CS-200
Computer Architecture
—
Part 0. Introduction

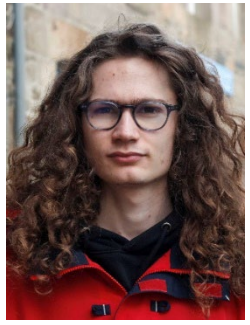
Paolo Ienne
<paolo.ienne@epfl.ch>

Who's Who

Lecturer: Paolo Ienne

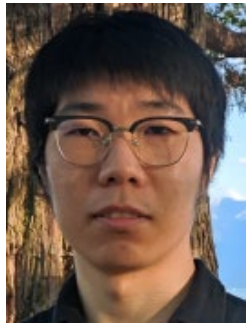
Teaching Assistants

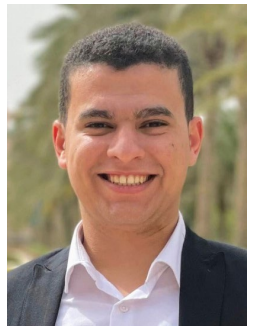

Ayan Chakraborty


Louis Coulon


Aya Elakhras

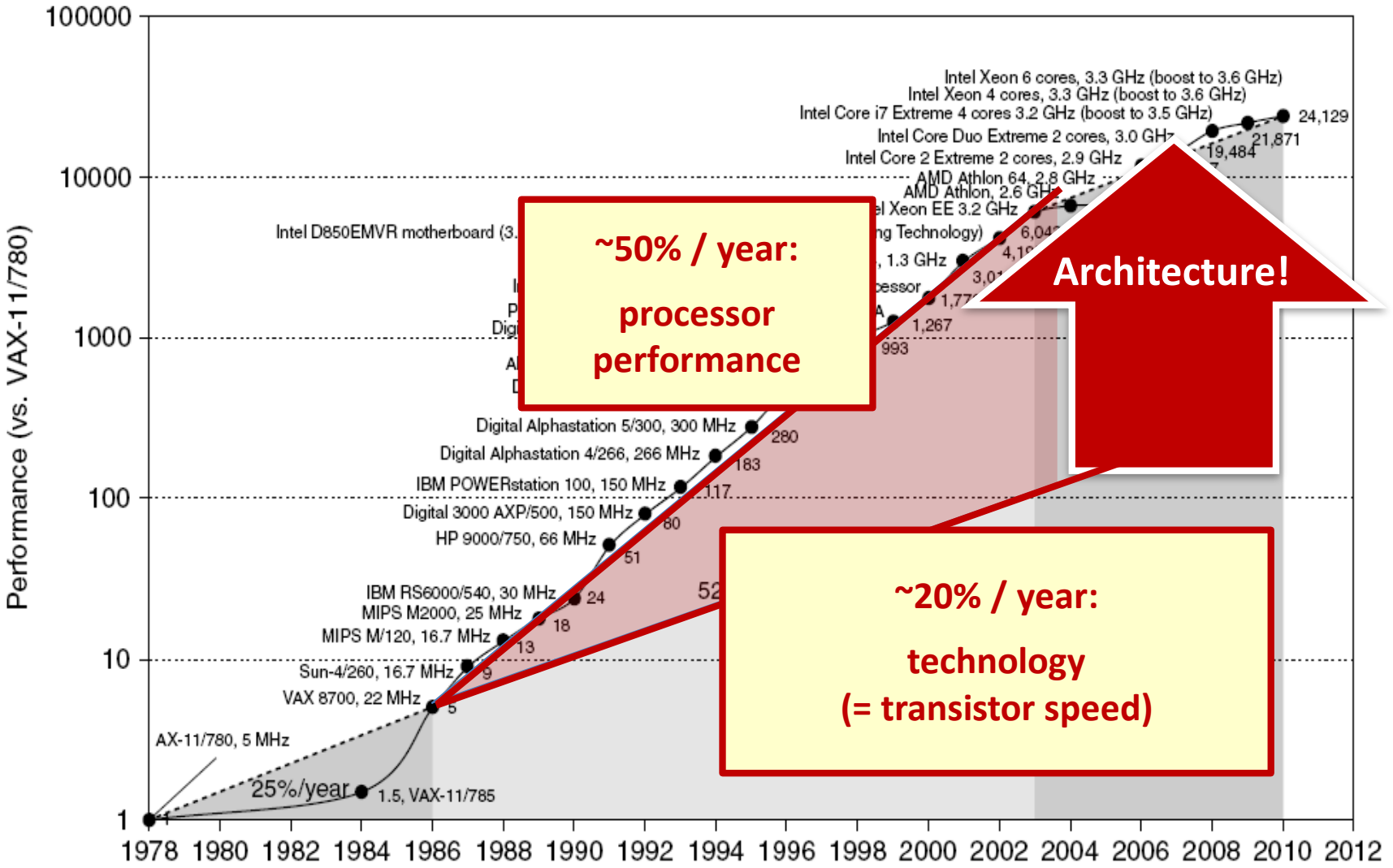

Buğra Eryılmaz


Yuanlong Li


Mohamed Shahawy

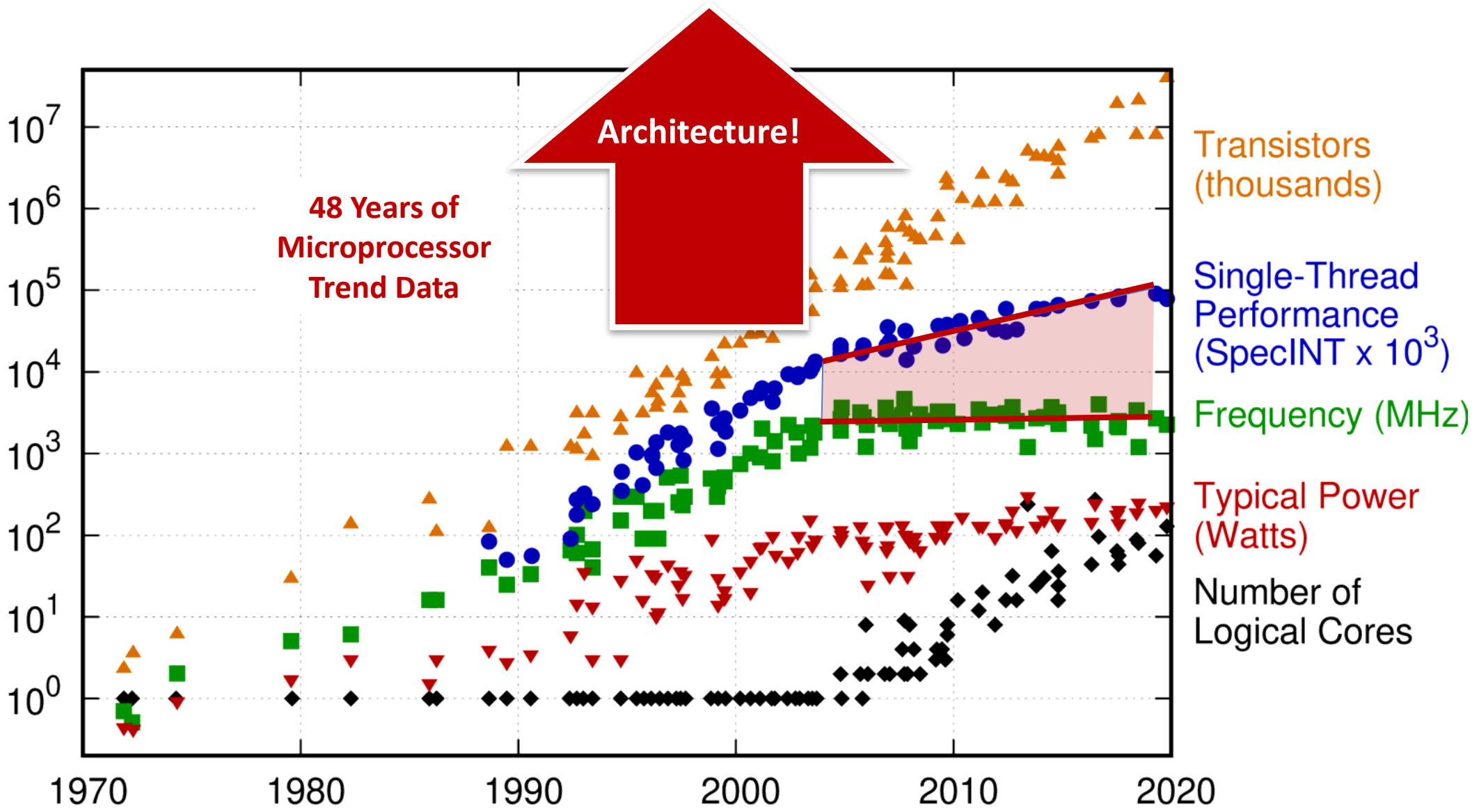
Student Assistants

Computer Architecture Rocks…



Source: Hennessy & Patterson, © MK 2011

…Especially Today!



