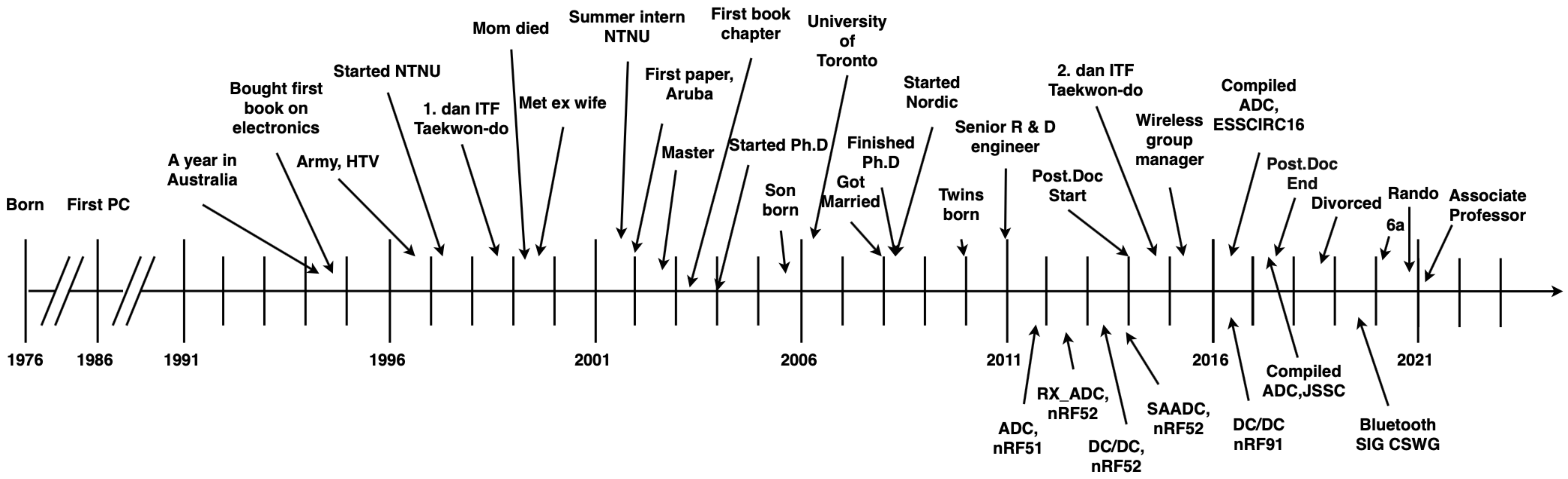
TFE4188 - Lecture 1

Advanced Integrated Circuits
Introduction

https://wulffern.github.io/aic2023

Who

Carsten Wulff carstenw@ntnu.no



Teaching assistants

- Fredrik Esp Feyling
- Jonathan Sæthre Ege

Why

I want you to learn the necessary skills to make your own ICs

Insights · Tech The Future

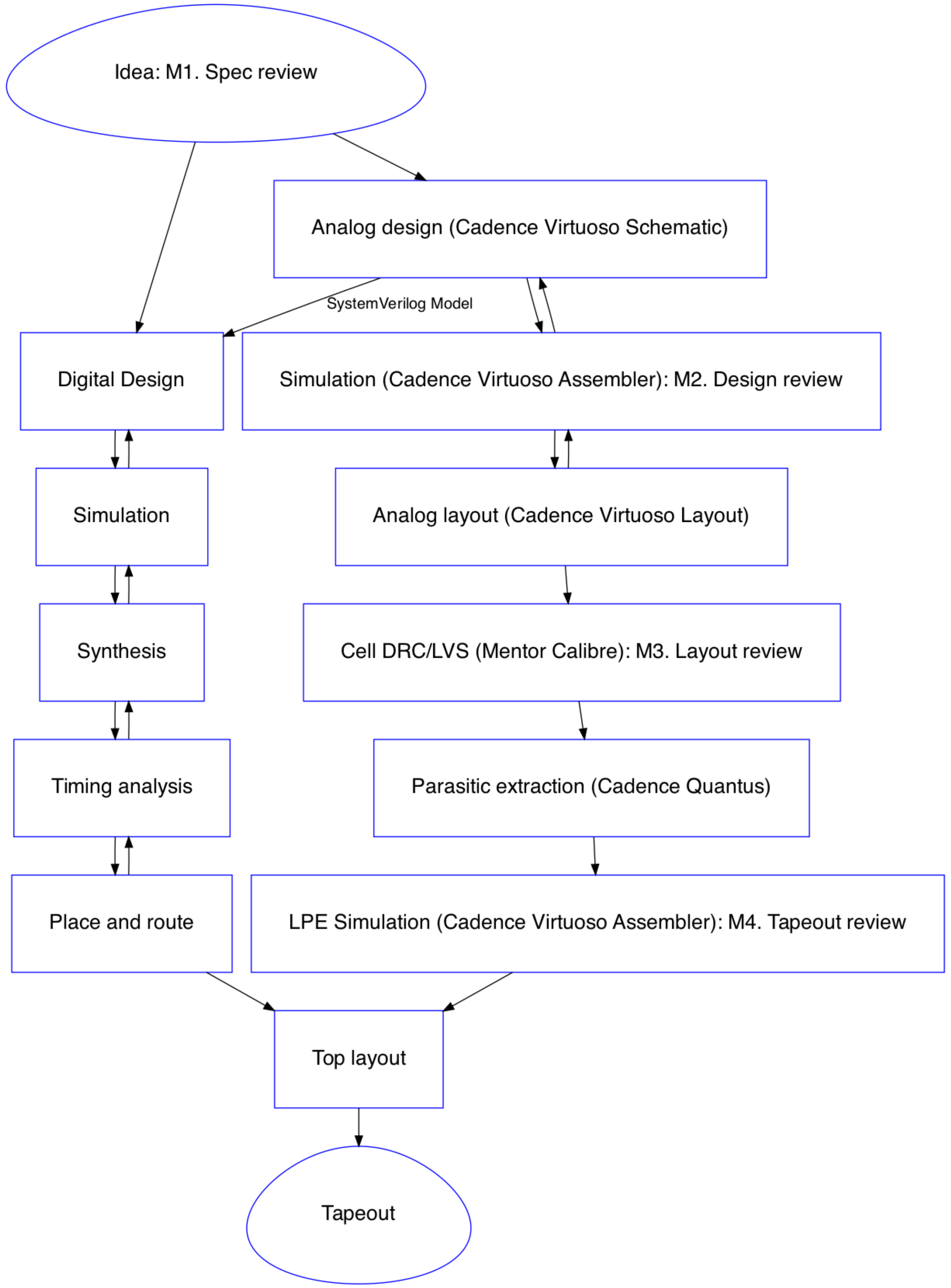
The World Is Analog

10/28/2014

Written by Peter Kinget

The world we live in is analog. We are analog. Any inputs we can perceive are analog. For example, sounds are analog signals; they are continuous time and continuous value. Our ears listen to analog signals and we speak with analog signals. Images, pictures, and video are all analog at the source and our eyes are analog sensors. Measuring our heartbeat, tracking our activity, all requires processing analog sensor information.

