

## Lab 1: Bootstrap your embedded system

Charlie Coleman

To familiarize myself with the Digilent Zybo, I followed a tutorial found here:

`https://reference.digilentinc.com/learn/programmable-logic/tutorials/zybo-getting-started-with-zynq/start`

This guide walks through creating a new project in Xilinx Vivado, designing a block diagram that exposes the switches/buttons/LEDs so the processor side, generating bit files, and using the SDK to program the processor side. All of this combined together allows us to toggle the LEDs using the corresponding switch and allows us to print to the Xilinx console when a button is pressed. The steps outlined in the guide are:

1. Create New Project
2. Create a new block design using the ZYNQ7 Processing System & 2× AXI GPIO core
  - (a) For AXI GPIO Core 0, enable Dual Channel
  - (b) Run the Connection Automation Tool for all automation
  - (c) For axi\_gpio\_0, select GPIO & set to btns\_4bits
  - (d) Select GPIO2 & set to sws\_4bits
  - (e) For axi\_gpio\_1, select GPIO & set to leds\_4bits
3. Generate HDL Wrapper, Validate Design, Generate Bitstream
4. File → Export → Export Hardware... & include Bitstream
5. File → Launch SDK, Create new Application Project
6. In helloworld.c, replace the code with the one given in the tutorial.
7. Program FPGA & Run & Test

Currently, I am planning on using the Zybo without an operating system. This may help with DMA required to write/read the frame buffer fast enough for the VGA output. It will also require some testing/research to find out how to get the Asteroids program to start at boot up.