Agenda

- Datapath
- Pipelining
- Hazards
Datapath
Full RISC-V Datapath
Full RISC-V Datapath: add
Full RISC-V Datapath: sw exercise
Full RISC-V Datapath: beq exercise
Pipelining
Full RISC-V Datapath: Stages
Why Pipeline?

Single Cycle

Pipeined
Converting To Pipelined Design
Hazards
Hazards

- **Structural hazard**
  - A single resource is required by more than one instructions
  - We have solved some structural hazard! (e.g. RegFile with 2 reading port)
- **Data hazard**
  - One (or more) source register is not ready when being used
  - Can be solved by forwarding (trade complexity for performance)
- **Control hazard**
  - For branch instructions, we cannot know if it is taken at next cycle
  - branch prediction
Data Hazard - Stall

- Consider a 5-stage pipeline:
  
  \[
  \text{add}(x_3, x_1, x_2) \\
  \text{sub}(x_4, x_1, x_2) \\
  \text{xor}(x_5, x_1, x_3) \\
  \text{or}(x_6, x_2, x_5)
  \]
Data Hazard - Stall

• Consider a 5-stage pipeline:

\[
\begin{align*}
\text{add} & \ x3, \ x1, \ x2 \\
\text{sub} & \ x4, \ x1, \ x2 \\
\text{xor} & \ x5, \ x1, \ x3 \\
\text{or} & \ x6, \ x2, \ x5
\end{align*}
\]

\[\frac{11 \text{ cycles}}{4}\]

\[\sim 2.75 \text{ cycles/inst}\]

\[\text{ppl-ex} \leq \text{ppl-id}\]

\[\text{ppl-ex} \leq \text{ppl-ex}\]
Control Hazard

How many missed cycles on a mispredict?

Instruction Fetch (F)
Instruction Decode/ Register Read (D)
ALU Execute (X)
Memory Access (M)
WB
Control Hazard

How many missed cycles on a mispredict?
Questions?