TÉCNICO LISBOA

Computer Electronics

Lecture 1: Course introduction

TÉCNICO LISBOA

About the course

- Basic organization of a modern computer system
- Main digital electronic components
- Practical approach: design, verify (simulate), and implement in silicon
- Commercial design tools and IP are very expensive (millions)
- Today there are open-source design tools and IP
  - Hardware design is being democratized!

About me

- José Teixeira de Sousa
  - IST/DEEC professor
  - IObundle manager
- https://www.**linkedin**.com/in/jose-t-de-sousa-a34b86/

The classes

- Theory lectures
  - 2 x 1h30 per week
- Laboratory sessions
  - 1 x 1h30 per week

Theory lectures

- Presentation of materials: 1 hour or less
- Online Google Form Quiz: ½ hour

TÉCNICO LISBOA

Lab sessions

- Groups of 1, 2 or 3 students
- 1 introductory lab assignment
- 1 project during the whole semester
  - A simple computing system will be developed
- Each group must have a git repository
  - Software: C code
  - Hardware: Verilog code of an accelerator for the software
  - Documentation: user guide done in Latex

Lab tools

- Linux distro and associated tools: Make, code editor (Emacs recommended), bash, sed, etc. 40+GB recommended size
- Icarus Verilog for digital circuit simulation
- Gtkwave for visualizing waveforms
- Tex Live (full installation)
- *FPGA synthesis and implementation*
  - *Xilinx ISE/Vivado?*
  - *Intel Quartus?*
- *OpenRoad ASIC synthesis and implementation?*

Evaluation

- Exam or 2 tests (T)
- Lab (L)
- Google form quizzes (E, optional)
- Minimum mark on T or L: 8/20
- Mark:

$$M = \max(0.4T+0.2E+0.4L,\ 0.6T+0.4L)$$

Tests and exams

- 2 Tests or Exam
- Duration:
  - Tests: 1h30
  - Exam: 3h00
- Check Fénix for dates
- Exam is divided in 2 parts which may be used to recover one of the tests

Lab evaluation criteria

- Instructor will
  - Git clone your group’s git repository
  - Run simulation (make sim)
  - Run FPGA implemenation (make fpga)
  - Generate document (make doc)
  - Make sure the above works flawlessly
  - Quality of the solution (performance, frequency, size trade-offs and presentation)

