

CmpE Internship Summer 2018

Design Doc Template

Pushbutton Interrupt Controller

Authors & Editors:

Colin Schardt

Revision History:

|  |  |  |
| --- | --- | --- |
| Revision | Date | Description |
| A.1 | 6-11-18 | Initial Project Proposal |

Table of Contents

[Objective](#_f3ut0x3v9lxf) 2

[Background](#_52hjxc9p7ctt) 2

[Overview](#_yy2djj5i8omd) 3

[Detailed Design](#_avw48lx4cmyr) 4

[Caveats](#_op45shsa5n8y) 5

[Testing Plan](#_eejuf5av24ql) 5

[Unit testing scheme](#_h5m3fobrdlly) 5

[Integration Testing](#_fubwzje1thtf) 5

[Demonstration Project](#_6z1lojsl95dl) 5

# Objective

The objective of this project is to incorporate an input device and an output device in a way that is designed to teach future students about interrupts.

# Background

A large component of the new CMPE 127 curriculum focuses on hardware interrupts. Designing a lab in which students need to find a way to handle multiple inputs and outputs simultaneously is paramount to successful instruction of course material. The most recent iteration of CMPE 127 had a laboratory projects that dealt with input and output devices but dealt with interrupts in only a superficial or theoretical manner. As a result, many students were left with an incomplete comprehension of a concept at the end of the semester that they should have had functional knowledge of by the second midterm. Providing a lab project that requires students to design a circuit to handle interrupt signals and routines will enhance student understanding of these concepts.

# Overview

This project takes input from 4 pushbutton switches and provides output to a 96x64 RGB OLED display. The buttons will be assigned functions as follows: turn the display on/off, display red, display green, and display blue. To successfully complete this project, the 127-toolkit provided by Khalil Estell will need to be utilized in addition to an interrupt controller that must be designed. This controller must prioritize interrupt signals and determine the appropriate course of action to perform the desired task of the user.

Implementation and final results of this design are not yet finalized, but two variations have been planned. One variation allows for inputs from the push buttons to be combined (i.e. if the red and blue pushbuttons are pressed simultaneously, the display will output purple). The other variation of the design has each input superseding the previous (i.e. if the OLED currently displays red, pressing the pushbutton for blue will make the OLED display blue). Both variations of this design have the pushbutton designated to powering the OLED display as the highest priority signal.

This Project will be designed in Verilog using Vivado Design Suite on the Nexys 4 DDR FPGA board.

# 

# Detailed Design

This project will be written in Verilog for the Atrix-7 FPGA architecture on the Nexys 4 DDR board from Digilent. It will be written in Vivado Design Suite.

The 127-toolkit provided for this internship contains many useful modules ready for implementation. This includes a module for a VGA control. An addition module to control the pushbuttons will also need to be designed. Both of these modules will need to be utilized and possibly modified for the proposed design. The VGA controller will need to be used to interface with the OLED display, and the pushbutton controller will be needed in order to parse signals from the pushbuttons.

With these modules complete, the main aspect of the project can be designed. A module or several modules will need to be created that will take inputs and send outputs to and from the pushbuttons, OLED, and a processor. The input from the pushbuttons will be strobe signals; the inputs from the processor will be an interrupt acknowledged signal, a read signal, and a write signal; and the input from the VGA controller will be an acknowledge signal. The outputs to the pushbuttons will be input buffer full signals, the outputs to the processor will be an interrupt signal and a vector address, and the output to the OLED will be a write signal as well as a value to display. The final functionality of this module or modules should resemble both mode 2 of the Intel 8255 Programmable Peripheral Interface and basic functionality of the Intel 8259 Programmable Interrupt Controller.

This project will also require use of SRAM in order to retrieve interrupt service routines for the processor. This aspect of the project will need to be designed during its creation, but may be provided material to a future student, at the instructor’s discretion. For SRAM functionality, the Nexys 4 board will need to treat the onboard DDR RAM as if it were SRAM. A module will likely need to be designed to do so. Once this has been set up, a vector address will need to be created atg a location that can be reached by the 8259-like module. The addresses reached here will contain addresses where the corresponding interrupt service routines are stored so that the processor can carry out the appropriate actions for the display.

# 

# Caveats

* VGA control will likely need to be heavily modified
* SRAM functionality may be superficial, and not accurately reflect real-world functionality
* Most design decisions are very much in the preliminary stage

# Testing Plan

### Unit testing scheme

I will write self-checking testbenches for every module I create.

### Integration Testing

In addition to testbenches, I will set several milestones of functionality that I plan on implementing on the Nexys 4 board to be certain that certain systems worm both independently and with other modules.

### Demonstration Project

The final project will be implemented on the Nexys 4 DDR board and will include all necessary physical components in order to test and demonstrate functionality. There are a low number of input devices, so it should also be possible to demonstrate every possible combination of inputs to prove the desired and expected results if need be.