Original graph by Horowitz et al.; this version © K. Rupp 2020

Content of CS-200 (1/2)

- **Part I: Processors and ISA**
  - What is a processor? How can we design one? How do programs look like when they are executed?
- **Part II: I/Os and Exceptions**
  - What is around a processor to make a full computer? How the processor exchanges information with the rest of the world?
- **Part III: Memory Hierarchy**
  - Processors are fast and memory is slow—how can one combine the two? How can one protect the data of users in memory?

Content of CS-200 (2/2)

- **Part IV: Instruction-Level Parallelism**
  - What makes a good processor? How real processors achieve ever increasing performances?

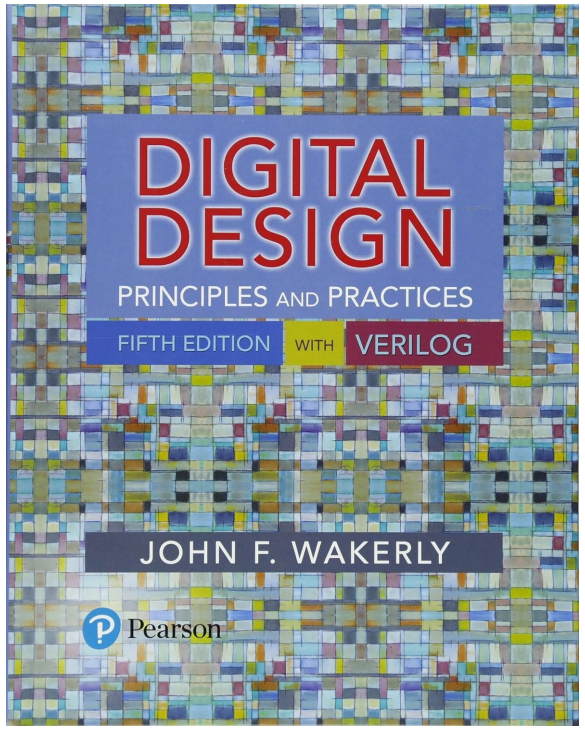
- **Part V: Multiprocessors**
  - What are the basic challenges of connecting many processors together? What changes from a single processor system?

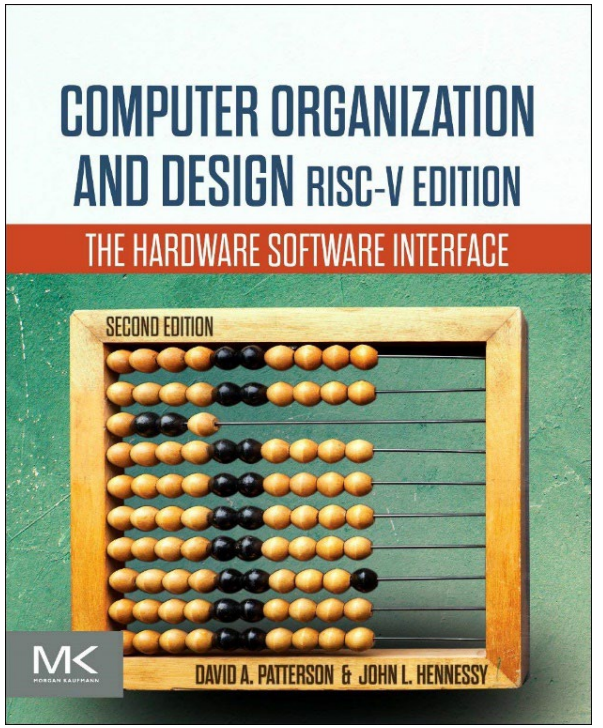
Two novelties

Verilog (instead of VHDL)
RISC-V (instead of Nios II)

Literature

- Everything recommended for **CS-173** applies here too
- In particular, **Wakerly** and **COD (RISC-V)**:





Moodle

- Moodle of this course:

https://moodle.epfl.ch/course/view.php?id=18287

- Everything is or will be there or linked there:
  - Schedule, slides, videos, forum, exercise book, labs, instructions, etc.

