

Project 1 Report

Project description:

In Project 1, we modified a given set of code to create an application that had some text on the LCD display, read the inputs from buttons on the LaunchPad and BoosterPack. We also had to send output signals to the LEDs on both the BoosterPack and the LaunchPad based on the input signals read from the buttons. The final product should be almost a game where the LED on the BoosterPack should match the Right LED on the LaunchPad by pressing certain buttons, or a combination of buttons, indicated by the text on the LCD display. When the correct buttons are pressed and the LEDs match, then the LED on the LaunchPad should switch to the next color in the cycle and repeat the process. The expected cycle of the LaunchPad LED colors are Red->Blue->Green->Purple->Orange->White and then restart from Red.

Hardware Abstraction Layer:

The advantage of using HAL in the code is that it is much easier to follow the flow of the code. By using HAL functions, understanding what each function does and how it is done is very easy and is much more efficient. The HAL coding style makes it much more portable, as only certain hidden functions will have to be adjusted, while the *main* function can be left untouched no matter what board it is used on.

Driverlib functions versus manipulating registers:

Driverlib functions appear to be very similar to using the memory mapped registers. An advantage with using the Driverlib functions is it makes the initializations shorter in line count. By having functions that directly state whether there is a pull up, pull down, or no resistor used it is slightly easier to understand what is happening in the code.

On the other hand, manipulating registers directly is much more specific to a certain board type. By working directly with the board's memory-mapped registers, each part of the process/code can be much more carefully tracked making it easier to see how each piece of the code works with the other.