ECE 201 Spring 2022 - Project 1

Assigned by Prof. Wujie Wen

Due Feb. 17th, 2022 23:59pm

Instructions:

- Please run the nclaunch based on the tutorial before you work on project1. The purpose of project 1 is to help you familiarize with the nclaunch.
- Run all your experiments under the folder: **/proj/ece201-spring2022/XXX** (XXX is your username), **DO NOT** run in your home directory /home/XXX.
- Keep all your source codes in the workstation, and include your running directory in the submitted report (e.g. **/proj/ece201-spring2022/XXX/Proj1)**. TA and instructors will go to your running directory and conduct simulations for grading purposes.
- Submit your report (pdf format) along with source codes (zip) to coursesite.

1. Design two Verilog code implementations for a simple 3 to 8 instruction decoder: 1) **Behavior level**, which only requires one Verilog module; 2) **Gate level** which consists of multiple low-level Verilog basic modules, such as AND, OR, NOT, XOR etc. You need to define these basic modules and then properly instantiate them to realize the function of 3 to 8 instruction decoder. These basic modules can have an input with multiple bits, e.g. two or three inputs AND gate: a&b or a&b&c. The input and output signals of the 3-to-8 decoder are defined as follows. Here Bi (i=0,…,7) can represent different instructions, e.g. ADD, JMP etc.

   Input: A[2:0]

   Output: B0, B1, B2, B3, B4, B5, B6, B7

   The decoder is based on one-hot coding, for example, if A=2'b00, B0=1, B1=0, B2=0, B3=0, B4=0, B5=0, B6=0, B7=0

   For the two implementations: (a) Draw your design schematic; (b) Write your Verilog code and the testbench; (c) Simulate and validate them using nclaunch.
   All design structures, source codes and simulated waveforms should be included in your report.

2. Design a Verilog decoder to decode LEGv8 R-format, I-format, and B-format instructions only (see Fig. 2.20 in the textbook). Note you can assume the input is a 32-bit bus, you should use its top bits as Opcode to decode the R-format, I-format, and B-format instructions. For example, Bus [31:21] (11 bits) and Bus[31:26] (6 bits) to decode R- format and B-format instructions, respectively.
   Opcode Input: Bus[31:0]
   Output: AND, ADD, ORR, ADDS, EOR, SUB, LSR, LSL, BR, ANDS, SUBS,
   ORRI, EORI, ADDI, ANDI, ADDIS, SUBI, SUBIS, ANDIS,
   B, BL;
   For example, Bus[31:21]=11'b10001011000 will lead to output ADD=1, which implies the R-format instruction-ADD, and an adder will be used to execute the instruction.

   (a) Write your Verilog code and the testbench; (b) Simulate and validate them using nclaunch.
   All design schematic, source code and simulated waveform should be included in your report.