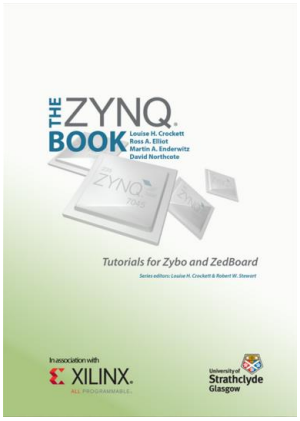
ECE3623 Embedded System Design Laboratory

Dennis Silage, PhD
silage@temple.edu

Vivado AXI Interrupt



In this Laboratory you will utilize the embedded development of a Vivado Zynq Processor System (PS) with a hardware interrupt in a standalone OS. The tasks are described in detailed in Chapter 2 of 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 study and initially execute the Exercises 2A, 2B and 2C.

The Laboratory tasks are as follows:

1. Run the complete *interrupt_controller_tut_2B.c* Vivado/SDK project without modification to verify its performance in a standalone OS.

2. Modify the original *interrupt_controller_tut_2B.c* project to perform the following tasks by configuring the *BTN_Intr_Handler* in SDK appropriately in a standalone OS:

   a) The LED count and display are initially at 0
   b) BTN0 upon an interrupt turns all the LEDs OFF with the LED count fixed at the last count value
   c) BTN1 upon an interrupt turns the LEDs ON with the last LED count value
   d) BTN2 upon an interrupt outputs to the LEDs with the 1-bit complement of the 4-bit LED count and the LED count is now at this new value
   e) BTN3 increments the LED count by 1 and output to the LEDs

   Describe in detail and list the modifications to the *BTN_Intr_Handler* to accomplish this Laboratory task. The LED count and display *rolls over* as 4-bit binary when incrementing.

3. Describe in detail the operation of the following lines of code in your AXI interrupt project. *Each and every term and operation of the function calls* must be adequately described in your Laboratory Report.

   a) static void BTN_Intr_Handler(void *baseaddr_p);
      static int InterruptSystemSetup(XScuGic *XScuGicInstancePtr);
      static int IntcInitFunction(u16 DeviceId, XGpio *GpioInstancePtr);

   b) if ((XGpio_InterruptGetStatus(&BTNInst) & BTN_INT) != BTN_INT)

   c) Xil_ExceptionRegisterHandler(XIL_EXCEPTION_ID_INT,
                                   (Xil_ExceptionHandler)XScuGic_InterruptHandler,
                                    XScuGicInstancePtr);

   d) IntcConfig = XScuGic_LookupConfig(DeviceId);
         status = XScuGic_CfgInitialize(&INTCInst, IntcConfig,
                                                   IntcConfig->CpuBaseAddress);

   e) status = XScuGic_Connect(&INTCInst, INTC_GPIO_INTERRUPT_ID,
                  (Xil_ExceptionHandler)BTN_Intr_Handler,
                  (void *)GpioInstancePtr);

   f) status = IntcInitFunction(INTC_DEVICE_ID, &BTNInst);

   g) XGpio_InterruptEnable(&BTNInst, BTN_INT);

   h) XGpio_InterruptGlobalEnable(&BTNInst);

You may be randomly asked to describe and demonstrate each of these tasks to the Laboratory Assistant at any point during the semester.

This Laboratory is for the week beginning of February 9th and due February 16th 11:59 PM with an upload to Canvas.