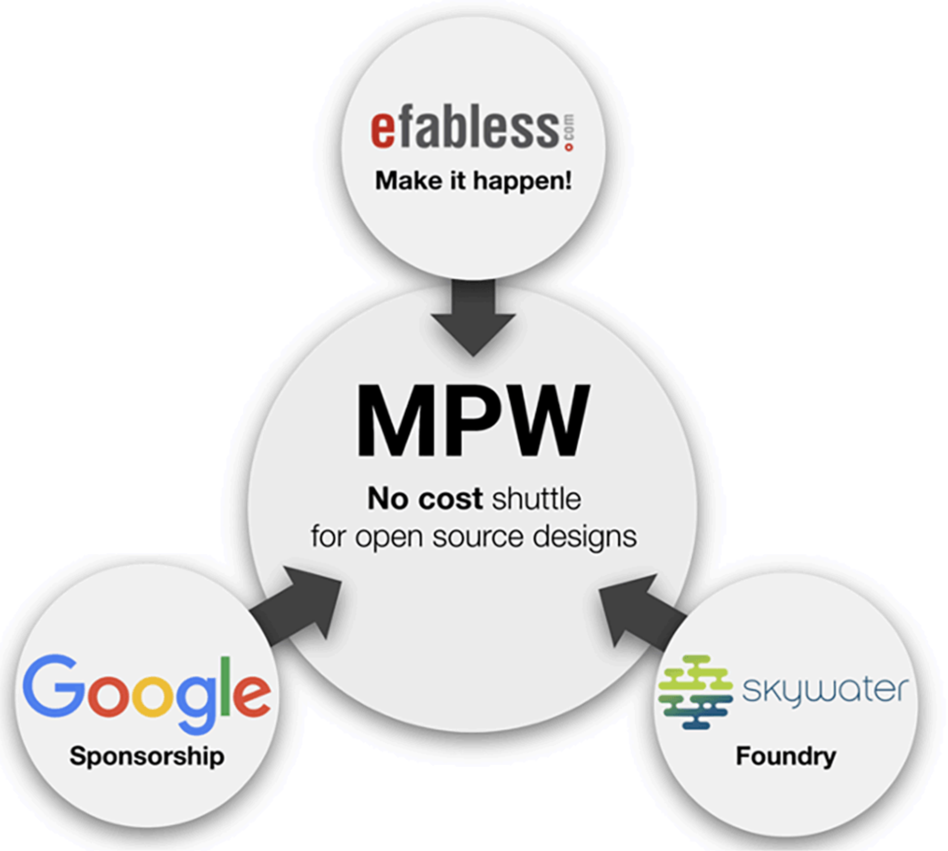
https://circuitcellar.com/insights/tech-the-future/kinget-the-world-is-analog/



Idea: M1. Spec review
Analog design (Cadence Virtuoso Schematic)
SystemVerilog Model
Digital Design
Simulation (Cadence Virtuoso Assembler): M2. Design review
Simulation
Analog layout (Cadence Virtuoso Layout)
Synthesis
Cell DRC/LVS (Mentor Calibre): M3. Layout review
Timing analysis
Parasitic extraction (Cadence Quantus)
Place and route
LPE Simulation (Cadence Virtuoso Assembler): M4. Tapeout review
Top layout
Tapeout

I would like some of you to tapeout an IC

Welcome to the Efabless Open MPW Program



The shuttle provides opportunities for designers to experiment and push the state-of-the-art without having to reconcile the risk associated with the cost of fabrication.

The shuttle program is open to anyone, provided that their project is fully open source and meets the other program requirements.

Costs for fabrication, packaging, evaluation boards and shipping are covered by Google for this program.

- *Project flow support*: **Confluence**, JIRA, risk management (DFMEA), failure analysis (8D)
- *Language*: **English**, **Writing English (Latex, Word, Email)**
- *Psychology*: Personalities, convincing people, presentations (Powerpoint, Deckset), **stress management (what makes your brain turn off?)**
- *DevOps*: **Linux**, bulid systems (CMake, make, ninja), continuous integration (bamboo, jenkins), **version control (git)**, containers (docker), container orchestration (swarm, kubernetes)
- *Programming*: Python, Go, C, C++, Matlab Since 1999 I've programmed in Python, Go, Visual BASIC, PHP, Ruby, Perl, C#, SKILL, Ocean, Verilog-A, C++, BASH, AWK, VHDL, SPICE, MATLAB, ASP, Java, C, SystemC, Verilog, and probably a few I've forgotten.
- *Firmware*: signal processing, algorithms
- *Infrastructure*: **Power management**, **reset**, **bias**, **clocks**
- *Domains*: CPUs, peripherals, memories, bus systems
- *Sub-systems*: **Radio's**, **analog-to-digital converters**, **comparators**
- *Blocks*: **Analog Radio**, Digital radio baseband
- *Modules*: Transmitter, **receiver**, de-modulator, timing recovery, state machines
- *Designs*: **Opamps**, **amplifiers**, **current-mirrors**, adders, random access memory blocks, standard cells
- *Tools*: **schematic**, **layout**, **parasitic extraction**, synthesis, place-and-route, **simulation**, (System)Verilog, **netlist**
- *Physics*: transistor, pn junctions, quantum mechanics

Find a problem that you really want to solve, and learn programming to solve it. There is no point in saying "I want to learn programming", then sit down with a book to read about programming, and expect that you will learn programming that way. It will not happen. The only way to learn programming is to do it, a lot.

