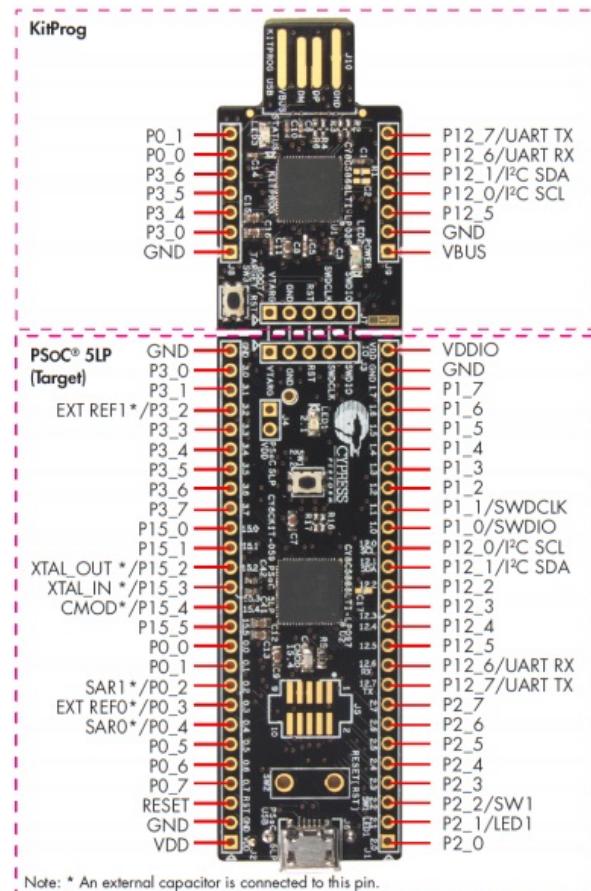


Laboratory 1

COMSYS 301 : Design : Hardware Software Systems

Objectives:

- To explore the PSoC environment
- To program a simple project
- To analyze a design



1. Abstract of the tasks

- Opening the workspace
- Making a project current
- Viewing and exploring the schematic
- Viewing datasheet
- Explaining the function of a schematic
- Viewing the source file
- Building a project
- Testing a program

