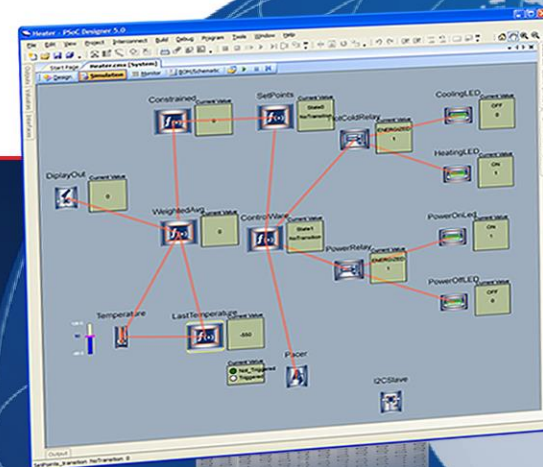


デュアルPWM クローンプロジェクトの演習

# lab2\_pwm\_lcd

## PSoC Experiment Lab

Experiment Course Material V1.31  
April 9<sup>th</sup>, 2019  
lab2\_pwm\_lcd.pptx (19Slides)  
Renji Mikami



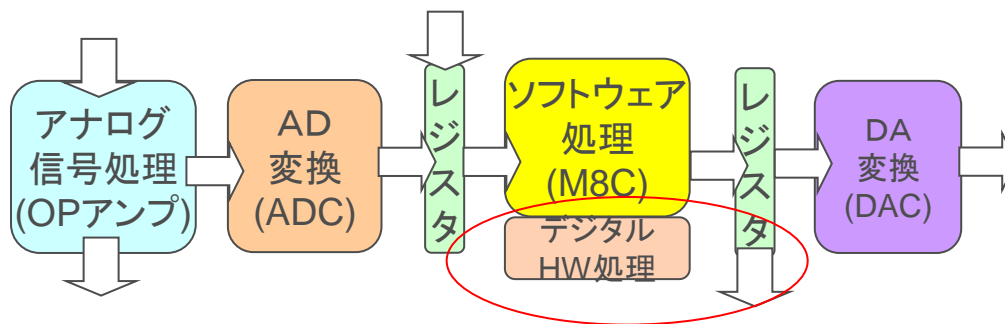


2つのPWMモジュールを使用して積分値を連続的に変化させて  
アナログ的な出力変化を作ります

# ラボ

## lab2\_pwm\_lcd

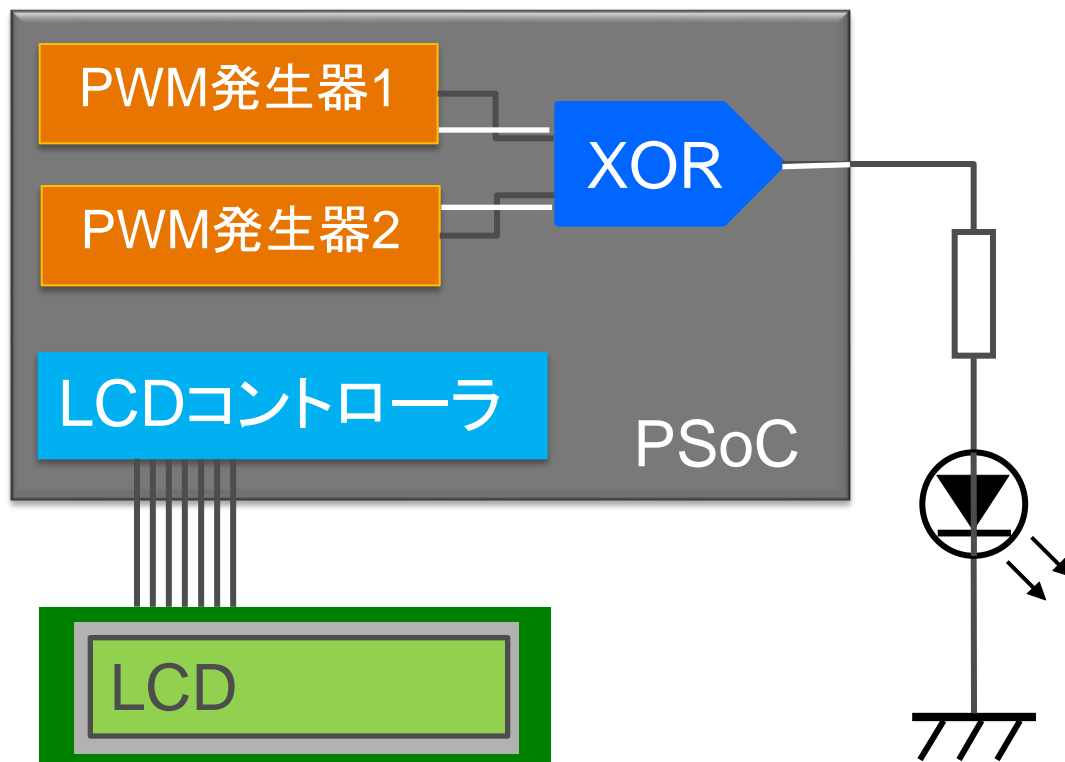
クローンプロジェクトの作り方と  
モジュール相互の論理関数の作り方がポイント





# lab2\_pwm\_lcd

- 周波数の少し違う2つのPWM8  
の出力信号のXORをとり、  
デジタル的に”うなり共振”  
を作り出し、LEDを蛍の光  
のように明滅させる
- LCDを使い文字を表示