TÉCNICO LISBOA

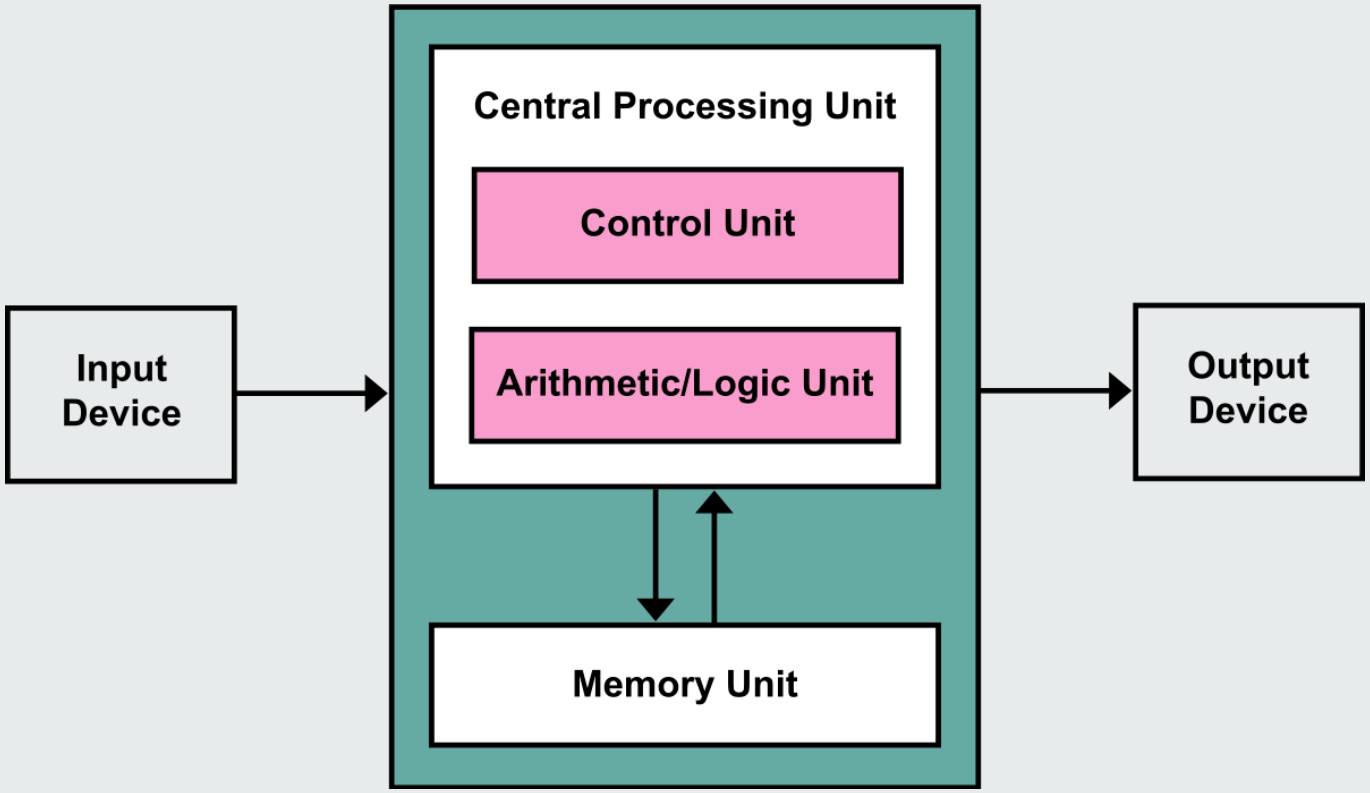
Course program

- Processor Systems on Chip
- The Verilog Hardware Description Language (HDL)
- The Arithmetic and Logic Unit
- Memories (SRAM, DRAM, CACHE)
- Peripherals, especially for acceleration