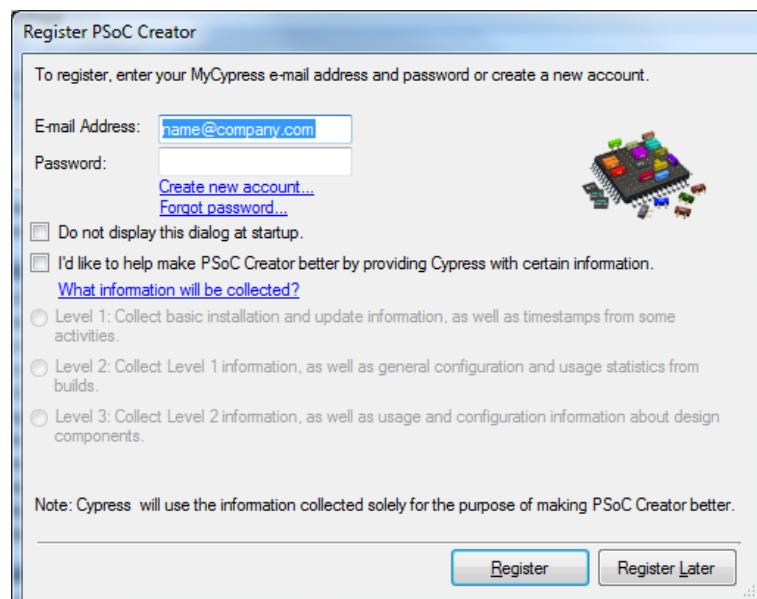
2. Tasks

2.1. Opening the workspace

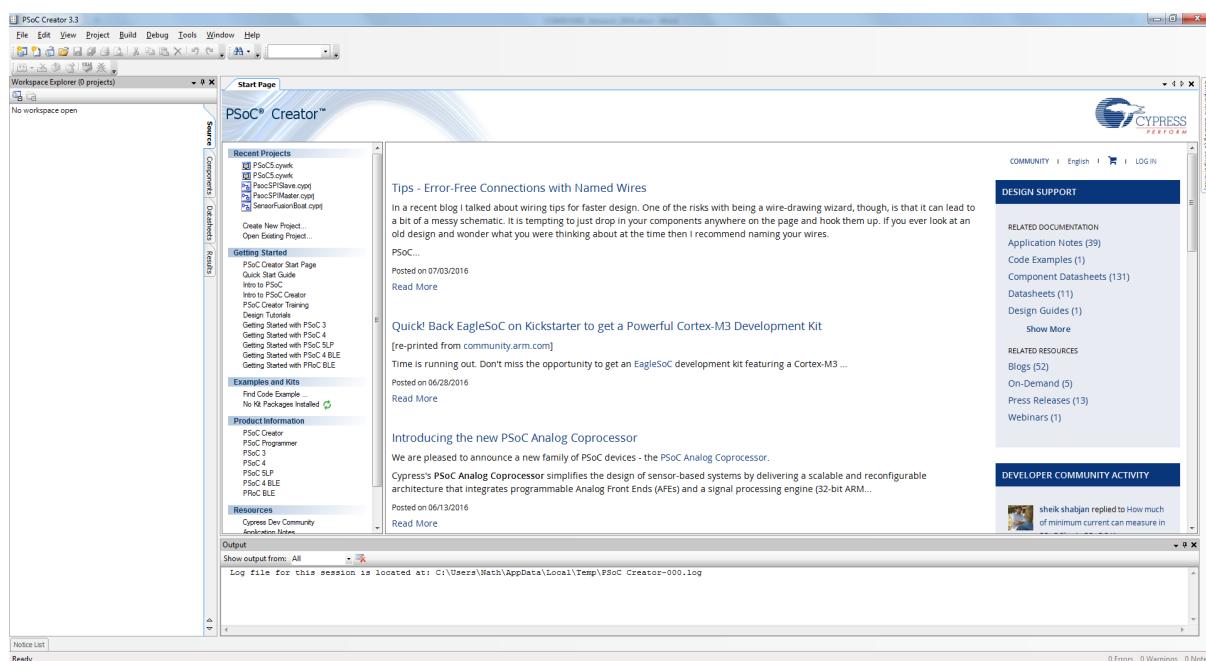
First open PSoC Creator.



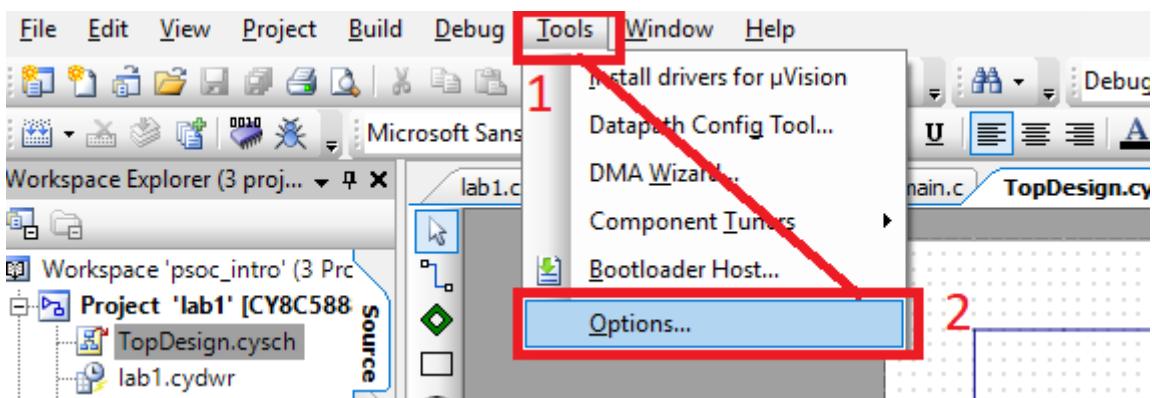
If prompted this dialog box (Register PSoC Creator), press “Register Later”.



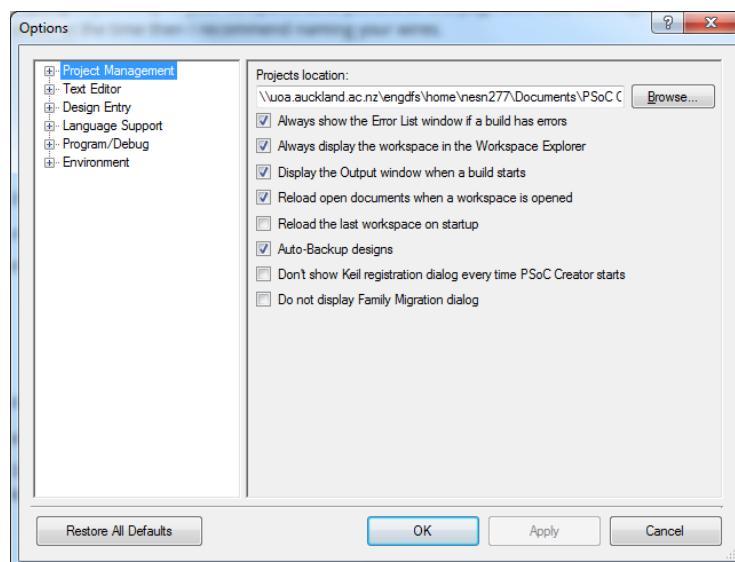
You should now have this on your screen:



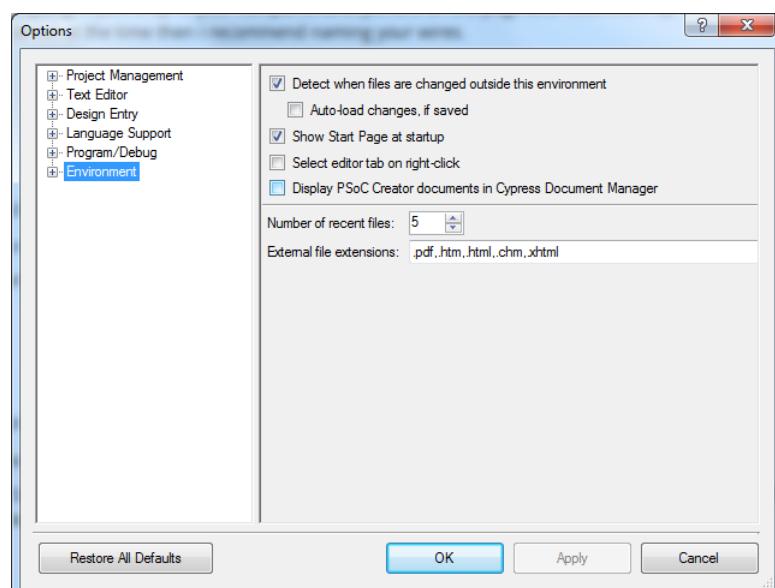
By default the first page you see is the Start Page. You can change that by going to *Tools->Options...* (Alt-T then O)



Then a dialog box opens up. Lots of different GUI options can be edited there to fit you convenience.

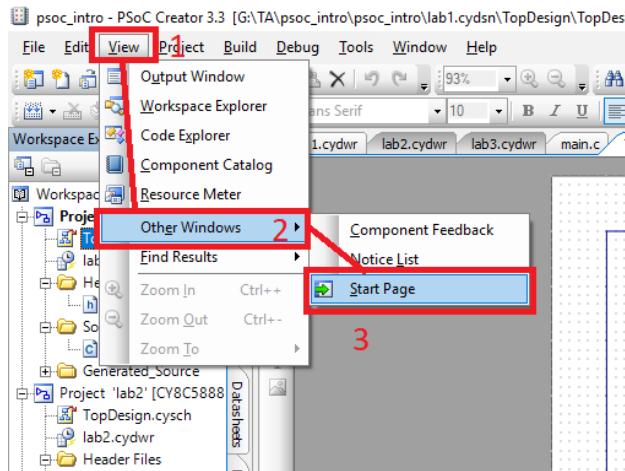


On the left part of the dialog box that opened up you can see a list of group of options, go *Environment*. You should now see a window like this:



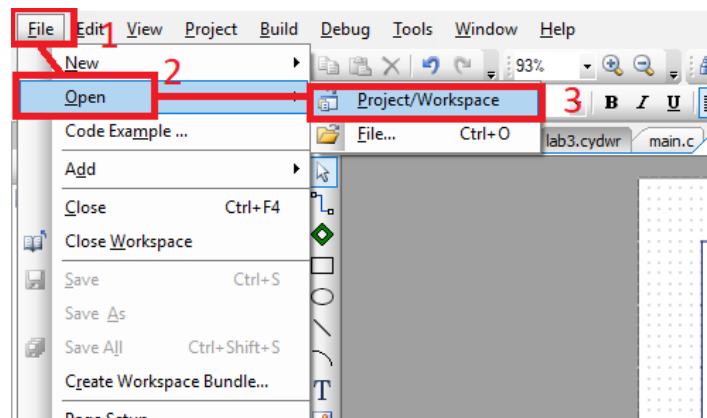
Now, you can untick "Show Start Page at startup" if you want to.

If you want to access the Start Page, you can go to *View->Other Windows->Start Page* (Alt -V then E then S)



Let's load a workspace. First make sure you know the path to the files that will be distributed during the lab by the TAs.

Then, in PSoC Creator go to *File->Open->Project/Workspace* (Alt-F then O then P). Note that there are lots of different options to open a workspace, it is up to you to find the one that suits you the most.



After that step, another windows opens up asking you to select the workspace file. Those file have a **.cywrk** extension (for Cypress Workspace). For this lab, the name of the workspace is **psoc_intro.cywrk**.