# ラボ lab2\_pwm\_lcd 手順

1.lab1\_pwm からクローン/プロジェクトを作成

2.PWM8 を追加しXORをとる

3.PWM8\_1とPWM8\_2のPeriod RegisterとPulse Width Register の設定を行います

4.LCDを追加します

5.プログラムしてLEDを点灯します.この音を聴いてみます.

6.プログラムからPWMのレジスタ値を変えてみます.

解説：クローンプロジェクトの作り方,日本語ユーザーモジュールの開き方も演習します

# 新規プロジェクト作成(旧版ソフトウェアの場合)

1. File > New Project をクリック

2. Chip-level Project を選択

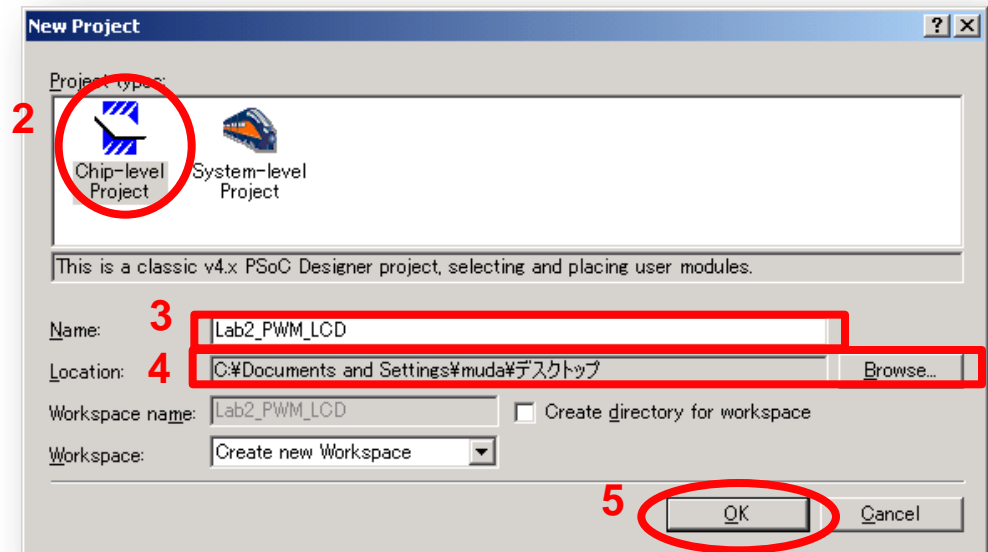
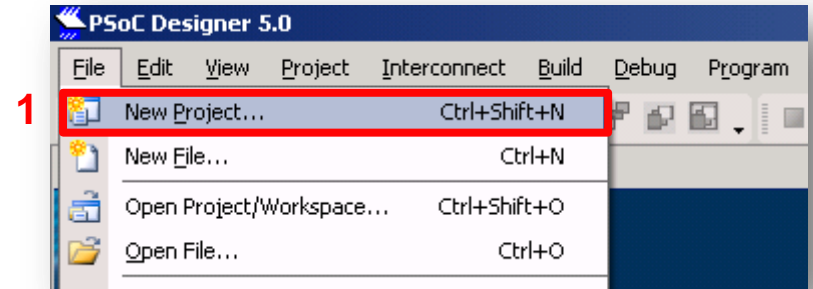
3. Name を入力