— Carsten Wulff

```
s/programming/analog design/ig
```

Zen of IC design (stolen from Zen of Python)

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Readability counts (especially schematics).
- Special cases aren't special enough to break the rules.
- Although practicality beats purity.
- In the face of ambiguity, refuse the temptation to guess.
- There should be one **and preferably only one** obvious way to do it.
- Now is better than never.
- Although never is often better than *right* now.
- If the implementation is hard to explain, it's a bad idea.
- If the implementation is easy to explain, it may be a good idea.

Course

Goal of lectures

- Enable you to read the book
- Enable you to read papers (latest research)
- Correct misunderstandings of the topic
- Answer any questions you have on the chapters

Plan

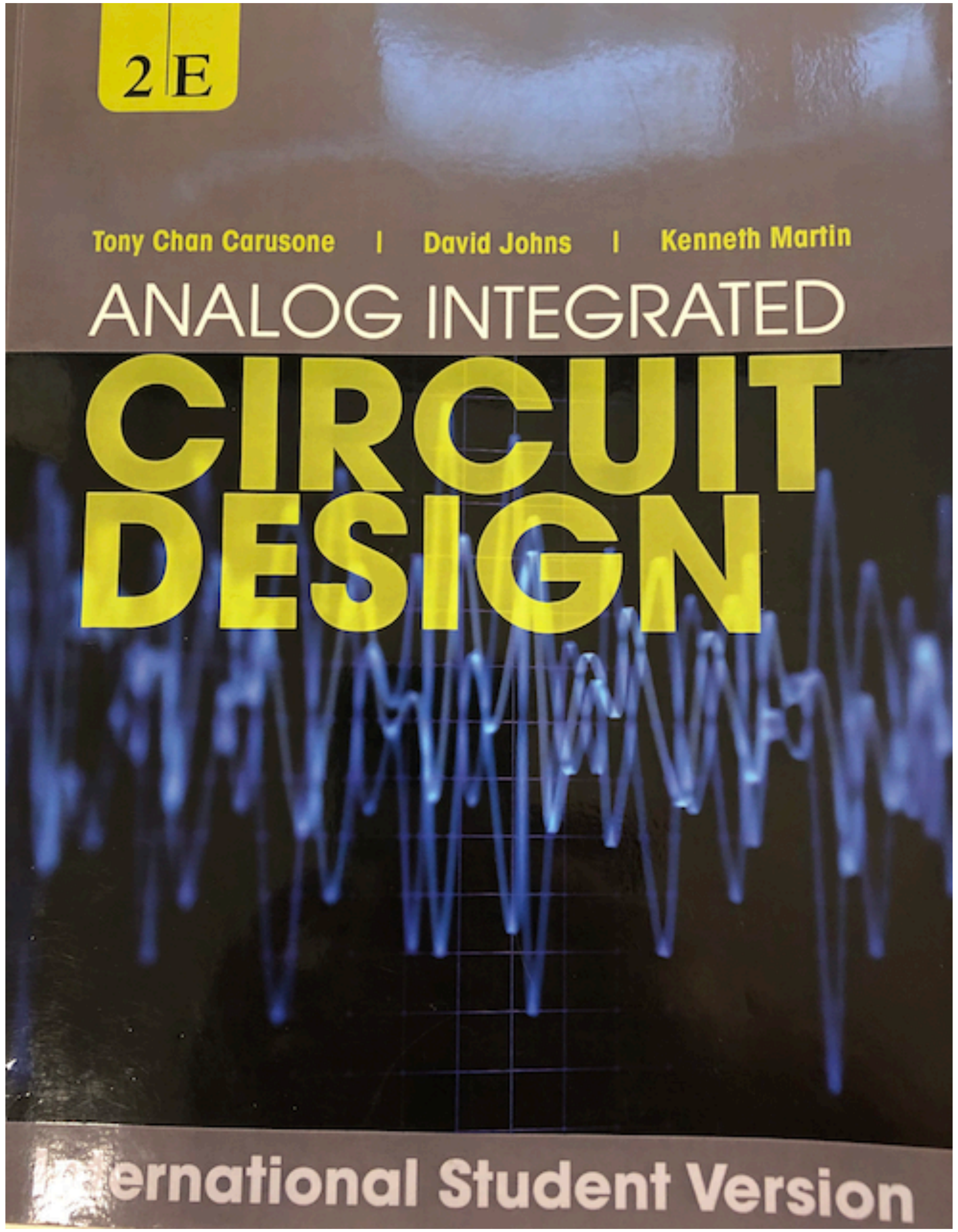
Lectures:
Thursday at 08:15 - 10:00

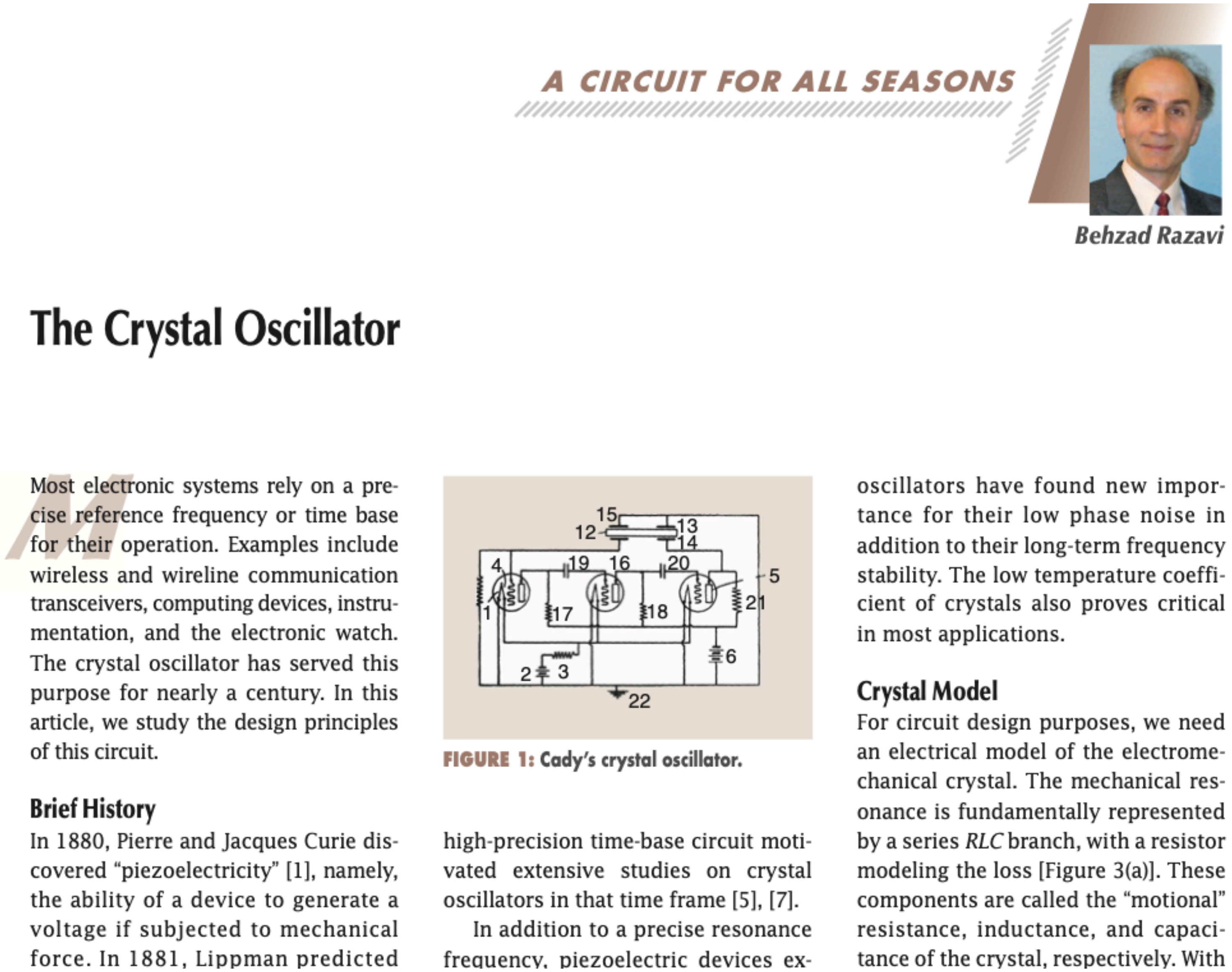
The "lectures" will be Q & A's on the topic. If no questions, then I'll ramble on

