CSE 141L: Introduction to Computer Architecture Lab
Implementing

Pat Pannuto, UC San Diego
ppannuto@ucsd.edu
Milestone 1 is due in 7 days

• What to submit?
  – SOMETHING

• M1 is graded for completion, not accuracy
  – The purpose of milestones is to help you manage large, long-term project
  – TAs will use gradescope “grades” to help give feedback
  – Recall: Only Milestone 4 (final submission) is actual grade*
    • *With exceptions for things such as skipping milestones altogether
Today’s Objectives:
What does implementing look like?

• Some tips for setting up CloudLabs

• Exploring the basic_proc example

• Exploring ModelSim/Questa and Quartus

n.b. the rest of these slides not presented, but here as a reference
The code editors in ModelSim/Questa/Quartus aren’t great

• You can use them, but more modern editors have some helpful tools
• Especially around version control
  – Because you are using version control, right?
    • And I don’t mean copying folders milestone1, m1_working, m1_real, m1_final, …
CloudLabs — What’s ephemeral and what’s not

• Spins up a ‘fresh machine’ each login
• But you have a shared L:/ drive that persists across sessions
  – So put stuff there
  – But also maybe don’t trust it 100%
    • [What fixes this? Version control of course!]
• Take advantage of “portable apps”
  – https://code.visualstudio.com/docs/editor/portable
  – https://git-scm.com/download/win – Choose “portable” and install in L:/
Portable VSCode

- Download the **zip**, not the installer
- Unzip the folder in your L:/ drive, and make a folder called data
If using portable apps, have to point VSCode to git

- File->Preferences->Settings; search `git-path`; write this json:

```json
{
  "git.enabled": true,
  "git.path": "L:\YOUR_FOLDER_NAME\bin\git.exe"
}
```
Quick reminder: No public repos please

• Q: What’s really clear and indisputable evidence of sharing code?
• A: A publicly accessible repository
The “github” built-in won’t find private repos, so you have to type the URL by hand

• You can use the default ‘credential helper’ (“manager-core”); authorize it to github; and then it will clone
Now you’re set up to code!

```verilog
module InstROM #(parameter A=10, W=9) (
    input [A-1:0] InstAddress,
    output logic[W-1:0] InstOut);
```
Careful, you (can) have two editors open
[Demo Plan]

- Create new project from basic_proc
- Compile
  - Show warnings; explore ALU 1 vs 1'b1
- Go to waveform viewer
  - Show results
  - Work examples / take Q’s