例: lab2\_pwm\_lcd

4. Location を選択

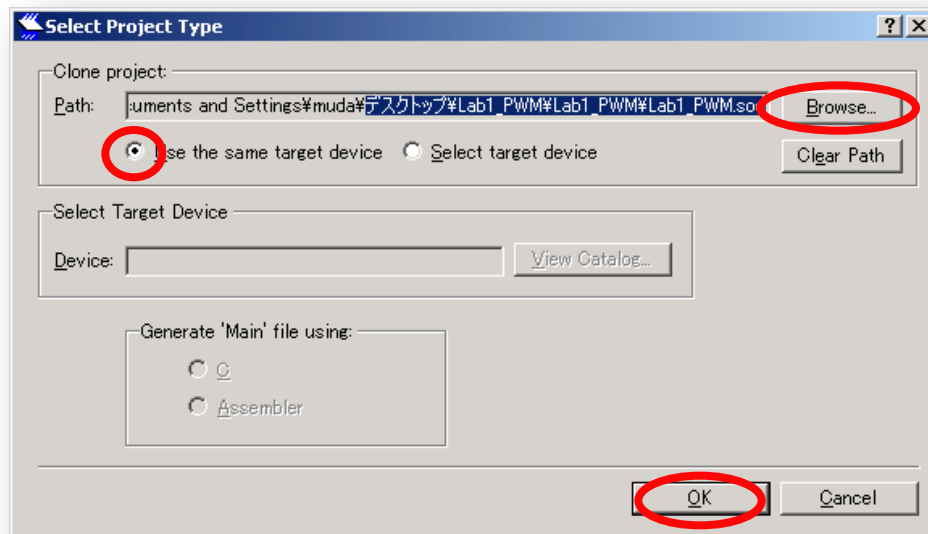
例: C:\psoc\_lab\lab2\_pwm\_lcd

5. OK をクリック



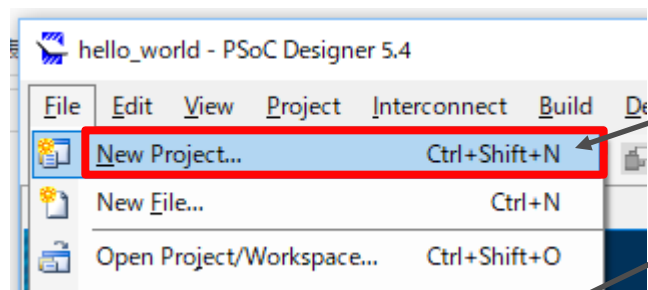
# クローン元プロジェクトの選択(旧版ソフトウェアの場合)

- Browse... をクリック
- lab1\_pwmで作成した .socファイル を選択  
例: C:\psoc\_lab\lab1\_pwm\lab1\_pwm.soc
- Use the same target device  
を選択(他のデバイスに変更  
する場合はここで入力変更)
- OK をクリック





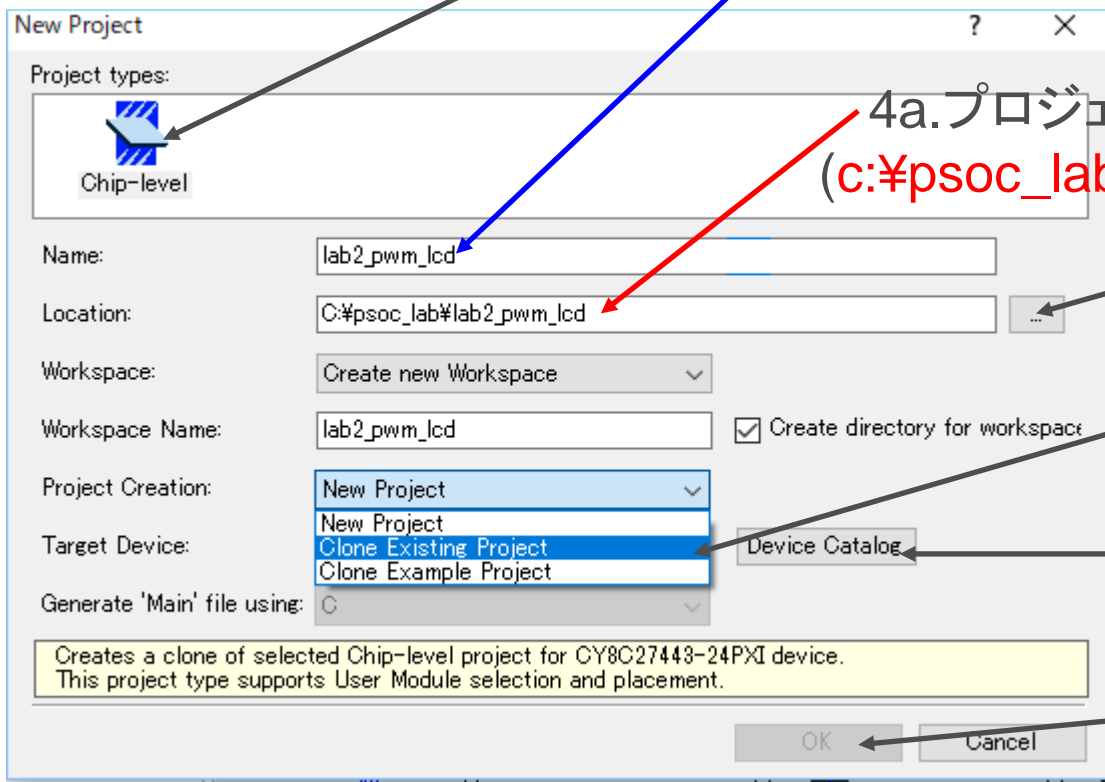
# New Project: lab2\_pwm\_lcd の作成 (新版ソフトウェアの場合)



1. File > New Projectをクリック

2. Chip-level Project をハイライト

3. プロジェクトの名前(lab2\_pwm\_lcd)を入力



4a. プロジェクトを保存するディレクトリ (c:\psoc\_lab\lab2\_pwm\_lcd) 指定する。

4b. 場所を選ぶときは右の...をクリック。

5. Clone Existing Projectを選択

(6. もしデバイスを変更する場合はクリック)

