

Digital Design & Computer Arch.

Lab 9 Supplement: The Performance of MIPS

Prof. Onur Mutlu

ETH Zurich

Spring 2021

18 May 2021

What we have done so far

- You have learned how **gates** work and how to build **small modules** out of them.
- You combined those **modules** into **more powerful** ones.
- You learned how to **test code** via **simulation** and **test benches**.
- You learned how to **write MIPS assembly code**.
- **And finally:** You combined all of that knowledge to build a **simple processor** and even ran your own programs on it.

Lab 9 Overview

- Extend the processor from the previous lab: `srl`, `mflo`, `multu`

- Measure the `performance` of the `processor`.

Lab 9: New Instructions

- The processor can already **add** and **subtract** but to be able to efficiently compute we want to add **multiplication**.
- By **shifting** we can **multiply** and **divide** by **powers of 2**.
- You are only required to implement a **logic right shift (srl)**
 - but feel free to implement a **arithmetic right shift** and **logic left shift**.
- Every left shift can be implemented by a normal multiplication so we don't need it.

Lab 9: How to do 32-Bit Multiplication

- The result of a multiplication of **two 32-bit numbers** can have up to **64 bits**.

- **Two stage approach:**
 - MIPS has **special registers** (**Hi** and **Lo**) where the **upper 32 bits** of the multiplication result are saved in Hi and the **lower 32 ones** in Lo (no output from the ALU.)
 - **mfhi** and **mflo** can then retrieve the values.

Lab 9: Performance

- We want to **determine** the **performance** of our processor on the example that you implemented in Lab 7: **summing up numbers**:
 - Once with a **loop that performs back to back additions**.
 - Once with the **Gaussian formula that performs multiplication**.
- You can use **your implementation** from Lab 7 and write a new code for calculating the Gaussian formula or **use the code that we provide**.
- Finally, you will **test your processor** with a **test bench**.

Last Words

- In this lab, you will **add instructions to the processor.**
 - **Adding instructions improves** the processor design.
 - **But this comes at a cost:** more instructions have to be decoded and more hardware resources are required.
- Processor designers need to face these **trade-offs.**
- We hope you have enjoyed the labs.
 - You have implemented **your own 32-bit processor.**
 - You have written programs for it.
 - You have improved its performance.

Report Deadline

23:59, 11 June 2021

Digital Design & Computer Arch.

Lab 9 Supplement: The Performance of MIPS

Prof. Onur Mutlu

ETH Zurich

Spring 2021

18 May 2021