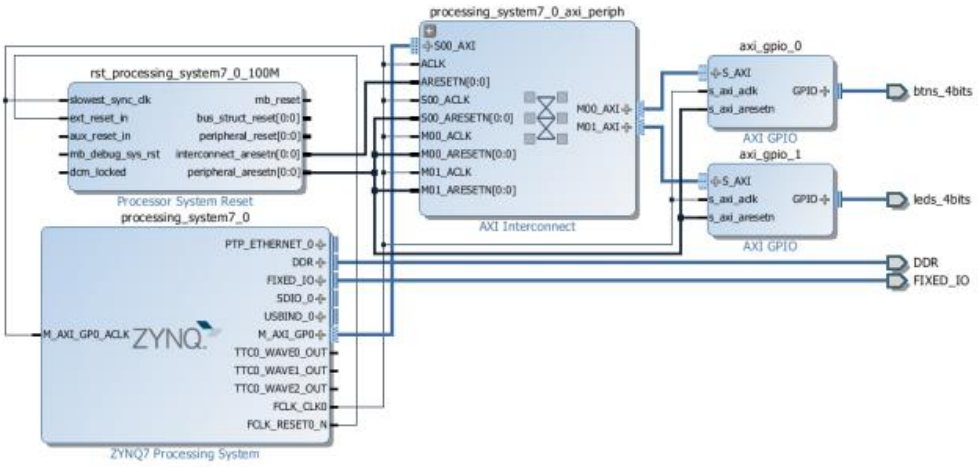
ECE3623 Embedded System Design Laboratory

Dennis Silage, PhD
silage@temple.edu

FreeRTOS Data Queue on Zybo v2

In this Lab 5 you will investigate the implementation of a FreeRTOS data *queue* on the Zybo board. In two previous Laboratories you created a Xilinx Vivado block design with two axi_gpio IP modules: axi_gpio_0 for the Zybo push buttons and axi_GPIO_1 for the Zybo LEDs as shown below. In Lab 2 the operating system was multitasking with FreeRTOS but did not pass data between the tasks. Lab 5 here will use the FreeRTOS *queue* function to pass data.



However, the GPIO at *XPAR_AXI_GPIO_0_DEVICE_ID* now is for the BTNs on channel 1 and the SWs on channel 2. The LEDs GPIO is at *XPAR_AXI_GPIO_1_DEVICE_ID*. SDK was imported with the hardware specifications from Vivado in the previous Laboratories. If required, complete the Vivaldi hardware designs and launch SDK.

In SDK do *File…New…Application Project* then enter a new project name and select *freertos10_xilinx* as the OS Platform. Not *Finish* but *Next* to continue. In the *New Project…Templates* select *FreeRTOS Hello World* and *Finish* to continue. The *Project Explorer* shows the *freertos_hello_world.c* as the *main()* with the tasks required for the FreeRTOS OS and will be used as a template for this Laboratory.

Using the *freertos_hello_world.c* file as a template the Laboratory task is to have a new *main*() application (using another name) that performs the following FreeRTOS task and queue management operations. Note that the *freertos_hello_world.c* has a queue

functionality but the timer function is not used as yet. The specified delays in the tasks are to be implemented by an appropriate delay loop.

1. Create a FreeRTOS main() application that creates two tasks and a queue. The queue can hold five entries each large enough to encode which button is depressed and switch is ON as a single data entry (see below) The FreeRTOS functions *XQueueCreate()*, *XQueueReceive()* and *XQueueSend()* utilizes a handle *xQueueBtnSw.* Data is sent to the queue only if any BTN or BTNs are depressed first and the SWs are then encoded in the data (see below).

2. Create a FreeRTOS task *TaskLED* at the Idle task priority+ 2 that receives a combined, encoded button and switch value (see below) from the queue *xQueueBtnSw* every 5 seconds using an appropriate loop delay and a block time of 60 seconds for the receive queue.

The appropriate LED or LEDs is to light for approximately 2 seconds if the same decode BTN is depressed and the same decoded switch is ON. All other BTN and SW combinations are to be ignored. For examples, if BTN0 and BTN1 are both depressed and SW0 and SW1 are ON then LED0 and LED1 are both ON for 2 seconds. If the decoded BTN and SW value indicates obtained from the queue indicates that there are no valid BTN and SW combinations, then all the LEDs are OFF.

An error condition of empty receive queue should flash all the LEDs for 1 second on and 1 second off until the error is resolved.

3. Create a FreeRTOS task *TaskBTNSW* that reads the BTNs and SWs at the Idle task priority+ 1. Encode BTN0 as 1 and SW0 as 16, BTN1 as 2 and SW1 as 32, BTN2 as 4 and SW2 as 64 and BTN3 as 8 and SW3 as 128. Combine the BTN and SW values as a single data entry for transmission via the queue to *TaskLED*. The block time for the transmit queue is 200 seconds.

- *The FreeRTOS timer (vTimerCallback) function in the original freertos_hello_world.c are not used in this Lab and should be removed.*

You should discuss and demonstrate *all possible conditions* of the task management with priorities and queue interactions for the *TaskBTN* and *TaskLED* including block times, queue empty and queue full in the Lab report.

The completed Laboratories should be archived on your laptop and will form the basis of Exams. You are to use the *Project Report Format* posted on *Canvas*.

You are to upload your *Report* to Canvas for time and date stamping to avoid a late penalty. This Laboratory is for the week starting March 2nd and due no later than 11:59 PM March 9th.