

# Digital Design & Computer Arch.

## Lab 8.1 Supplement: Full System Integration

Prof. Onur Mutlu

ETH Zürich

Spring 2020

28 April 2020

# Lab 8 Overview

---

- You will build a **whole single-cycle processor** and **write assembly code** that runs on the **FPGA board**.
- **Don't worry!** You have **2 sessions** for the lab, and it will give you **up to 14 points (+6 points from the reports)**.
- You will learn how a processor is built.
- Learn how the processor **communicates with the outside world**.
- Implement the MIPS processor and demonstrate a simple "snake" program on the FPGA starter kit.

# Lab 8 Sessions

---

- **Session I:** The Crawling Snake
- **Session II:** Speed Up the Snake

# Lab 8 Session I: The MIPS Processor

---

- Download the Vivado project from the course website
- A lot of parts are **already implemented** for you!
- What you will have to implement:
  - Compute the **Instruction Memory** address and read the instruction.
  - Connect the **ALU**.
  - Compute the **Data Memory** address and add the necessary wires.
  - Instantiate the **Control Unit**.

# Lab 8 Session I: Memory-Mapped I/O

---

- Your goal is to control the 7-segment display with your assembly program.
- You will need to **complete the I/O controller** so the output of the processor will be correctly mapped to the display.

# I/O in Assembly

---

- How do we **communicate** with the display?
- **Memory Mapped I/O**
  - We designate **specific addresses** for the **I/O**
  - We can **read** and **write** to those addresses.
    - Example

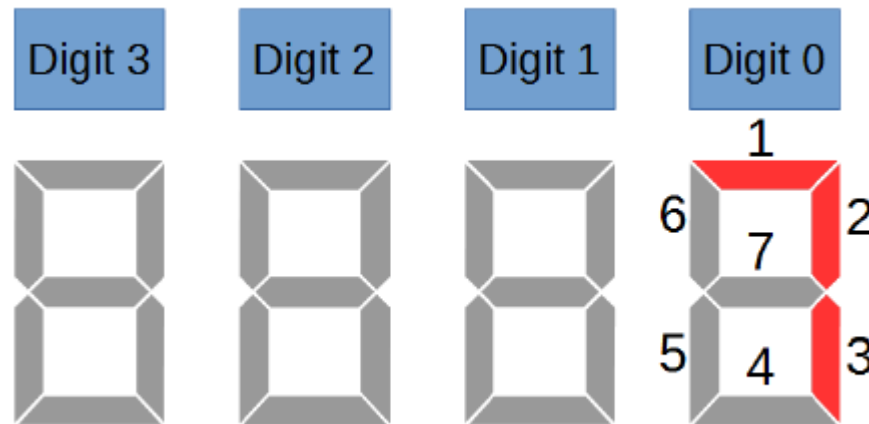
```
# write contents of $t0 into memory at address 0x7FF0  
# so that the I/O controller can send it to the display
```

```
sw $t0, 0x7FF0($0)
```

# Lab 8 Session I : The Crawling Snake

---

- You learned how to **write assembly code** in Lab 7
- We provide you with the code for a **crawling snake** on the **7-segment display**



- We will run this program on the processor you build in this lab!

# Lab 8 Session II: Speed Up The Snake

---

- You will **modify your assembly code** to **accept inputs**.
  - The snake should crawl at **different speeds** for **different inputs**.
  - The **inputs** will be controlled by **switches** on the **FPGA board**.
- **Modify** the **I/O controller** to **accept the inputs**.
- **Challenges**
  - Change the **direction** of the snake.
  - Change the **pattern** of the snake.



# Last Words

---

- You will build a **whole single-cycle processor** and **write assembly code** that runs on the **FPGA board**.
- You will learn how a processor is built.
- Learn how the processor **communicates with the outside world**.
- Implement the MIPS processor and demonstrate a simple “snake” program on the FPGA starter kit.
- You will have some questions to answer in the report.

# Digital Design & Computer Arch.

## Lab 8 Supplement: Full System Integration

Prof. Onur Mutlu

ETH Zürich

Spring 2020

28 April 2020