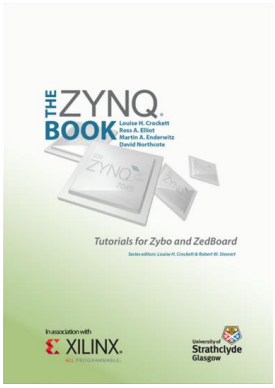
ECE3623 Embedded System Design Laboratory

Dennis Silage, PhD
silage@temple.edu

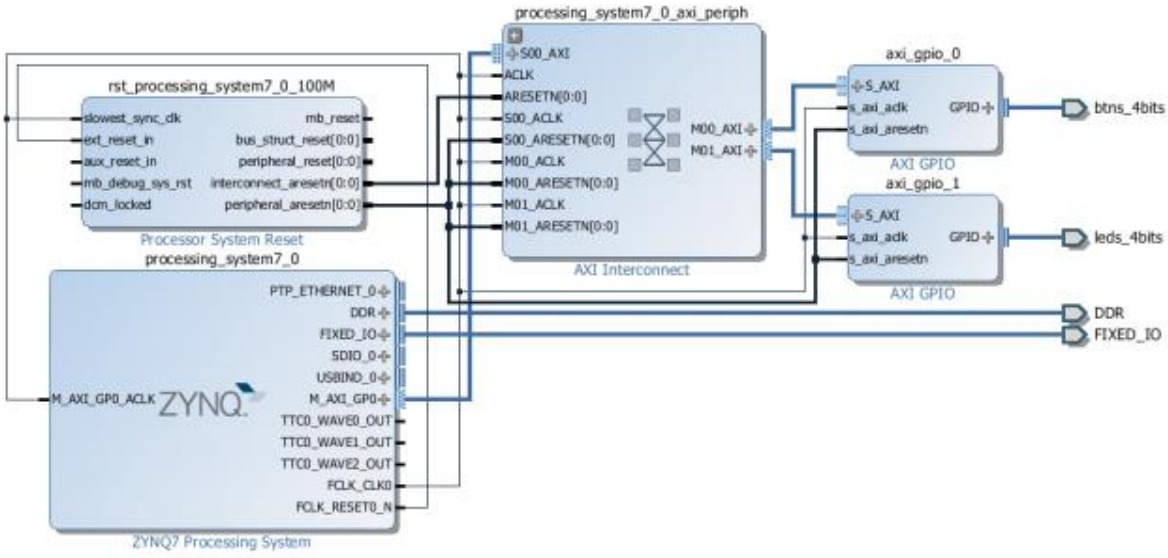
Vivado SDK Basic IP Integrator

In this Laboratory you will be introduced to the embedded development of a Vivado Zynq Processor System (PS) with a basic IP integrator on the Zybo Board. The initial task and references are the eText *The Zynq Book Tutorials* with the supporting files in the *The Zynq Book Tutorials Sources* both of which are posted on *Canvas* and the Lecture PowerPoints.



The Laboratory requires you to generate the Vivado hardware and modify the SDK software for a new process. Exercises 1A, 1B and 1C result in a project that uses the LEDs with a single GPIO interface. As a reference Exercise 2A and 2B add a second GPIO that inputs the push buttons. But here you are *not* to add *interrupts* as described in Exercise 2B.

A new project *lab1.prj* with two GPIO interfaces is to be generated with the basic Vivado hardware design shown below from the start of Exercise 2B.



Once the Vivado hardware design is verified, SDK would import the template program *LED_test_tut_1C.c* which you are to modify and name as *lab1.c* to input

the push buttons on the second GPIO interface and implement the task. The count delay remains the same. The modification includes integrating the *#define* and setup from sections of the program *interrupt_controller_tut_2B.c* and into your *lab1.c.* The sections are shown below:

```
#define BTNS_DEVICE_ID          XPAR_AXI_GPIO_0_DEVICE_ID
#define LEDS_DEVICE_ID          XPAR_AXI_GPIO_1_DEVICE_ID
```

```
int status;
//-----------------------------------------------------
// INITIALIZE THE PERIPHERALS & SET DIRECTIONS OF GPIO
//-----------------------------------------------------
// Initialize LEDs
status = XGpio_Initialize(&LEDInst, LEDS_DEVICE_ID);
if (status != XST_SUCCESS) return XST_FAILURE;
// Initialize Push Buttons
status = XGpio_Initialize(&BTNInst, BTNS_DEVICE_ID);
if (status != XST_SUCCESS) return XST_FAILURE;
// Set LEDs direction to outputs
XGpio_SetDataDirection(&LEDInst, 1, 0x00);
// Set buttons direction to inputs
XGpio_SetDataDirection(&BTNInst, 1, 0xFF);
```

Engineering projects are often stated as procedures which must be rendered as process control software on the appropriate embedded hardware. You are to reconfigure the new project file *lab1.c* to perform the specified tasks as follows.

1. The rightmost push button 0 is the system RESET and when *released*, regardless of any other push button being depressed, the LED count starts at −8 (1000) and increases as 4-bit *signed binary* with roll over (that is, −8,−7, −6, −5, −4, −3, −1, 0, 1, 2, 3, 4, 5, 6, 7, −8, −7…)

2. If push button 1 is depressed *singly* with no other push button being depressed, all further count and LED operations are *suspended* (LED count is fixed). When push button 1 is *released* the LED count starts at 7 (0111) and decreases as 4-bit *signed binary* with roll over (that is, 7, 6, 5, 4, 3, 2, 1, 0, −1, −2, −3, −4, −5, −6, −7, −8, 7, 6…).

3. If push button 2 is depressed *singly* with no other push button being depressed, all further count and LED operations are *suspended* (LED count is fixed). When push button 2 is *released* the LED count is set to 0 (0000) and increases as an increasing and decreasing, repeating *bar graph* (that is, 0000, 0001, 0011, 0111, 1111, 0111, 0011, 0001, 0000, 0001, 0011…).

4. If the leftmost push button 3 is pressed *singly* with no other push button being depressed, all further count and LED operations are *suspended* (LED

count is fixed). When push button 3 is *released* the LED count is set to 1001 and is a repeating *pattern* (that is, 1001, 0110, 1010, 0101, 1100, 0011, 1001, 0110, …).

You are to *describe in detail* and *list the modifications* to the basic template program *LED_test_tut_1C.c* to accomplish this Laboratory task. Verify the performance of the Laboratory task and be prepared to demonstrate and describe the project to the Laboratory Assistant.

These completed Laboratories should be archived on your laptop and will form the basis of the Exams.

You are to use the *Project Report Format* posted on *Canvas*. The *Laboratory Procedures* document on *Canvas* is the guideline for performance. You are to upload your *Report* to Canvas (but *not* the project) for time and date stamping to avoid a late penalty.

This Laboratory starts on Tuesday January 26th and is due no later than Tuesday 11:59 PM February 2nd.