Course Overview

• One big project
  – Design a processor and toolchain that implements an ISA of your own design.

• 8 Labs
  – Roughly one per week.
  – Due on Friday before class.

• Work in groups of 2 (after Lab 1)
  – No “divorces” allowed
  – Make sure you pick someone you can work with
  – You will both receive the same grade

• Watch the website and read the Announce form on the web board.
This course is a lot of work.

- “2 units” has more to do with how I’m compensated than with how much work it is.
- You will learn an enormous amount
- You will design a (multi)processor.
- You will spend a lot of hours in the lab.
- Plan accordingly.
Grading

• 85% Labs
  – Do not fall behind, since the labs build on one another

• 15% web board
  – Web board is mostly for tool/Verilog support
  – Post 1-2 times per week (questions or answers)
  – You can post illustrative code snippets but not big pieces of your project.
  – Use common sense
Collaboration

• You can discuss your project with other teams
• You can (and should) discuss tool issues with other teams
• You cannot share code
• Be familiar with UCSD policies about academic honesty, cheating, etc.
• If in doubt, ask me.
Project overview

- There are four parts
  1. Xilinx tools orientation (Lab 1)
  2. Fetch unit (Labs 2-3)
  3. ISA design and assembler (Lab 4-5)
  4. Execution unit/cool tricks (Lab 6-8)
- There will be an interview after each part.
  - Discuss with Raid and I what you did, etc.
- Preview of the labs to come:
  - The last 141L
  - http://www-cse.ucsd.edu/classes/sp07/cse141L/
Part 1: Xilinx tool orientation

• Xilinx tutorial
  – Entering, compiling, simulating, measuring a design.
  – We will be using the free Xilinx verilog/FPGA tools
  – Instructions for installing them are linked off the web page
  – These tools are the best available, but they are still idiosyncratic.

• Web board set up
  – Sign up. Post a hello message
Part 2: Implementing instruction fetch

• Lab 2 -- Fetch unit datapath
  – Implement the datapath for the fetch unit
  – We provide some scaffolding

• Lab 3 -- Fetch unit control
  – Implement the control for the fetch unit.
  – Combine it with the datapath.

• Goals: Datapath/control design discipline; verilog and Xilinx practice; learn how fetch units work.
Part 3: ISA and toolchain

- Lab 4: Design your own ISA
  - A set of target benchmarks
  - Be creative. We’ve constrained the design space to keep things interesting.
- Lab 5: Build an assembler
  - An ISA is nothing without an assembler.
  - This will make programming your processor tractable
Part 4: Build it!

• Lab 6: Back end data path
  – The hardware to implement your ISA
• Lab 7: Back end control
  – Orchestrate the ballet that is your processor.
• Lab 8: Optimize and debug
  – The sky’s the limit
  – Maybe build a multiprocessors!!!
Staff

• Professor: Steven Swanson
  – General architecture questions, lab questions, class policy questions.
  – Contact info at http://www.cse.ucsd.edu/users/swanson/

• TA: Raid Ayoub
  – Main contact for lab-related stuff
  – Office hours TBA in the lab

• TA: Joe Auricchio
  – Xilinx tools questions
  – Lab questions
  – Office hours TBA
About me

• Stuff I built: WaveScalar architecture, compiler, verilog model, and emulation platform