Backup	18/07/2016 5:06 p...	Dossier de fichiers
lab1.cydsn	19/07/2016 7:30 a....	Dossier de fichiers
lab2.cydsn	18/07/2016 5:05 p....	Dossier de fichiers
lab3.cydsn	18/07/2016 5:05 p....	Dossier de fichiers
labpack	18/07/2016 5:05 p....	Dossier de fichiers
psoc_intro.cywrk	18/07/2016 6:24 p....	PSoC Creator Wor...
		2 Ko

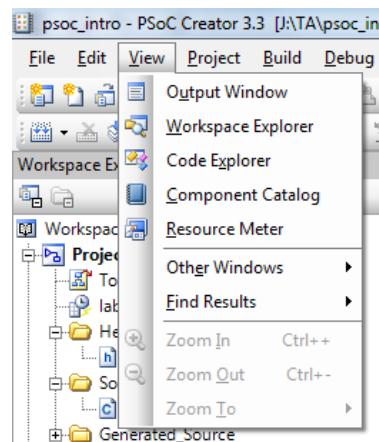
Once selected click on *Open* or press Enter.

Now that your workspace is loaded let's stop there for a moment and identify the different windows.

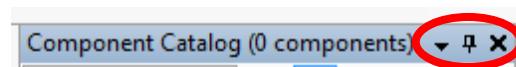
By default, PSoC Creator is divided in 5 parts:

- On top: the *menu bar*
- On the left side the *workspace explorer*
- On the bottom side: the *output window* (mainly compiling messages)
- On the right side : depending on the context, the *component catalog*, *resource meter* or *code explorer*
- On the centre : the files you are currently editing (organised in tabs)

Concerning the right side, you can open different windows by going to *View->* (Alt then V then ...) and select the window you want to see.

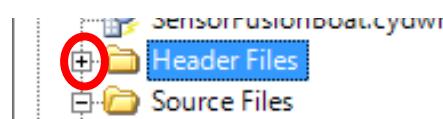


You can change the window options, make them auto-hidable or close them by using the symbols on the top right part of every window.

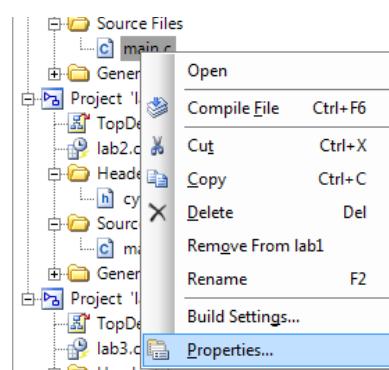


Let's focus on the workspace explorer for a moment.

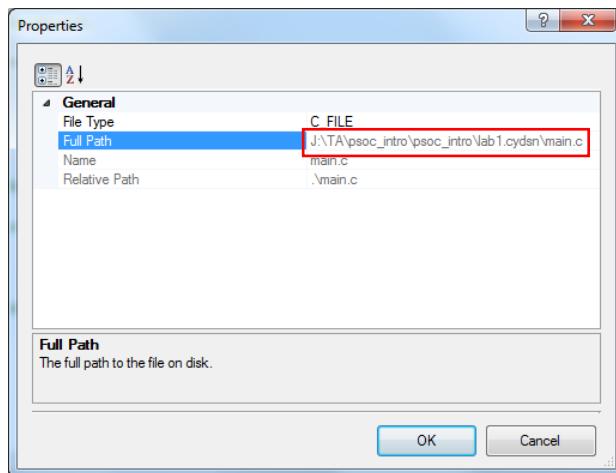
If you put your mouse cursor on the + symbol, a hand icon appears letting you know that you can expand the content of this folder to see what's inside.



Last but not least, it can be useful to get a workspace file path. To do that, on the workspace explorer, find the file you want to see. Right click on it and chose *Properties*.



A window appears and you can see their path in the *Full Path* item.

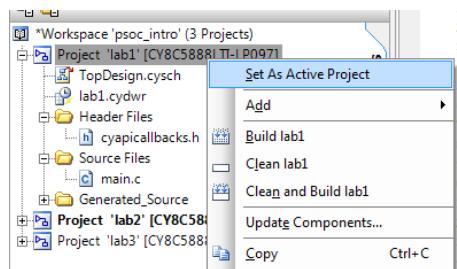


This trick is useful to locate headers (.h) and C files quickly if you need to modify them in another software.

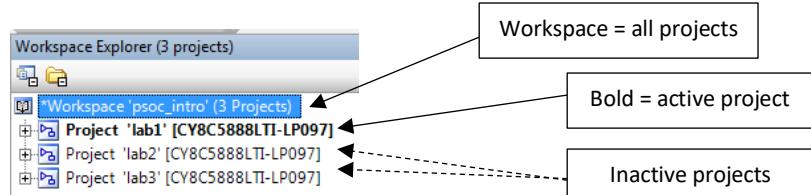
2.2. Making a project current

When you opened the workspace, 3 different projects have been opened (lab1, lab2 and lab3). You can only load 1 project in a PSoC and you can only debug 1 project at a time. In order to tell PSoC Creator on which project you want to work, you have to set one of them as “active”.

