Lab 1

- Write a checker for the simple ALU in Lab0 that checks the following rules:
  1. when reset_n is asserted (driven to 0), all outputs become 0 within 1 clock cycle.
  2. when opcode_valid is asserted, valid opcode and valid data (no X or Z) must be driven on the same cycle.
  3. Output "done" must be asserted within 2 cycles after both valid data have been captured.
  4. Once "done' is asserted, output "result" must be correct on the same cycle.
  5. Once "done' is asserted, output "overflow" must be correct on the same cycle.

Due: Saturday (4/18/2015) by 11:59pm (Midnight)

Lab 1

- **Deliverables:**
  - – Checker code must be in a separate module (alu_chkr)
  - – Code your checker rules separately so that individual checker rules can be enabled and disabled from the test environment.
  - – Instantiate and connect the checker at the top level.
  - – Compile and simulate and see if the checker flags any error.
  - – Include a block diagram showing all top level connections.
  - – Document any false failure and/or checker bug you discover in the process.
  - – Add separate bugs in the DUT and show the transcript that your checker flags errors for rule#1, 4 & 5. Document where you introduced your bugs in the DUT.
  - – Add a bug in the test generator and show the transcript that your checker flags errors for rule#2. Document where you introduced your bug in the test generator.
  - – For each of the above cases, see if you can disable the checker rule and get your simulation to "pass"
  - – Bonus: Introduce a bug in the DUT and see if you can get your checker to flag an error for rule#3.

Guidelines

- Should be based on Lab0 you just finished.
- Reuse as much as possible from Lab0.
- Use pure Verilog features. Do not use SV assertion. Will use it in Lab2.
- You can use the same DUT that you created.
- Or, you can use the lab0 solution, once I post it.
- Use the new test module. You can edit it for debug purpose but your final simulation must have my test module.
- You can reuse the top level testbench. Just instantiate your new checker there.