ES 4 Lab 4: Introduction to fast signals and FPGAs

Lab report due week of October 8 2023

1 Introduction

There are two goals for this lab: first, you should become comfortable using the oscilloscope on the lab bench to view signals too fast for your eyes (or multimeter) to capture. Second, you will work through the process of implementing an AND gate using VHDL and your FPGA, so that you're ready to tackle next week's lab.

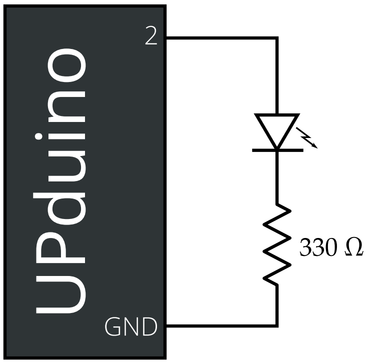
2 Analyzing circuit behavior with an oscilloscope

L1: Follow the video tutorial posted on the course website to get started using the oscilloscope. Because our current workspace has equipment from several different labs, your oscilloscope may look slightly different. However, the basic functionality will be the same.

L2: Get your note-taking setup handy — like lab 1, there is no lab report associated with this lab but we still expect you to take notes like the thoughtful and methodical engineers that you are!

L3: Connect an LED to pin 2 of your UPduino according to the schematic below, and connect the UPduino to a USB port on a computer. The LED should blink at 1 Hz.

WARNING: Make sure to disconnect the benchtop power supply from your breadboard before plugging in the UPduino. The UPduino will draw current from the USB port port to power the circuit, and external power is unnecessary. There are relatively few things that will destroy the UPduino, but incorrectly connecting it to an external power source is one of them.



> **L4**: Connect an LED to pin 47 of your UPduino. The LED should be flickering.

Obviously the LED is turning off and on, but exactly how fast is that happening? We're going to need a tool that can measure voltages at higher speeds than our eyeballs can see. Not coincidentally, we have tools in the lab to do that! In fact, we have *two* tools, *oscilloscopes* and *logic analyzers*. These have different strengths, summarized below:

	Oscilloscope (Tek TDS 2012)	**Logic analyzer (Digital Discovery)**
Measures	Analog voltages (0-200V)	Digital voltage levels (low/high) up to 5V
Simultaneous signals	2	40
Capture speed	Fast (25MHz)	Wicked fast (800MHz)
Interface	A bazillion buttons and knobs	Complicated PC program with a bazillion settings

For now we'll be using the oscilloscope, but you'll have an opportunity to use the logic analyzer later in the course when we need to monitor many signals at once.

> **L5**: How fast is the LED blinking? Draw the waveform (i.e., the voltage vs time graph), making sure to label the relevant units.

> **L6**: Now connect an LED to pin 48 of your UPduino. The light should be on (and not blinking), but you might notice the light is dimmer than when you used the switch. Examine the voltage on the diode using the oscilloscope, and document what you see.

> **L7**: Connect an LED to pin 4 of your UPduino. It should be on, but dimmer than the previous one. Record a hypothesis about what is happening, and then test it with the oscilloscope. Make some notes about what you discover.

> **L8**: Connect an LED to pin 6 of your UPduino. It should be on (and not blinking). Examine it with the oscilloscope, and make notes about what you find.

> **L9**: Connect an LED to pin 11 of your UPduino, and observe the blinking behavior of the LED for a few seconds. Again, write down a hypothesis about what is happening, and then examine what is really going on using the oscilloscope. Record your observations.

> **L10**: (Optional) If you'd like more practice, you can test the following pins:

Pin	Expected signal
28	500 Hz
42	1 kHz, 80% duty cycle
43	10 kHz, 2% duty cycle
37	2MHz

3 Flashing your own design on the FPGA

Follow the tutorial on the course website to work through the process of creating your own design in VHDL and flashing it to the FPGA using Lattice Radiant.

If you'd like to install Radiant on your own computer, you can download it for free from Lattice. You'll need to register for a free license; you put in your email and they send you the license file a few minutes later.

For the Linux nerds / open-source zealots / command-line junkies: there is an open-source toolchain for the iCE40, which you can install via the oss-cad-suite package or using apio. Given our limited staff resources, we cannot officially support these tools, but you're probably used to figuring this stuff out on your own anyway! Get in touch with TA Lucas Polidori if you have questions.

4 What to turn in

There is no formal lab report to turn in, just submit the notes you took along the way.