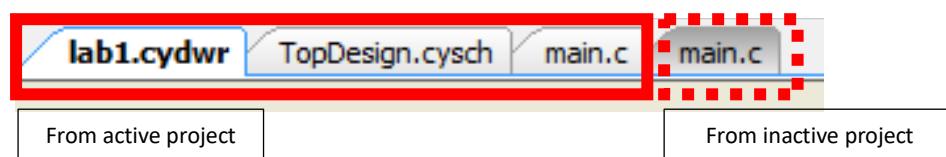
To do that just right click on the project you want to make “active” (in the workspace explorer) and click on “Set As Active Project”.



You can see which project is active by going to the workspace explorer and look for the project which name is in **bold** (There is only one).



If you have files from different projects opened in the central window, you can see that the colours of the tabs is grey for all the inactive project files and white for the active project files.



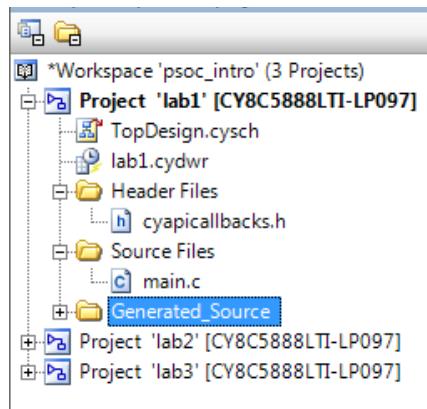
An important remark is that you could be editing an inactive project file but when you press “Program” then the active project will be programmed, not the one you are editing. This could be disconcerting.

Your turn now, set lab1 as the active project.

2.3. Viewing and exploring the schematic

In a project there are 5 main groups of files, you can see them by expanding the content of a project in the workspace window.

Expand the lab1 project now by clicking on the + icon on the left side of the “Project ‘lab1’”. If there is a – icon that means that the project is already expanded. You should see something like that:



You see there are:

- A TopDesign.cysch (for **Cypress Schematic**), it contains the different blocs that you need
- A lab1.cydwr, it contains parameters concerning the FPGA part of the PSoC (port selection, interrupt priority, etc.)
- A “Header Files” folder containing all your .h files
- A “Source Files” folder containing all your .c files
- A “Generated Source”* folder containing a lot of files concerning the blocs you have put in the schematic.
DO NOT EDIT THEM UNLESS YOU KNOW WHAT YOU ARE DOING. It can save hours of debugging.

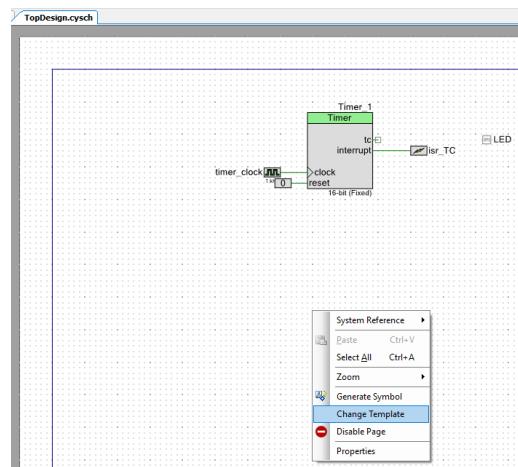
*Note: you may not have this folder if the TopDesign.cysch have never been build or if the API has never been generated.

2.4. Managing the schematic

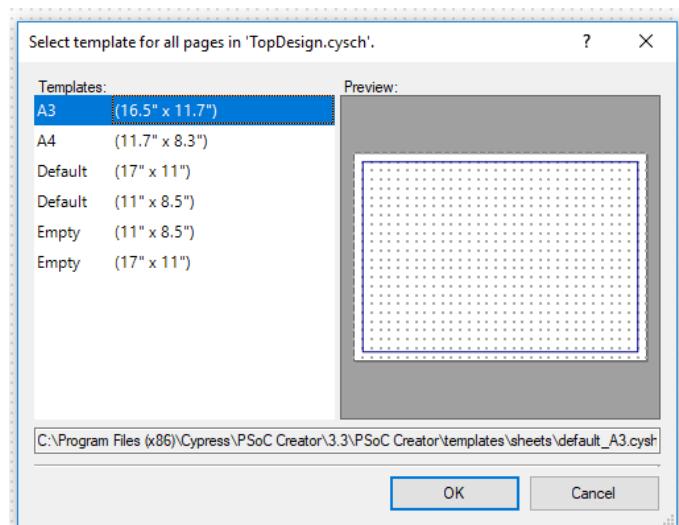
The PSoC5LP has a lot of resources and you might run out of space on the TopDesign page to put all your components. Organising your TopDesign is important for clarity.

The first possibility, is to make the size of the page bigger. By default it is an A4 format. For the sake of the exercise you are going to put it as an A3, allowing more space.