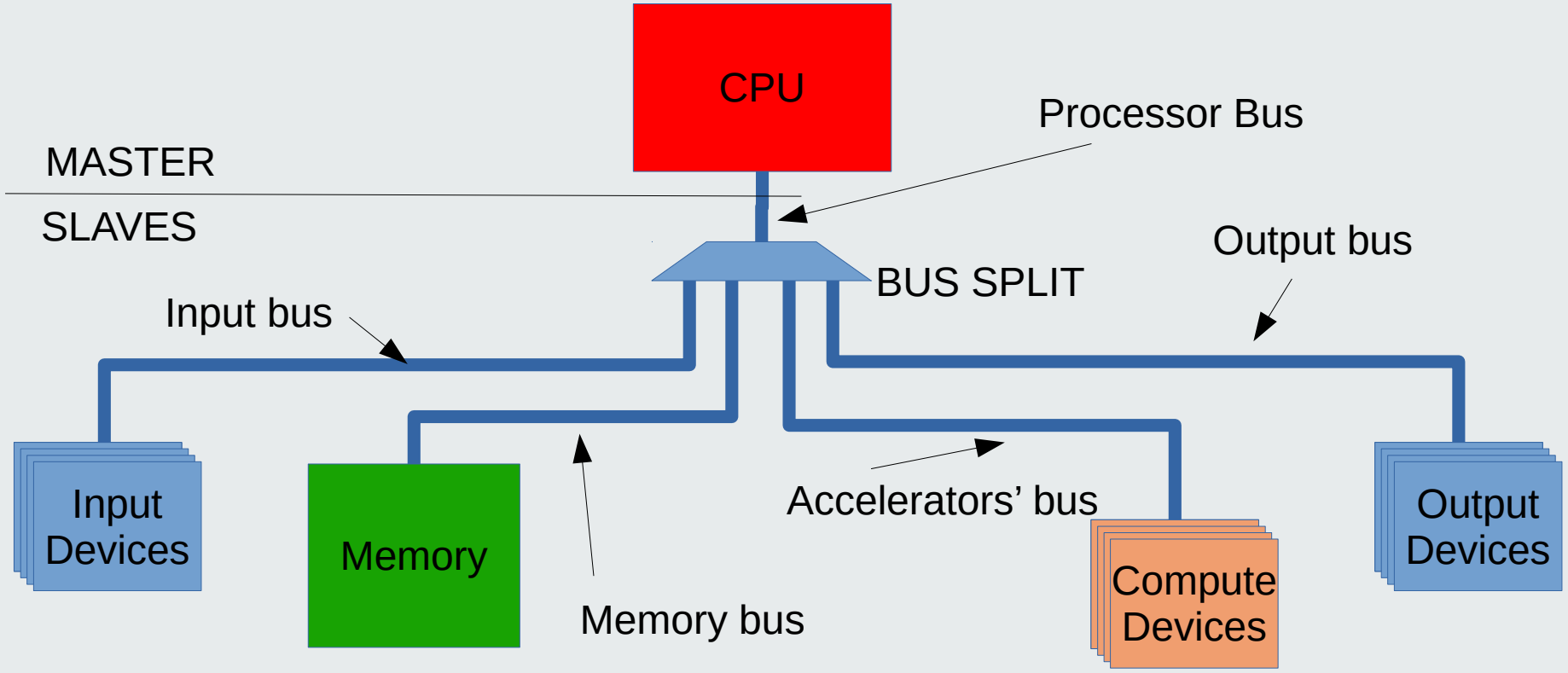
Bibliography

- The Internet!
- Lecture slides
- Exercises and their resolution
- Lab materials