Exercise Hours:
Friday at 14:15 - 16:00

The TA will be in the "exercise hours", and I also will hopefully join most days.

- Description
- Time schedule
- Lecture plan
- Syllabus





A CIRCUIT FOR ALL SEASONS

Behzad Razavi

The Crystal Oscillator

Most electronic systems rely on a precise reference frequency or time base for their operation. Examples include wireless and wireline communication transceivers, computing devices, instrumentation, and the electronic watch. The crystal oscillator has served this purpose for nearly a century. In this article, we study the design principles of this circuit.

Brief History

In 1880, Pierre and Jacques Curie discovered "piezoelectricity" [1], namely, the ability of a device to generate a voltage if subjected to mechanical force. In 1881, Lippman predicted

FIGURE 1: Cady's crystal oscillator.

high-precision time-base circuit motivated extensive studies on crystal oscillators in that time frame [5], [7].

In addition to a precise resonance frequency, piezoelectric devices ex-

oscillators have found new importance for their low phase noise in addition to their long-term frequency stability. The low temperature coefficient of crystals also proves critical in most applications.

Crystal Model

For circuit design purposes, we need an electrical model of the electromechanical crystal. The mechanical resonance is fundamentally represented by a series *RLC* branch, with a resistor modeling the loss [Figure 3(a)]. These components are called the "motional" resistance, inductance, and capacitance of the crystal, respectively. With

Exam

- May/June 2022?
- 4 hours
- A - F grade (F = Fail)

Time to take responsibility for your own future

Exercises

- Exercises on blackboard now
- Solutions on blackboard after the deadline
- **One compulsory exercise: Exercise 0 - Skywater 130nm tutorial**
- For the rest, two options:
  - Don't do the exercises, don't get feedback
  - Do the exercises, hand them in within deadline, get feedback
- The TA's will only support the exercises in the marked weeks

Project

Compulsory submission, no exam without submission.

Minimum: Schematic, and simulation of a temperature sensor in sky130B and a project report

Maximum: Tapeout submission to Google/Efabless Open MPW Shuttle. **No report necessary.**

Do it in groups, or personal, you decide.

Deadline: 29 of April

Strict deadline, if you hand in 30 of April at 00:00:01, then no exam.

RIPLEY (2023)

"I say we take off and nuke the site from orbit. It's the only way to be sure.", Ellen Ripley, Aliens

Design a integrated temperature sensor with digital read-out.

Project Report $\Rightarrow$ Paper

A Compiled 9-bit 20-MS/s 3.5-fJ/conv.step SAR ADC in 28-nm FDSOI for Bluetooth Low Energy Receivers

IEEE journal template, Example

Must use `\documentclass[journal,11pt,letterpaper]{IEEEtran}`

Strict page limit for report, max 8 pages (excluding bio and references). More than 8 pages $\Rightarrow$ Not approved submission, and no exam.

Software

Open source software (xschem, ngspice, sky130B PDK, Magic VLSI, netgen)

rply_ex0_sky130nm

aicex

Lower your expectations on EDA software

Expect that you will spend at least 2π times more time than planned *(mostly due to software issues)*

Questions

Do

- google
- ask a someone in your class
- use the "øvingstime and labratorieøvelse" to talk to teaching assistants and hopefully me. Don't ask about future exercises
- come to the office (B311) on Fridays

Thanks!