7. 確認したらOKをクリック

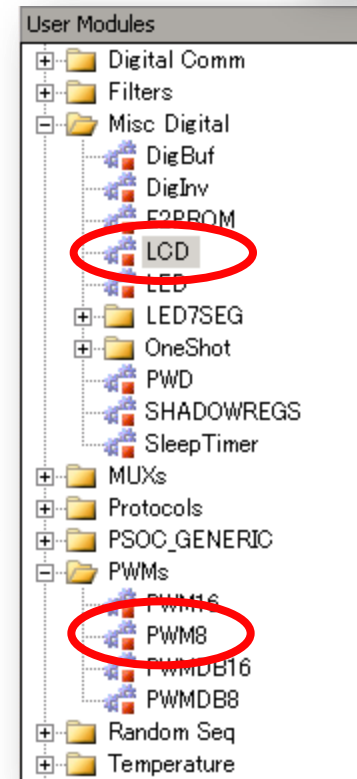
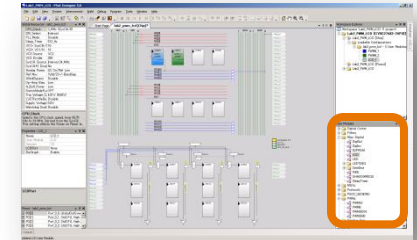
# ユーザーモジュールの追加

View > User Module Catalog

1.PWMs > PWM8 をダブルクリック

2.Misc Digital > LCD をダブルクリック

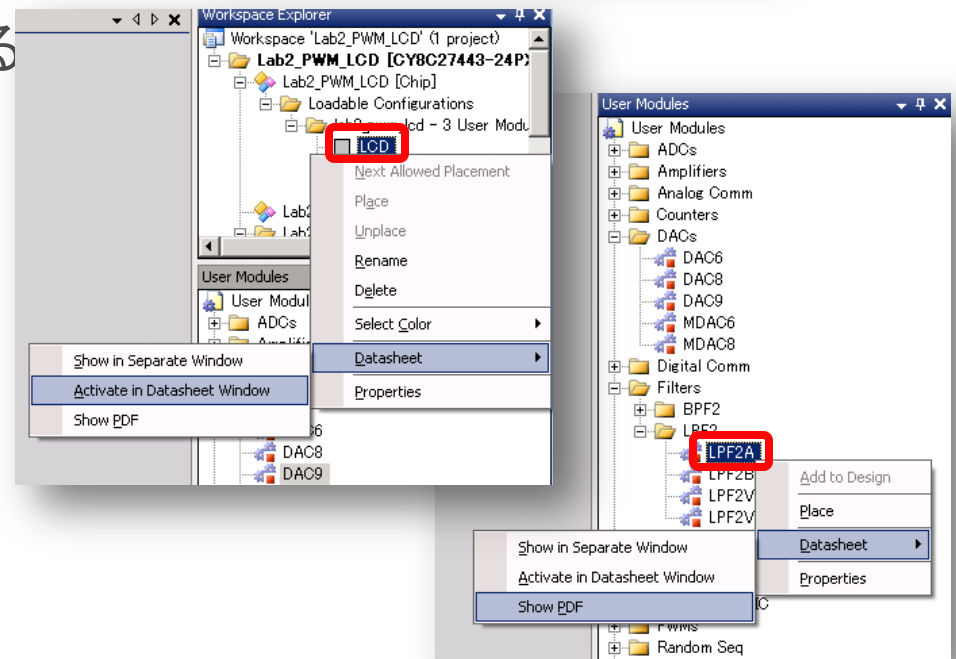
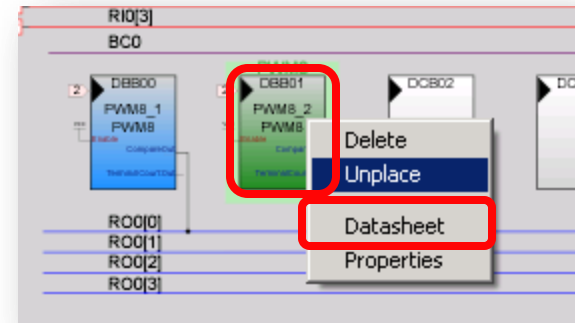
LCDユーザーモジュールは  
ブロックを使わない。  
ソフトウェアユーザーモジュール





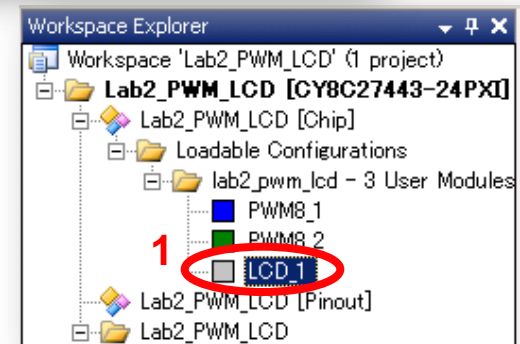
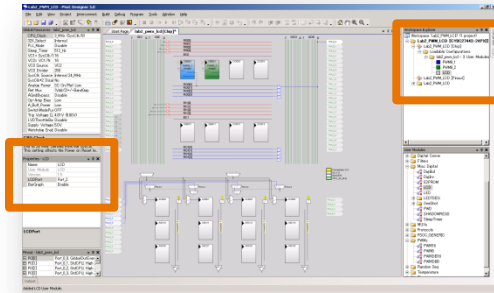
# ユーザーモジュールのデータシートの開き方

