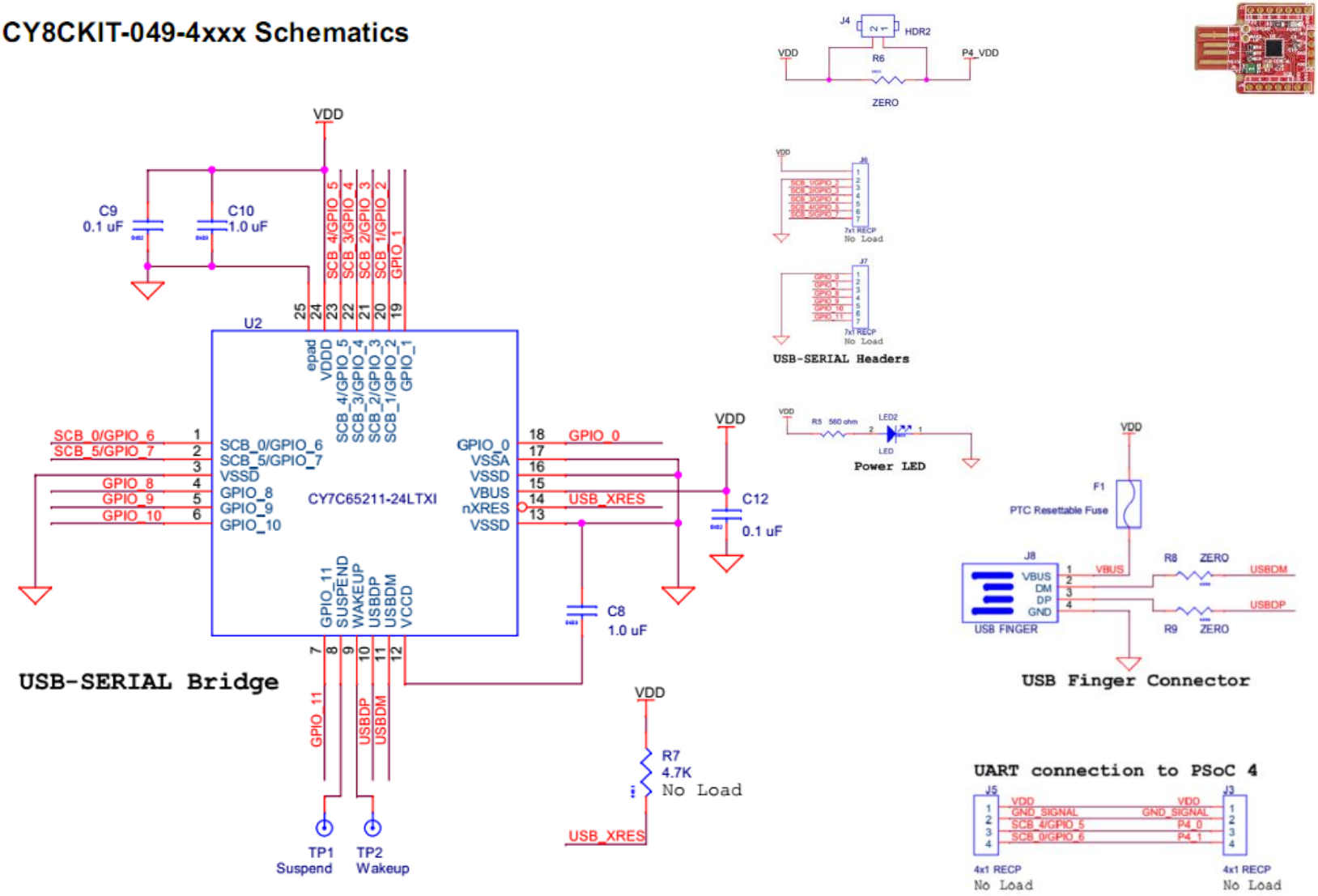




Place Capacitors close to Power Pins



# CY8CKIT-049 回路図 ( USB-Serial変換側)

## A.1 CY8CKIT-049-4xxx Schematics