ECE 6740 Advanced Embedded Systems _____
Prof. D. Hanna **Name**
Winter 2024

LAB # 1: Hello Galaxy

In this lab we will become familiar with using the Vivado design suite to generate a system block with the ZYNQ-7 core, integrate custom IP, and write basic software to interface the ZYNQ-7 core with the custom IP through memory-mapped registers. We will use the UART for our *stdio*.

PART A

Create a new Vivado project and create a block design with the ZYNQ-7 core and the UART enabled to use for *stdio*. Disable the timers and other items that we will not be using in this lab. Synthesize and Implement this design. Export the design to the SDK. Create a board support package (BSP) in the SDK and sample 'Hello World' application. Modify the 'Hello World' application to perform the following calculation and print the *answer* to the screen (using the UART as *stdio*, of course).

```
int a, b, answer;
a = 32;  b = 41;
answer = a + b;
```

Download the system to the board and test it using the built-in terminal program.

PART B

Create a new Vivado project and create an entity and architecture for a 2-input 32-bit adder. Don't worry about an input nor output carry. Create an AXI interface IP package with 4 interface registers (we will only use 3 registers) and an IP package for your adder component. Modify the AXI interface package to:

1. incorporate your adder where two of the interface registers will be input to the adder and the output from the adder will be input to the third interface register.
2. connect the adder output separately out of the AXI package so that it can be output to the LEDs on the general IO pins.

Import your adder/AXI IP into the project from part A. Connect it to the ZYNQ-7 core using an AXI interface. Connect the least significant bits of the adder's output to the LEDs. Synthesize and implement your design and export it to the SDK. Create a BSP and sample application. Modify the application to write the 32 and 41 values to the appropriate registers and read the answer back from the third memory-mapped register. Download the system to the board and

display the result on the screen through the UART *stdio*. Verify that the answer is correct and that the least significant bits are correctly output to the LEDs.

Submit the following items in a ZIP file when you are finished:

1. A copy of this lab instruction sheet
2. A listing of the parts of the IP AXI interface that you modified
3. A listing of your main c-file
4. A screenshot of the results from Part B
5. A short narrative on challenges that you encountered and how you overcame them –or– [better yet] a posting to the tips forum on Moodle.