- Enrolment key:

CS200-2024

Slides & Video Recordings

- Slides will be available on **Moodle,** ideally **before** the course—but since the course is new, they may be available only right after (**sorry!**)

- All lectures are **streamed** and recorded:
  - SG1: **https://mediaspace.epfl.ch/media/0_5vkbg0rj**
  - STCC C: **https://mediaspace.epfl.ch/media/0_2o1cxd5m**
  - BCH2201: **https://mediaspace.epfl.ch/media/0_x1b0sxje**

- **Videos** will be available on

  https://mediaspace.epfl.ch/channel/CS-200/55022

CS-200 – Computer Architecture
Fall 2024

Week	Date	Time	Room	Lecture	Lab	Deadline
	Mon 09.09.24	1pm - 3pm	INF1 - INF3		Introduction to the infrastructure	Monday 09.09: Lab A (published)
1	Wed 11.09.24	10am - noon	SG1	Introduction + Processors & ISA (I) -- ISA Reminder, Assembly Language, Compilers		
	Wed 11.09.24	1pm - 3pm	INF1 - INF3		No labs	
	Fri 13.09.24	10am - noon	STCC C	Processors & ISA (II) -- Branches, Routines, Stack, and Heap		
	Mon 16.09.24					
2	Wed 18.09.24	10am - noon	SG1	Processors & ISA (III) -- Memory and Addressing Modes		
	Wed 18.09.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 20.09.24	10am - noon	BCH 2201	Processors & ISA (IV) -- Examples of Assembly Programming		
	Mon 23.09.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
3	Wed 25.09.24	10am - noon	SG1	Processors & ISA (V) -- Computer Arithmetic		
	Wed 25.09.24	1pm - 3pm	INF1 - INF3		Exercises on Processors & ISA	
	Fri 27.09.24	10am - noon	STCC C	I/Os & Exceptions (I) -- Multicycle Processor and Memory Organization		
	Mon 30.09.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
4	Wed 02.10.24	10am - noon	SG1	I/Os & Exceptions (II) -- Inputs and Outputs		
	Wed 02.10.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 04.10.24	10am - noon	BCH 2201	I/Os & Exceptions (III) -- Interrupts		
	Mon 07.10.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
5	Wed 09.10.24	10am - noon	SG1	I/Os & Exceptions (IV) -- Exceptions, Privilege Leve		
	Wed 09.10.24	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 11.10.24	10am - noon	BCH 2201	I/Os & Exceptions (V) -- Exception Handlers and Pr		Sunday 13.10 @ 23:59: Lab A (final)
	Mon 14.10.24	1pm - 3pm	INF1 - INF3		Exercises on I/Os and Exceptions	Monday 14.10: Lab B (published)
6	Wed 16.10.24	10am - noon	SG1	Memory Hierarchy (I) -- Caches		
	Wed 16.10.24	1pm - 3pm	INF1 - INF3		Lab B.1 (RISC-V Multicycle Processor): ALU	
	Fri 18.10.24	10am - noon	STCC C	Memory Hierarchy (II) -- Examples of Caches		
	Mon 28.10.24	1pm - 3pm	INF1 - INF3		Lab B.2 (RISC-V Multicycle Processor): Core	
7	Wed 30.10.24	10am - noon	SG1	Memory Hierarchy (III) -- Virtual Memory		
	Wed 30.10.24	1pm - 3pm	INF1 - INF3		Lab B.2 (RISC-V Multicycle Processor): Core	
	Fri 01.11.24	10am - noon	STCC C			Sunday 03.11 @ 23:59: Lab B.1 (partial)
	Mon 04.11.24	1pm - 3pm	INF1 - IN		Lab B.2 (RISC-V Multicycle Processor): Core	
8	Wed 06.11.24	10am - noon	SG1			
	Wed 06.11.24	1pm - 3pm	INF1 - IN		Exercises on Memory Hierarchy	
	Fri 08.11.24	10am - noon	STCC C			Sunday 10.11 @ 23:59: Lab B.2 (partial)
	Mon 11.11.24	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
9	Wed 13.11.24	10am - 1:15pm	TBA	Midterm		
	Wed 13.11.24					
	Fri 15.11.24	10am - noon	STCC C	Instruction-Level Parallelism (II) -- Pipelining		
	Mon 18.11.24	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
10	Wed 20.11.24	10am - noon	SG1	Instruction-Level Parallelism (III) -- Pipelining (cont'd)		
	Wed 20.11.24	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
	Fri 22.11.24	10am - noon	STCC C	Instructi		
	Mon 25.11.24	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	Monday 25.11: Lab C (published)
11	Wed 27.11.24	10am - noon	SG1	Instructi		
	Wed 27.11.24	1pm - 3pm	INF1 - INF3		Lab C.1 (RISC-V Interrupts): Adding Interrupt Support	
	Fri 29.11.24	10am - noon	STCC C	Instruction-Level Parallelism (VI) -- Examples of Scheduling		Sunday 01.12 @ 23:59: Lab B (final)
	Mon 02.12.24	1pm - 3pm	INF1 - INF3		Lab C.1 (RISC-V Interrupts): Adding Interrupt Support	
12	Wed 04.12.24	10am - noon	SG1	Instruction-Level Parallelism (VII) -- Intel x86 and ARM		
	Wed 04.12.24	1pm - 3pm	INF1 - INF3		Exercises on Instruction-Level Parallelism	
	Fri 06.12.24	10am - noon	STCC C	Multiprocessors (I) -- Cache Coherence		Sunday 08.12 @ 23:59: Lab C.1 (partial)
	Mon 09.12.24	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
13	Wed 11.12.24	10am - noon	SG1	Multiprocessors (II) -- Cache Coherence with Examples		
	Wed 11.12.24	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
	Fri 13.12.24	10am - noon	STCC C	Multiprocessors (III) -- Memory Consistency		
	Mon 16.12.24	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
14	Wed 18.12.24	10am - noon	SG1	Multiprocessors (IV) -- Synchronization		
	Wed 18.12.24	1pm - 3pm	INF1 - INF3		Exercises on Multiprocessors	
	Fri 20.12.24	10am - noon	STCC C	Exercises		Sunday 22.12 @ 23:59: Lab C (final)