- デジタルブロックを右クリック  
Datasheet をクリックで  
データシートを表示してみる
- View > User Module Catalog  
からもデータシートを表示できる



# LCDパラメータの設定

1. 画面右上Workspace Explorer内の LCD\_1 をクリック
2. 画面左LCD\_1のパラメータを設定  
名前の変更 LCD\_1 → LCD  
使用ポートの指定 Port2



2

Properties - LCD	
Name	LCD
User Module	LCD
Version	1.5
LCDPort	Port 2
BarGraph	Enable

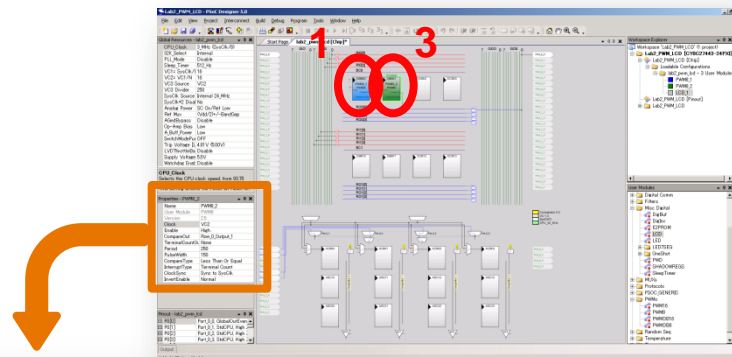
# PWM8パラメータの設定

1. デジタルブロック上のPWM8\_1をクリック

2. PWM8\_1のパラメータを入力  
Period 249, Pulse Width 124

3. デジタルブロック上のPWM8\_2をクリック

4. PWM8\_2のパラメータを入力  
Period 250, Pulse Width 150



2

Properties - PWM8_1	
Name	PWM8_1
User Module	PWM8
Version	2.5
Clock	VC2
Enable	High
CompareOut	Row_0_Output_0
TerminalCountOut	None
Period	249
PulseWidth	124
CompareType	Less Than Or Equal
InterruptType	Terminal Count
ClockSync	Sync to SysClk
InvertEnable	Normal

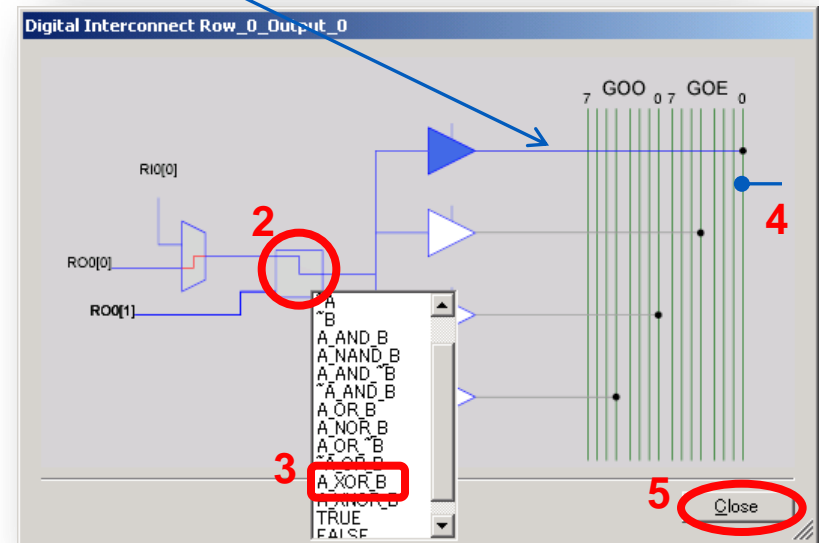
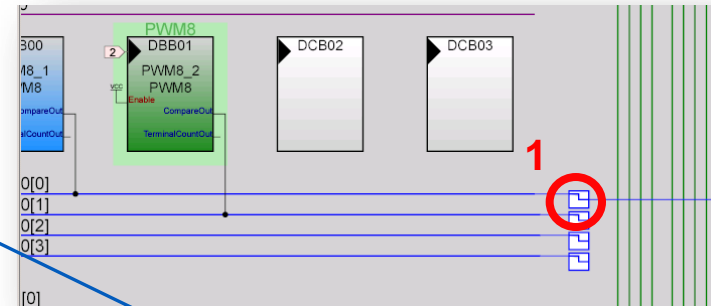
4