Go to your top design. Right click on an empty space of the white background and choose “Change Template” as shown below.



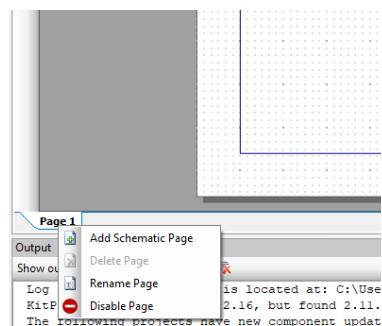
Then a window pops up. On the left box of the window, choose “A3” and press ok, as shown below.



Notice that the area for putting the block is quite larger. But now it might be difficult to read the name of the blocks, also you can still run out of place, especially for your project.

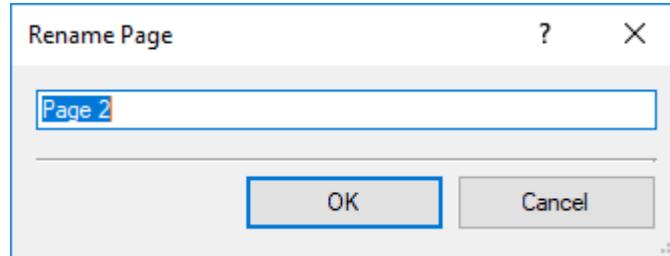
Revert back to an A4 template, we are now going to add pages to the TopDesign and name them appropriately. This is not a complicated task but it is difficult to find how to do it.

On the bottom part of your TopDesign you could find 1 tab actually called “Page 1”. Right click on the “Page 1” text and a list window should appear as below:



Choose add Schematic page. Notice you now have 2 tabs: “Page 1” and “Page 2”. We are now going to rename them “TheTimer” and “Empty” (you are free to choose other names as long as they are appropriate and descriptive). To do

so right click on the “Page 1” text on the tab and choose “Rename Page”. You can now type in the textbox that appeared on the screen as shown below:



You can now change the name of the page 2 as well.

2.5. Viewing datasheet

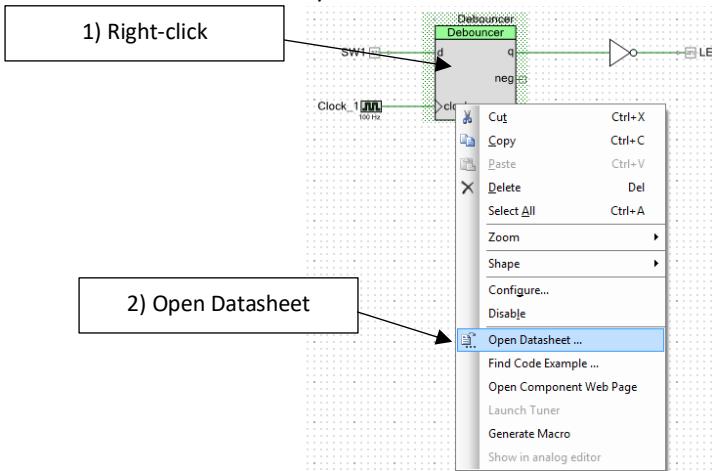
Before using any component from the Cypress library it is primordial to read its datasheet. This task could seem like a waste of time but it is absolutely not. It will save you a **lot** of time of debugging. Moreover, Cypress datasheets are very well done and information can be accessed quickly.

A good example in how a datasheet is critical lies in the fact that some blocks have hidden in/out ports. These ports are shown only if you configure the block correctly (by reading the datasheet).

Another example lies in the fact that some blocks need an initialisation in the .c files and some doesn't. To know if the block you added need this kind of initialisation, just go to the API part of the datasheet.

To open a datasheet component, 2 methods are possible.

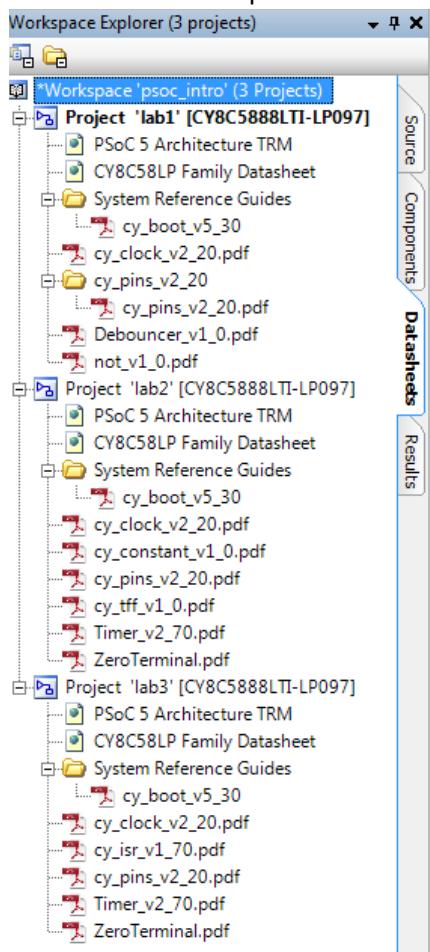
1. Double click on the TopDesign.cysch file on the workspace window, then right click on the component you want the datasheet and the click on *Open Datasheet...*



