PSoC Creator 101: PSoC Creator File Structure

Hello, I'm Alan Hawse; I'm Executive Vice President of Software at Cypress Semiconductor. In this lesson, I will walk you through a simple design and show you the key files that go into a PSoC Creator project.

When you start a new project in PSoC Creator, it places three new application files in the workspace explorer. These files let you create the PSoC design, setup the system resources like the clocks, and the pins, and write the application firmware. First of all, we have topdesign.cysch; this is the schematic file. When you create a new project this file is opened in the editor. This is the file where you draw your PSoC design using the components and wires. In this design I'm going to have just a pin, which I'll use to control and LED on the Pioneer Kit. I'll set the pin name and disable the hardware access to it so I can drive it from the firmware found in main.c. Now, I'll open the design-wide resources file. Which by default has the same name as your project and the extension is cydwr. In this file you'll notice a set of tabs along the bottom of the editor. Each tab corresponds to a specific hardware feature in the device. The first one is for pins; we put a pin in the schematic a moment ago and this is where we choose the physical pin on the device. I'll assign the pin component to port 1, pin 6. That's the LED on the PSoC 4 Pioneer Kit.

The other tabs in the DWR include the Analog tab; that shows you the internal analog routing resources. This design does not have any analog, so it is mostly grayed out. But you could use this viewer to lock in the analog routes and determine their parasitic resistance, an important function in analog design.

Next, there is the Clock editor where you can set up the clock sources for the ARM CPU and the clock components in your schematic. You can double click on any of the yellow lines to launch the clock editor dialogue. This is where you set up the internal and external clock sources that are used to drive the high frequency clock and the low frequency clock sources.

The Interrupts tab lets you pick the priority level for all of the interrupts you put into your schematic.

The System tab lets you choose things like the heap and the stack size, the debug options, and the operating conditions for your application.

The Directives tab lets you force a component into a specific resource in the device. It's not often required, but sometimes you might have a very timing sensitive design that requires a particular routing solution.

Lastly, there is the Flash Security tab which lets you protect your IP by disabling read-access to sections of the flash memory.

The third file is main.c which is the template for your c-code. In this file I write the firmware to drive the LED. Once the project is built, a lot more files are created. This includes the component files that are placed in the generated sources folder. These files contain the automatically generated API's to support the pin that you placed. Note that the file names correspond to the name you gave the component.

The last section in the workplace explorer is a results tab. After you build a design the back-end tools like the fitter and the compiler generate some more useful files. In this tab you will see the compiler listings file, the linker map, and the fitter report file. The report file is important as it shows you all of the resources you used in your design and is a great way to access how much head room you have as your application grows.

That's a quick look through the project files in PSoC Creator. Take a little time to look through these files for yourself and soon you'll get a better handle on what's going on inside the genius that is PSoC Creator.