Properties - PWM8_2	
Name	PWM8_2
User Module	PWM8
Version	2.5
Clock	VC2
Enable	High
CompareOut	Row_0_Output_1
TerminalCountOut	None
Period	250
PulseWidth	150
CompareType	Less Than Or Equal
InterruptType	Terminal Count
ClockSync	Sync to SysClk
InvertEnable	Normal

# PWM8出力の配線

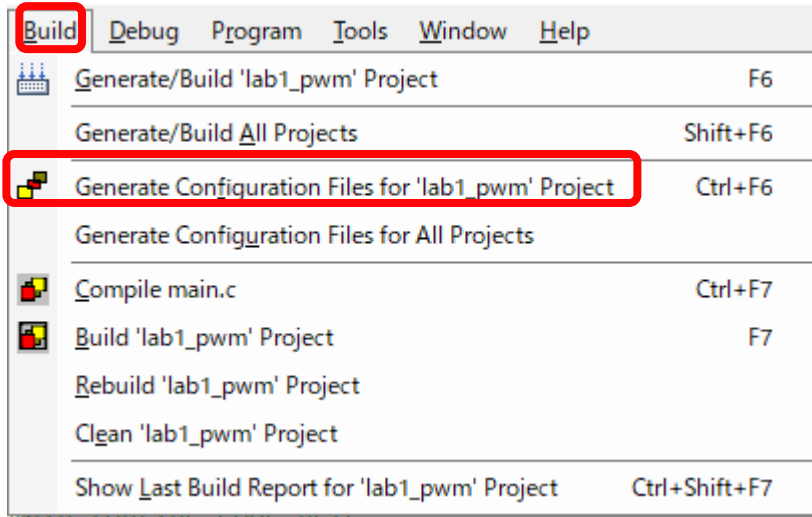
1. RO0[0]の右端のブロックをクリック
2. 正方形のブロックをクリック
3. A\_XOR\_B を選択
- 4.自分で考えて出力ピンまで配線を行う
- 5.Close をクリック
- 6.出力ピンのボックスソケットとLEDのボックスソケットをジャンパーでつなぐ

移動	Alt+ドラッグ
拡大	Ctrl+クリック Ctrl+ドラッグ
縮小	Ctrl+shift+クリック Ctrl+shift+ドラッグ

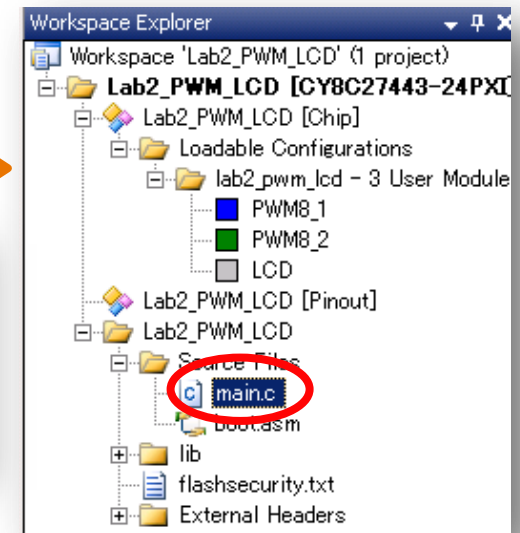
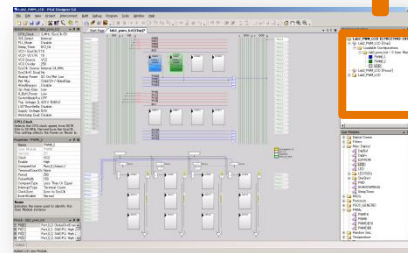


# GC(Generate Configuration)

- Build >  Generate Configuration Files... をクリック



- GCが終了したら、main.c をダブルクリック  
ソースコード記述画面へ



# ソースコード記述

- main関数内にプログラムを入力

LCD\_Position文を追加すると文字の表示位置を指定できる。

LCD\_Position(0,0);