2. On the right side of the workspace window, you can see 4 tabs.



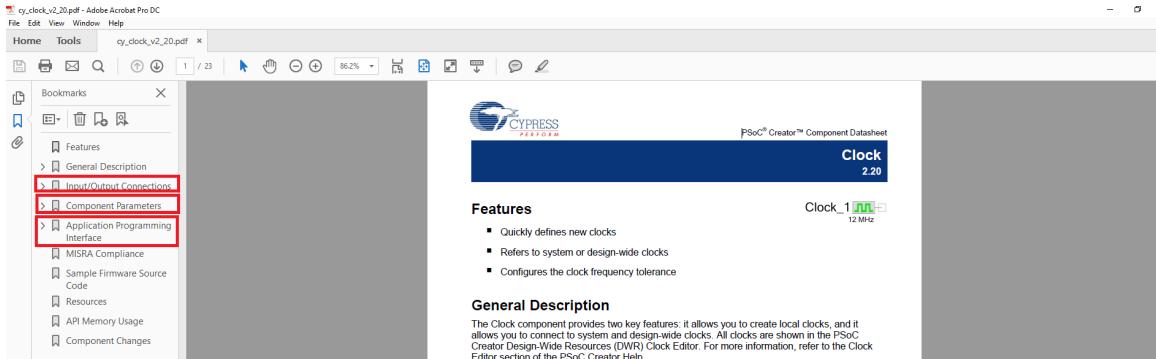
Go to the one called Datasheets and double click on the pdf file which name correspond to the component.



The datasheets contain a lot of information and among them you want particularly to look at:

- Input/Output Connections
- Component Parameters

- Application Programming Interface (API)



This is just the basic minimum, you are welcomed to read all of the other sections of the datasheets.

Let's just open the datasheet of the SW1, Clock_1, Debouncer, not_1 (the not gate), LED.

As you can see the datasheet of the LED is the same as the datasheet of the SW1. The reason is that they both are “cy_pins” components but the parameters of the bloc made them change their aspects.

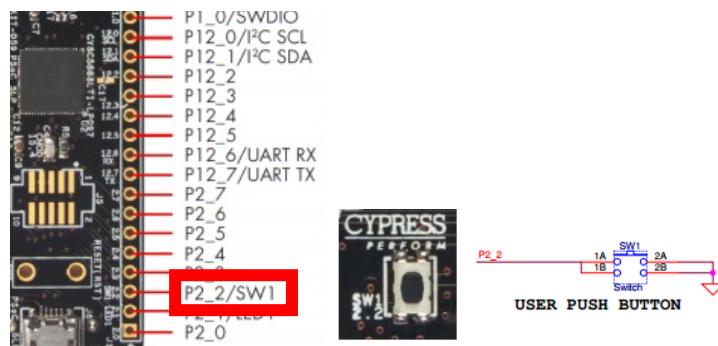
Finally, for each datasheet, look if the section Application Programming Interface (API) exists. Only the clock and the pins have this section. That means that only this type of component have routines that you can call in the .c files.

If a component need to be initialised in the .c code then there would be a “__Start()” routine in the routine list. As you can see neither of the clock and pin have them. That means you don't need to initialise them in the .c code, you can use them directly.

2.6. Explaining the function of a schematic

First let's take a look at the hardware we are using:

- Push Button: The target PSoC 5LP board contains a single push button connected to the P2.2 pin on the PSoC 5LP device. This button can be used for general user inputs or to control different states in an application.



- The blue LED: This is the user LED, connected to P2.1 of the target PSoC 5LP device.



Now open the TopDesign file.

We are going to look at their configuration, each component at a time. Listen to the TA.

2.7. Viewing the source file

To open your main file go to the workspace explorer window in the *source* tab and expand the *Source Files* folder if it was collapsed. Double click on the main.c file to open it. You should see something like that:

```

1  /* ===== */
2  * Laboratory Exercise 1
3  *
4  * Build and download project
5  * Push SW1 to light led
6  *
7  * Copyright Univ of Auckland, 2016
8  * All Rights Reserved
9  * UNPUBLISHED, LICENSED SOFTWARE.
10 *
11 * CONFIDENTIAL AND PROPRIETARY INFORMATION
12 * WHICH IS THE PROPERTY OF Univ of Auckland.
13 *
14 * =====
15 */
16 /*
17 #include <project.h>
18
19 int main()
20 {
21     CyGlobalIntEnable; /* Enable global interrupts. */
22
23     /* Place your initialization/startup code here (e.g. MyInst_Start()) */
24
25     for(;;)
26     {
27         /* Place your application code here. */
28
29     }
30 }
31
32 /* [] END OF FILE */
33
34

```

Note the colour of the tab that just opened in the central window, it should be white if you active project is lab1.