Course not at STCC C!

Five exercise sessions

Midterm!

Important lab deadlines

WELCOME TO THE SWISSTECH CONVENTION CENTER

THIS COURSE IS BEING HELD IN A PRIVATE BUILDING. ONLY USE THE ENTRANCE AND FACILITIES CLEARLY INDICATED «COURS EPFL»

PLEASE DO NOT EAT OR DRINK IN THE AUDITORIUM

DON’T FORGET TO TAKE ALL YOUR BELONGINGS AND ANY WASTE WITH YOU

SHOULD YOU FORGET SOMETHING, NOTE THAT THE «LOST & FOUND» IS LOCATED AT THE WELCOME DESK ON THE EPFL CAMPUS

THANK YOU FOR YOUR COOPERATION !

SwissTech
CONVENTION CENTER

Grading

- Midterm test (**3h**):
  - **Wednesday 13th November, 10:15am-13:15pm**, rooms **TBA** (the afternoon lab is cancelled)
  - On **Processors and ISA, I/Os & Exceptions,** and **Memory Hierarchy**
  - Books, notes, exercises, etc. are **not permitted**
  - Essential quick references will be **distributed** (e.g., ISA)
  - The midterm test counts for **35% of the final grade**
- Final exam in the exam session (**3h**):
  - Will be fixed by SAC in due time
  - On **Instruction-Level Parallelism** and **Multiprocessors**
  - Same rules as above for the Midterm
  - The final exam counts for **35% of the final grade**
- Labs:
  - **Three labs (A, B, C)** composed of various sessions each; each lab counts for **10% of the final grade**

Labs

- **Graded** automatically with **anti-plagiarism check**
- Must be done **individually**
- **Deadlines** on the schedule
- **Verilog** code must follow our **guidelines** or (i) **no support** during the lab sessions and (ii) **50% grade lost!**
- They need a variety of tools, so **use the pre-configured VM**
  - **First lab was a tutorial** (check on Moodle if you missed it)
  - Everything is open source but you are on your own if you want to install on your computer (rather easy on Linux, harder elsewhere)

Lab Submissions

- Submission deadlines marked **“final”** are **graded**
  - **Three final submissions** for the three labs A, B, and C
  - You can submit **twice** and the **last one** counts for the grade
  - Limited diagnostic information—essentially, what passed and what not

