ENGR 3410: Miniproject 4

due December 4, 2025

In this miniproject, you will use the OSS CAD suite to design an unpipelined, multicycle (i.e., the processor is only working on executing one instruction at a time), 32-bit RISC-V integer microprocessor with a Harvard architecture (i.e., the processor has separate instruction and data memories that share a unified address space). You will be supplied with a memory module that implements 8kB of physical memory at the bottom of the 32-bit address space (4kB of data memory from 0x0000 to 0x0FFF and 4kB of instruction memory from 0x1000 to 0x1FFF) in addition to several memory-mapped hardware peripherals that are accessible through the data memory ports at the highest addresses. These include 8-bit PWM generators for the user LED and the RGB LEDs on the iceBlinkPico board and two running 32-bit timers that respectively count the number of milliseeconds and microseconds (mod 2^{32}) that have elapsed since the processor started. Your task in this miniproject is to implement the base RV32I instruction set (minus the `ecall`, `ebreak`, and the atomic read/write instructions) that is capable of running a simple RISC-V assembly-language program that blinks the LEDs through the provided memory-mapped peripherals in an interesting manner.

This miniproject is a *group* project that can be done in teams of up to three students. You can discuss design approaches between groups and help each other with learning SystemVerilog and how to use the OSS CAD suite, but each team must complete all aspects of this assignment *without sharing code between groups*.

Requirements. Your design must meet the following requirements:

1. Your processor must implement all of the instructions in the base RV32I instruction set except for `ecall`, `ebreak`, `csrrw`, `csrrs`, `csrrc`, `csrrwi`, `csrrsi`, and `csrrci`.
2. Your processor must be specified in one or more SystemVerilog source files.
3. You must provide a SystemVerilog (or other) test bench and simulation results using Icarus Verilog (iverilog) verifying proper operation of your processor for representative examples of each class of RV32I instructions.

Deliverables. By the start of class on December 4, you must submit the following items to the course Canvas site:

1. A PDF file containing a brief report explaining the design of your processor and its operation. You should include simulation results demonstrating the proper operation of your processor for representative examples of each class of RV32I instructions.
2. Copies of all of the source files specifying your circuit as well as your test bench. You may provide the URL of a Github repo or a shared folder containing your source files.