Demetris Coleman

The objectives of lab 1 were to familiarize the students with the uVision software and the debugger for the Discover Board microcontroller. My team managed to learn the software well enough to create our first project, compile the code provided, and find what the warning and error messages meant. The first compile gave no errors and one warning. The warning pertained to a variable that was set but not used. This brought the fact that memory was limited to my attention.

The team also became familiar with the debugger and many of its capabilities. These include the ability to see the instructions carried out by the microcontroller in c and assembly, breakpoint functionality, the watch window and the call stack window. In the debugger, if you clicked on a line in the C program, the corresponding microcontroller instructions were highlighted. For instance, when we clicked on a line that stored a value into a variable, you saw move, load, and store instructions in the assembly box. The call stack window showed the type and values of all variables in the program, but the watch window allowed you to single out particular variable to watch. The debugger also had breakpoints that stop the program right before the line with the breakpoint is executed. After stopping at a breakpoint, the debugger you allowed you to execute a single line of code using the single step command or to execute a block of code such as a function by using the step over command.

The function of the program seemed to be a simple toggle with a delay. If you held down the user button, LED 1 would blink with a delay. If you just pressed the button, the button would toggle or switch to the opposite binary state after a short delay.

By the end of the lab, I learned to use the uVision software, to use many of the debugger’s functions, and that the microcontroller does not have as much memory as a computer. Other lessons taken from this lab were to make sure you always have your boards connected and on when trying to run a program, to make sure the correct debugger is selected, and how to write hexadecimal and binary values in C.