- Before, there are submission deadlines marked **“partial”** (e.g., not the complete lab B but only part B.1)
  - You can submit **twice**
  - They are **not graded** but should give you a sense of what is expected
  - You will get a bit more information on what did not work
  - Do not expect full debugging info—testing and debugging is **your** job!

```
======== Testing summary ========
Test clear_leds: passed (clear_leds on a random screen)
Test clear_leds: 1 out of 1 passed -> Score: 2.0 / 2
Test set_pixel: passed (set_pixel setting some pixel on empty screen)
Test set_pixel: passed (set_pixel setting some pixel on non empty screen when writting on LEDS[0])
.
.
.
Test pause_game: passed (changes the paused state again)
Test pause_game: 2 out of 2 passed -> Score: 5.0 / 5
Test change_steps: passed (Changes the number of steps in the INIT state with button 2, 3 and 4 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 2 and 3 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 2 and 4 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 2 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 3 and 4 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 3 set)
Test change_steps: passed (Changes the number of steps in the INIT state with button 4 set)
Test change_steps: passed (Changes the number of steps in the INIT state with no button set)
Test change_steps: 8 out of 8 passed -> Score: 5.0 / 5
Test increment_seed: passed (Increments the seed from a random number)
Test increment_seed: passed (Increments the seed from a random number)
Test increment_seed: failed (Increments the seed from N_SEEDS)
Test increment_seed: 2 out of 3 passed -> Score: 3.3333333333333335 / 5
Test update_state: passed (INIT, button 2)
Test update_state: passed (INIT, button 3)
.
.
.
Test find_neighbours: passed (2 neighbours)
Test find_neighbours: 9 out of 9 passed -> Score: 10.0 / 10
Test update_gsa: passed (update gsa from a random state, RUNNING)
Test update_gsa: passed (update gsa from a random state, PAUSED)
Test update_gsa: 2 out of 2 passed -> Score: 10.0 / 10
Test mask: passed (Mask 3 on a random gsa)
Test mask: passed (Mask 0 on a random gsa)
Test mask: 2 out of 2 passed -> Score: 5.0 / 5
======== Total score: 98.33 / 100 ========
Note that 100% score corresponds to 80% of the maximum possible score on the
project. For the remaining 20% you will need to make a successful live
demonstration of the game to the Teaching Assistants.
```

Lab A Submission

Several checks in each unit test

Details on the failing checks

20% of the grade for a live demonstration

Labs B & C
“Partial” Submission

```
Running lint...
Lint failed for one or more files

============================================================

Compiling testbench: testbench/tb_controller.v
Running testbench: testbench/tb_controller.v

First error encountered:
Error in test case 39 at time 1530000:
  instruction_i: 0x00000013
  Expected:
    imm = 0x00000000, branch_op = 1, ir_en = 0
    pc_add_imm = 0, pc_en = 1, pc_sel_alu = 0, pc_sel_pc_base = 0
    pc_sel_mtvec = 1, pc_sel_mepc = 0
    rf_we = 0, sel_addr = 0, sel_b = 0, sel_mem = 0
    sel_pc = 0, sel_imm = 0, sel_csr = 0, we = 0, alu_op = 000000
    csr_write = 0, csr_set = 0, csr_clear = 0
    csr_interrupt = 1, csr_mret = 0
  Got:
    imm = 0x00000000, branch_op = 0, ir_en = 0
    pc_add_imm = 0, pc_en = 1, pc_sel_alu = 0, pc_sel_pc_base = 0
    pc_sel_mtvec = 1, pc_sel_mepc = 0
    rf_we = 0, sel_addr = 0, sel_b = 0, sel_mem = 0
    sel_pc = 0, sel_imm = 0, sel_csr = 0, we = 0, alu_op = 000000
    csr_write = 0, csr_set = 0, csr_clear = 0
    csr_interrupt = 1, csr_mret = 0

One or more tests failed

============================================================

Compiling testbench: testbench/tb_cpu.v
Running testbench: testbench/tb_cpu.v

All tests passed

.
.
.
```

Linter fails: no details because you can run it yourself

Details on the first error of each testbench

More errors?