The von Neumann Architecture



The von Neumann Architecture in practice

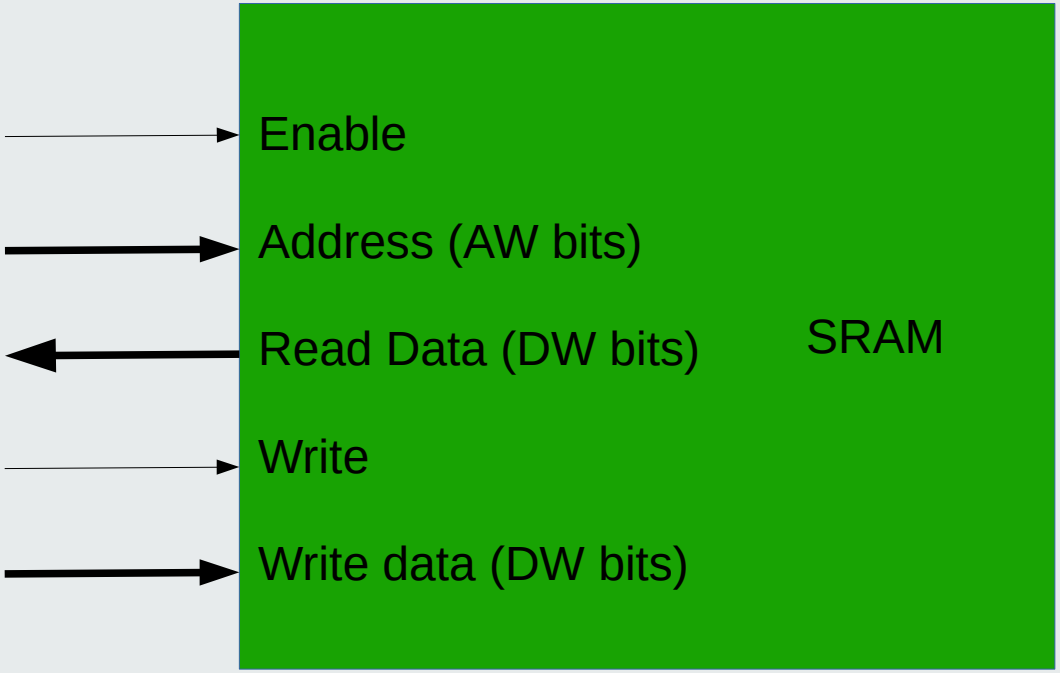


The Processor Bus

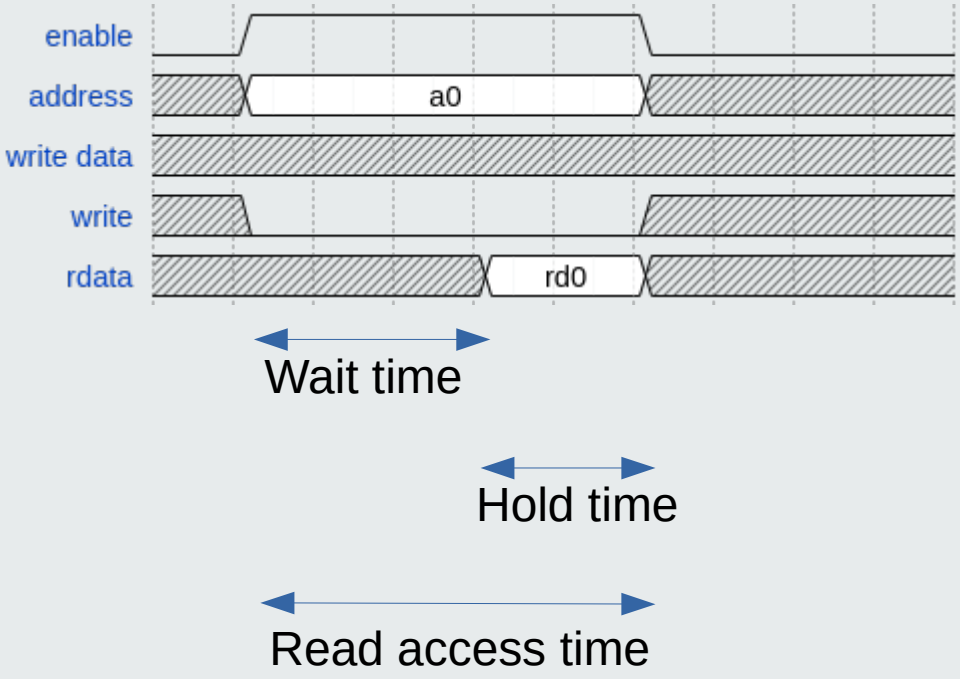
- Processor buses can be complex, for example ARM's AXI
- In this course a minimalistic yet effective 6-signal bus called the Native Bus will be used:

	Signal Name	Direction wrt CPU	Width in bits (wires)	Description
REQUEST	Valid	Output	1	CPU wants to access data or instructions
	Address	Output	32	Address of the data
	Wdata	Output	32	Data to write
	Wstrb	Output	4	Byte write
RESPONSE	Rdata	Input	32	Data read
	Ready	Input	1	Slave is ready

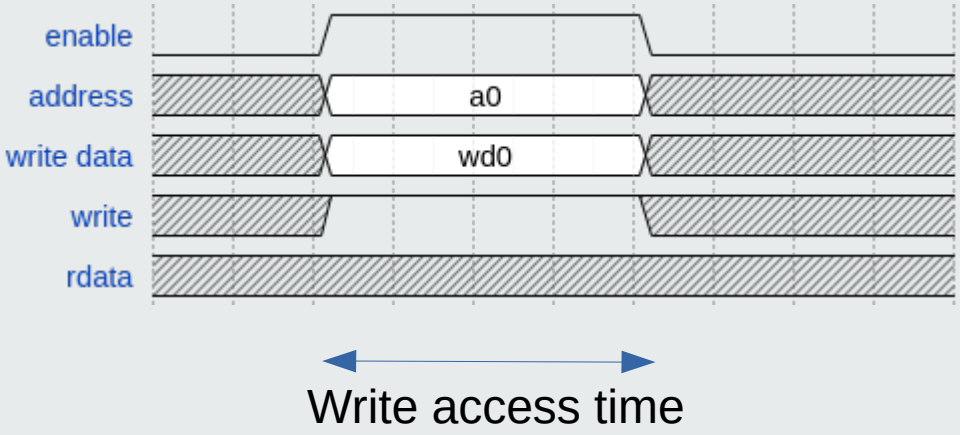
Static Random Access Memory (SRAM): the simplest memory



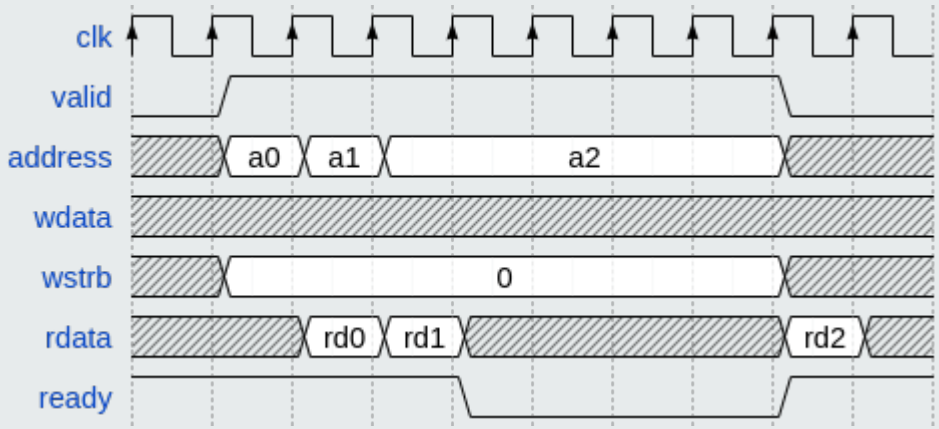
SRAM: reading the memory



SRAM: writing the memory



Master reading from slave



Master writing to slave