0:一行目

1:二行目

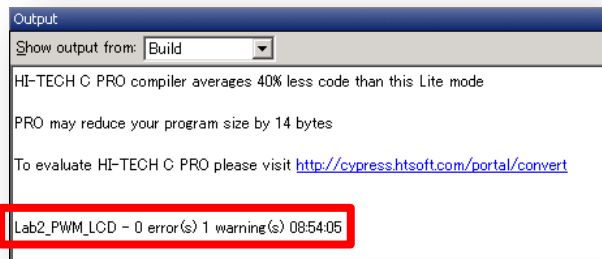
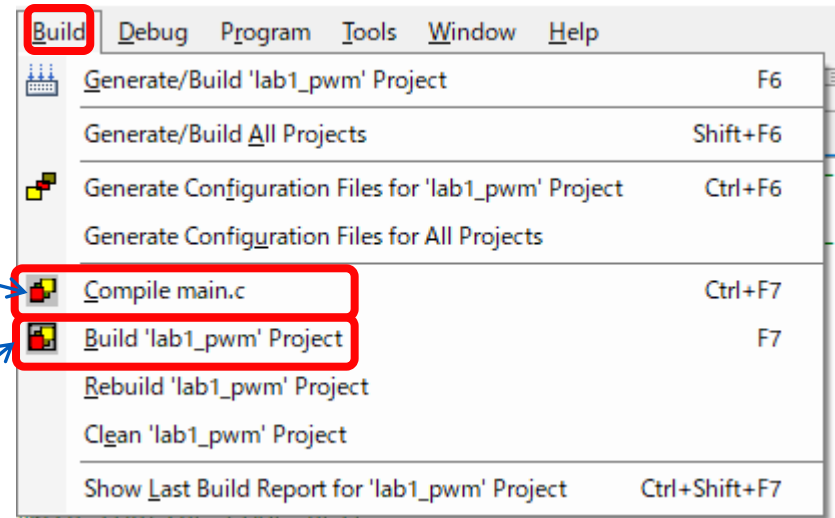
文字の書き出し位置  
左端が0

```
Start Page lab2_pwm_lcd [Chip] main.c
1 //-----
2 // C main line
3 //-----
4
5 #include <m8c.h> // part specific
6 #include "PSoC_API.h" // PSoC API defi
7
8
9 void main()
10 {
11     PWM8_1_Start();
12     PWM8_2_Start();
13     LCD_Start(); LCD_Position文はここに入れる
14     LCD_PrCString("Hello, world!");
15 }
16
```

# コンパイルとビルド

1. Build > Compile mail.c をクリック

2. Build > Build 'lab2\_pwm\_lcd'  
をクリック

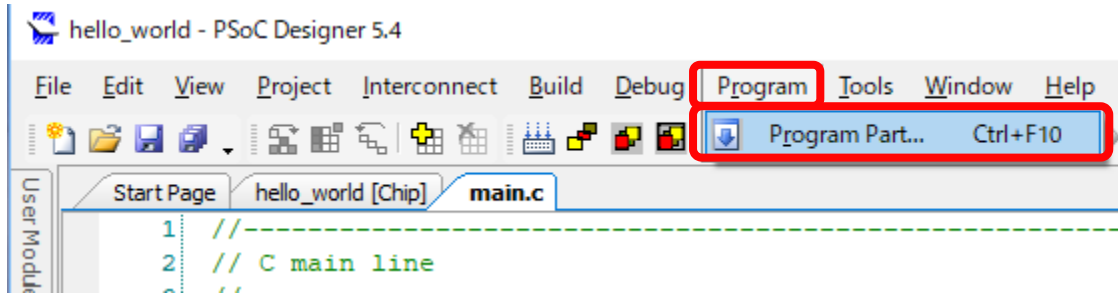


0 error(s) と出れば成功

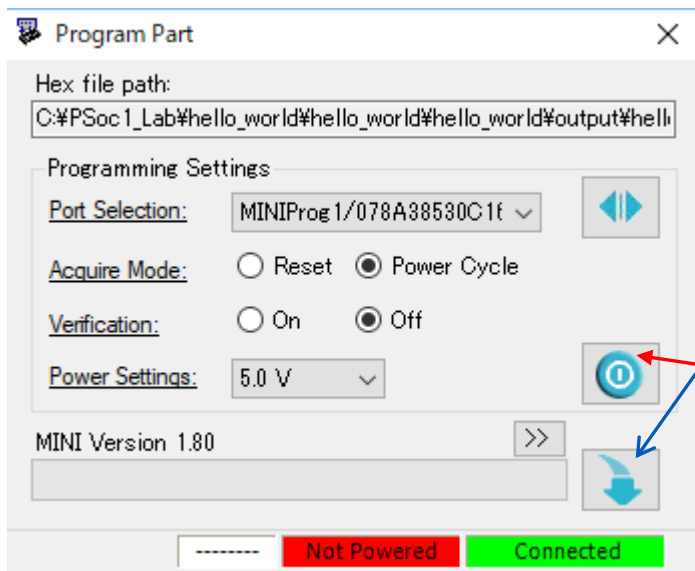
1 warning(s) は無視してよい



# 書き込み

- Program > Program Part をクリック



PSoC Designer から Program > Program Part をクリックすると、PSoC Programmer が自動的に起動し作成されたhex ファイルがロードされる。

