

Hardware Pins

My name is Alan Hawse and this is PSoC 101. In this is lesson I am going to give you a glimpse at some of the unique aspects of PSoC and prove to you how easy it is to use. I am going to tie pins together inside the device and control the LEDs from the switch without any intervention from the ARM core at all. This functionality is completely unique to Cypress PSoC.

First, make a copy of the last project. Delete the interrupt and disable the interrupt terminal from the customizer. Also, check the hardware connection box to expose a new terminal on the input pin. In the red LED pin I also check the hardware connection and then I simply draw a wire between the two.

In main.c, I will just delete all the old code. Now I'll build and program the board to prove that the switch is now controlling the LED without any firmware at all. There is no microcontroller on the planet that allows you to do this. I can also extend my design to drive two of the pins, the red and the green at the same time so that it produces a yellow color. Don't forget to expose the HW connection on the second pin component.

As simple as this example is, you can already see that with PSoC, it is easy to control multiple hardware blocks, immediately, with a single signal inside the device. There is no latency caused by API calls to turn peripherals on and off – they can be gated in perfect synchronization. As we do more of these lessons you'll start to see many simple, energy-efficient and elegant solutions to problems that can't be solved by traditional MCU-based designs.

Please reproduce this design and then add a not gate to one of the LEDs so that the red and green pins go on and off in sequence. As a hint, make sure you use the internal NOT gate component, not the external inverter, which is blue in color and generates errors when connected to internal components.

As always you are welcome to email me at alan_hawse@cypress.com.