Control Path Design and Lab 3
Signals controlling the datapath  Signals providing information to control

Data inputs

Datapath

Control inputs

Control

clk  reset

clk  reset

Outputs

control outputs
Designing the Control Path

• Identify control lines
  • Inputs from datapath.
  • Outputs to datapath.
• Figure out how the output lines should be set for each instruction.
• Implement!
Identifying Outputs

- Any element of the data path that has a control input
  - muxes
  - alus
  - enabled registers
- Outputs to the outside world that are not data
  - Data valid lines
- If you designed your datapath correctly, this should be all of them.
Identify the Inputs

• These are all the bits of information that the datapath generates to allow your design to make decisions.
• If you thought through your datapath carefully, you should already know what these are.
Computing Outputs

- For this lab, the control unit is a combinational block that computes the outputs from inputs.
- That’s all!
- Arithmetic; R-Type
  - $\text{Inst} = \text{Mem}[\text{PC}]$
  - $\text{REG}[\text{rd}] = \text{REG}[\text{rs}] \, \text{op} \, \text{REG}[\text{rt}]$
  - $\text{PC} = \text{PC} + 4$

<table>
<thead>
<tr>
<th>Inst</th>
<th>write_reg_mux_sel_out</th>
<th>operand_B_mux_sel_out</th>
<th>reg_write_data_mux_sel_out</th>
<th>reg_file_write_en_out</th>
<th>dmem_write_en_out</th>
<th>dmem_read_en_out</th>
<th>dmem_size_o</th>
<th>alu_func_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td>100000</td>
</tr>
</tbody>
</table>
• **ADDI; I-Type**
  - \( \text{PC} = \text{PC} + 4 \)
  - \( \text{REG}[rt] = \text{REG}[rs] \text{ op SignExtImm} \)
Also today:

- Lab Overview
- Test app overview
- Debugging examples
- Example output for both tests