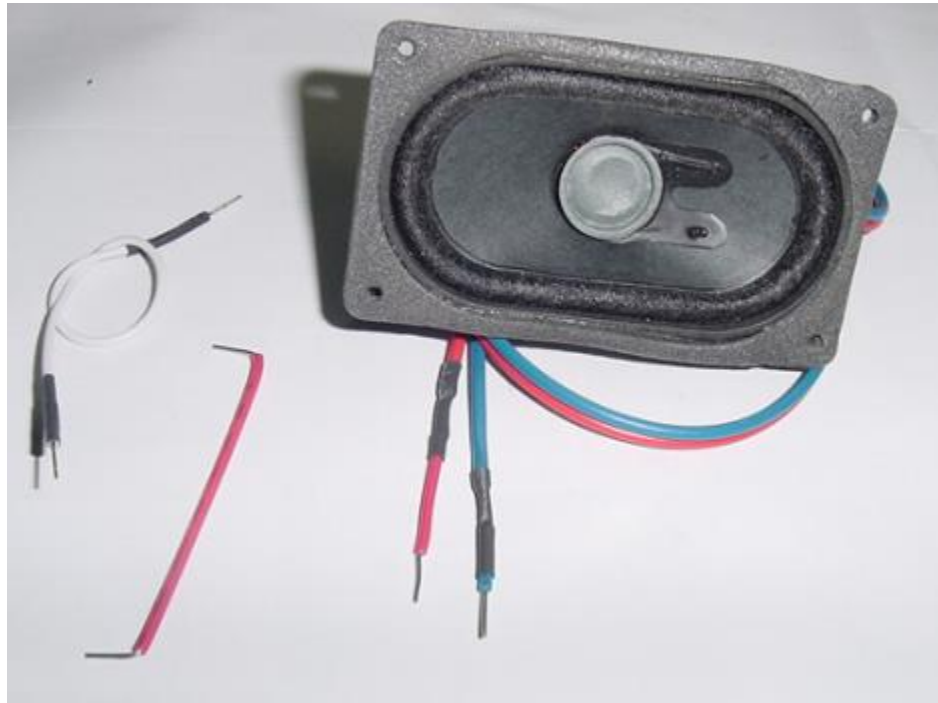


-  をクリックすると書き込み開始
- Actions を読んで状況を確認
-  をクリックすると MINIProgを通じて電源を供給
- LEDの光り方を観察し、考察



# 自由課題:スピーカーでPWMの音を聴く

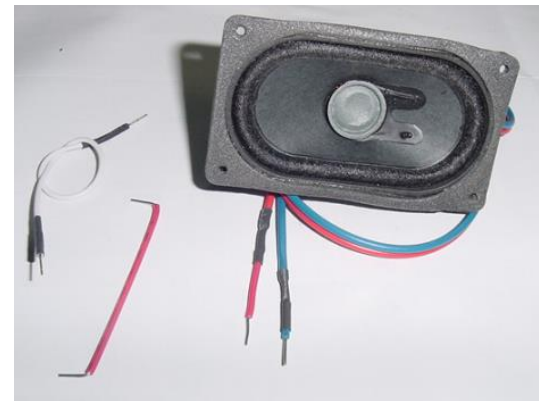
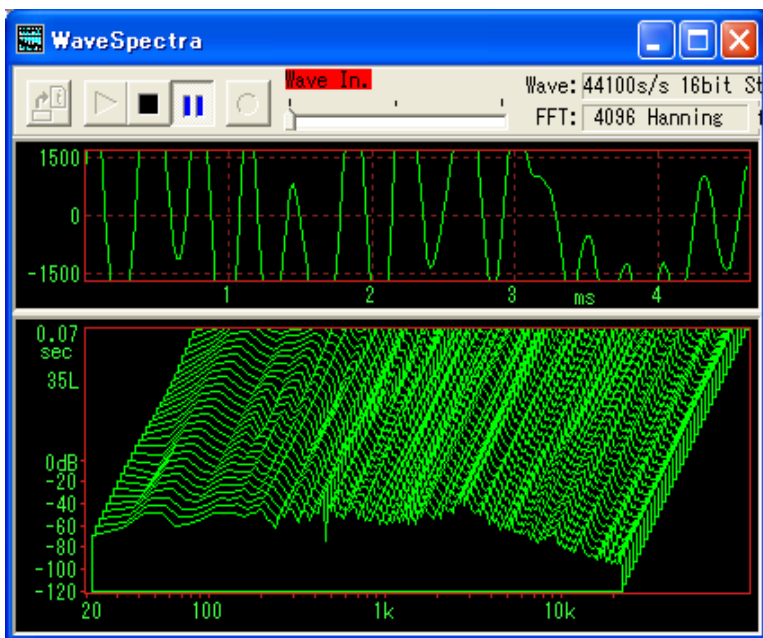
LEDのホタル的明滅は,どんな音になるだろう



LEDの場所にスピーカーをつないで音を出してみる(WaveSpectraで波形を観測してみてもよい)  
2つのPWMのデューティー(周波数)の差を変更して音と光を変えてみる  
この現象を考察してみる

# 自由課題: WSで観察

WS(Wave Spectra)で見てみる



いろいろな  
チャレンジをしてみよう  
やる前に考えてみる  
自分なりに**仮説**をたててから試行  
思いどおり動かないときこそチャンス

時間のない人は最後の自由課題時間に試行してみてください  
ワークショップ終了後に各自で試してみてもいいです

# Memo

フォローアップURL

<http://mikami.a.la9.jp/meiji/MEIJI.HTM>



担当講師

三上廉司(みかみれんじ)

Renji\_Mikami(at\_mark)nifty.com (Default - Recommended)

mikami(at\_mark)meiji.ac.jp (Alternative)

[http://mikami.a.la9.jp/\\_edu.htm](http://mikami.a.la9.jp/_edu.htm)