Same for all testbenches included in the “partial” submission

Labs B & C
“Final” Submission

```
Running lint...
Lint failed for one or more files

============================================================

Compiling testbench: testbench/tb_controller.v
Running testbench: testbench/tb_controller.v

One or more tests failed

============================================================

Compiling testbench: testbench/tb_cpu.v
Running testbench: testbench/tb_cpu.v

All tests passed

.
.
.

Compiling testbench: testbench/tb_buttons.v
Running testbench: testbench/tb_buttons.v

All tests passed

============================================================

Compiling testbench: testbench/tb_mem.sv
Running testbench: testbench/tb_mem.sv

All tests passed

============================================================

Final grade = 47.5/100

============================================================
```

Linter fails: no details because you can run it yourself

Only fail/pass info per testbench

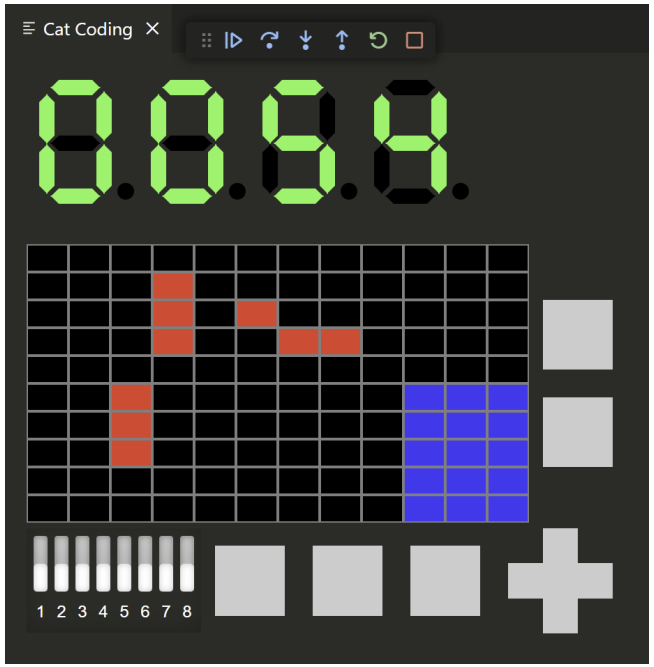
Final grade halved because linter failed

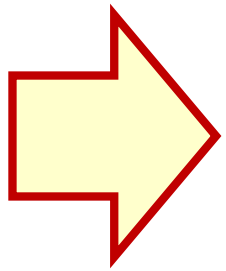
Labs
(if you are taking CS-200 again)

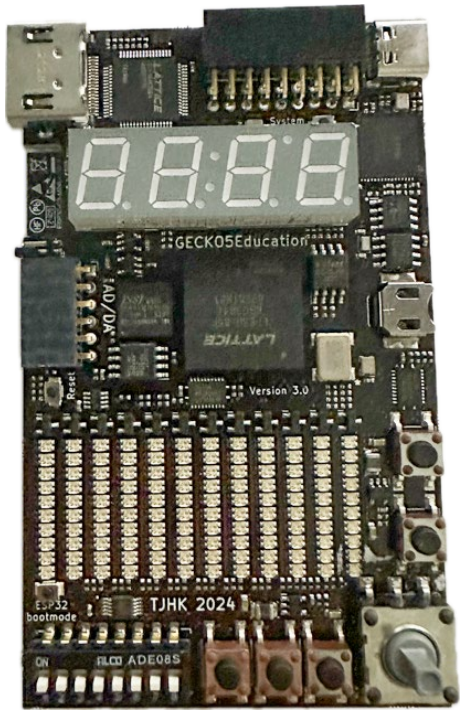
- Must do the labs **again!**

Gecko5 Board

- Labs **only in simulation** but on a custom simulator of a **real board**
- Optionally, you can run **your designs** on a **real board!**
- **No impact on the grade**, only for the fun of it…







Q&A Forum

- Q&A Forum on **Ed Discussion**
- Provides **answers to FAQs** and a way to get help efficiently, from both TAs and colleagues
- Please **read the rules** before posting (no posting of solutions, “netiquette”, etc.)

Exercise Book

- Available soon on **Moodle**
- A collection of **old exam questions (translated to RISC-V and Verilog)** with full solutions
- **Best way to assess if you are prepared for the tests!**
- Some of the in-class exercises and lab-session exercises are also (modified) exam questions

On Your Mark!… Get Set!… Go!

- Any info missing? Ask now…