In the code you should just have:

#include <project.h> including the project header

int main() Definition of the main

{

CyGlobalIntEnable; /* Enable global interrupts. */ we don't need that for now

/* Place your initialization/startup code here (e.g. MyInst_Start()) */

for(;;) Your infinite loop, note that you can change it to While(1) if you prefer

```
{
    /* Place your application code here. */

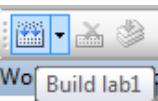
}

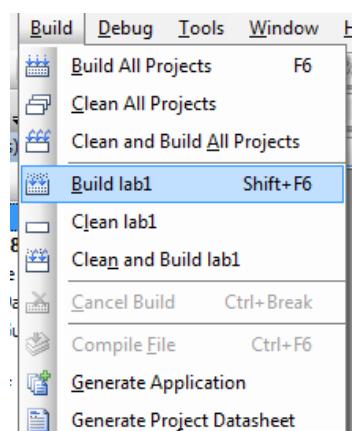
}
```

2.8. Building a project and Programming it

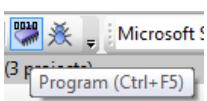
You have to be aware of the difference between Building and Programming. When you build a project, PSoC Creator generate all the files to be programmed in the PSoC but without sending them to the device. Programming is when PSoC Creator take all the generated files and send them to the PSoC.

To build a project you can either:

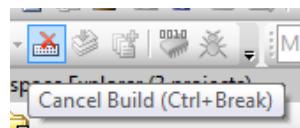
- Press Shift + F6
- Press Alt then B then B
- Click on the build icon 
- Go to Build->Build lab1



To program a project on a PSoC device, you can either:

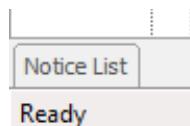
- Press Ctrl + F5
- Press Alt then D then P
- Click on the program icon 
- Go to Debug->Program

Note you can cancel a build any time by clicking on the Cancel Build icon. Same for Programming.

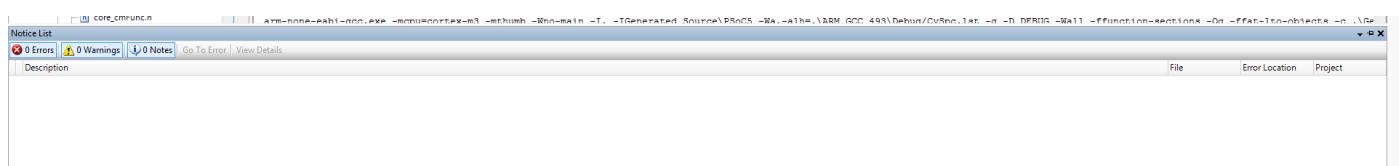


Now build lab1.

If you see something like that “----- Build Succeeded: MM/DD/YYYY HH:MM:SS -----”. Your compilation is successful. To make sure, you can open the Notice list. To do that, just click on the tab on the bottom left.



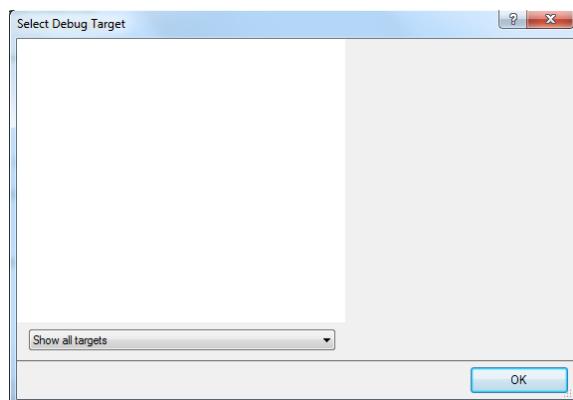
You should see this:



Now plug your PSoC in a USB port of your computer and program it.

In the output window you should see something like that: “Device ‘PSoC 5LP CY8C5888LT*-LP097’ was successfully programmed at _____”

If this windows opens up, that means that PSoC Creator didn’t recognise your PSoC. Unplug it, re plug it, wait 20s and try to program it again.



2.9. Testing the program

Your PSoC is powered by the USB PSoC and is running at 5V. You should see 2 LED (not the one you are controlling) lighting up: amber and green.

- Amber : indicates power on
- Green : indicates status (programming, programmed, sleep mode)

Now press the switch button on the PSoC and look at the blue LED.

3. Your notes

You are encouraged to take notes of the quick questions and remarks made in the lab because some of them may be in the quiz.