Advanced Integrated Circuits

Lecture Notes 2025

ASICedu.com

Built on Sun Dec 7 11:45:10 PST 2025

from e96ed4f30e55a1ec25a2ccd2b014831789172ca5

Contents

1 Background

In the spring of 2024 I lectured Advanced Integrated Circuits for the third time. I have an inherent need to make things better, and the course is no different.

In the first round I noticed that little of what I had on slides, or said in lectures, made it into the student brain. That annoyed me, and I realized that probably a few things needed to change.

I think the lectures have gotten better, but I don't have any specific proof. There were 19 students that took the exam in 2024. An indication of lecture quality could be attendance. I don't have all the dates, but an average attendance of 76 % I think is pretty OK.

Date	Attendance
2024-02-02	19
2024-02-09	17
2024-02-16	16
2024-03-01	14
2024-03-07	14
2024-03-15	12
2024-03-22	13
2024-04-12	16
2024-04-19	10

For the third semester I finally felt I achieved a balance. I spent Thursday's preparing for the lecture, writing these notes, making a YouTube video (so I'll remember next year what I wanted to talk about). I passed 1k subscribers. Friday's I had the lecture and the group work.

For the group work I forced students into groups, and I forced that they for the first 5-10 minutes do a check-in. That I need to do next year too.

For the check in, they had go around in the group and answer one of the following questions:

- What is one thing that is going on in your life (personal or professional)?
- What is one thing that you're grateful for right now?
- What is something funny that happened?

The check-in led to excellent team work for those students that showed up.

Thanks to Jonathan for helping out in the exercise hours.

I love programming and automation. Not much makes me more happy than using the same source (the slide markdowns), to generate the lecture notes, to translate into the book your looking at right now.

If you find an error in what I've made, then fork analogIC, fix , commit, push and create a pull request. That way, we use the global brain power most efficiently, and avoid multiple humans spending time on discovering the same error.

Introduction 2

2.1 Who

My name is

Carsten Wulff carstenw@ntnu.no

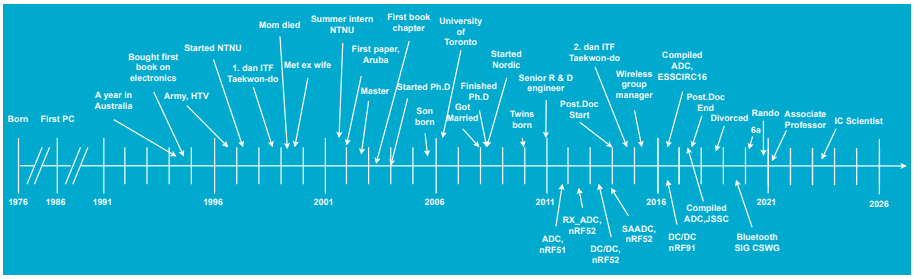
I finished my Masters in 2002, and did a Ph.D on analog-to-digital converters finished in 2008.

Since that time, I've had a three axis in my work/hobby life.

I work at Nordic Semiconductor where I've been since 2008. The first 7 years I did analog design (ADCs, DC/DCs, GPIO). The next 7 years I was the Wireless Group Manager. The Wireless group make most of the analog and RF designs for Nordic's short-range products. Now I'm the IC Scientist, and focus on technical issues with our integrated circuits that occur before we go into volume production.

I work at NTNU where I did a part time postdoc from 2014 - 2017. From 2020 I've been working on and teaching Advanced Integrated Circuits

I have a hobby trying to figure out how to make a new analog circuit design paradigm. The one we have today with schematic/simulation/layout/verification/simulation is too slow



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

In 2020 the global integrated circuit market was 437.7 billion dollars! The market is expected to grow to 1136 billion in 2028.

Integrated circuits enable pretty much all technologies.

I will be dead in approximately 50 years, and will retire in approximately 30 years. Everything I know will be gone (except for the small pieces I've left behind in videos or written word)

Someone must take over, and to do that, they need to know most of what I know, and hopefully a bit more.

That's were some of you come in. Some of you will find integrated circuits interesting to make, and in addition, you have the stamina, patience, and brain necessary to learn some of the hardest topics in the world.

> Making integrated circuits (that work reliably) is not rocket science, it's much harder.

In this course we'll focus on analog ICs, because the real world is analog, and all ICs must have some analog components, otherwise they won't work.

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.

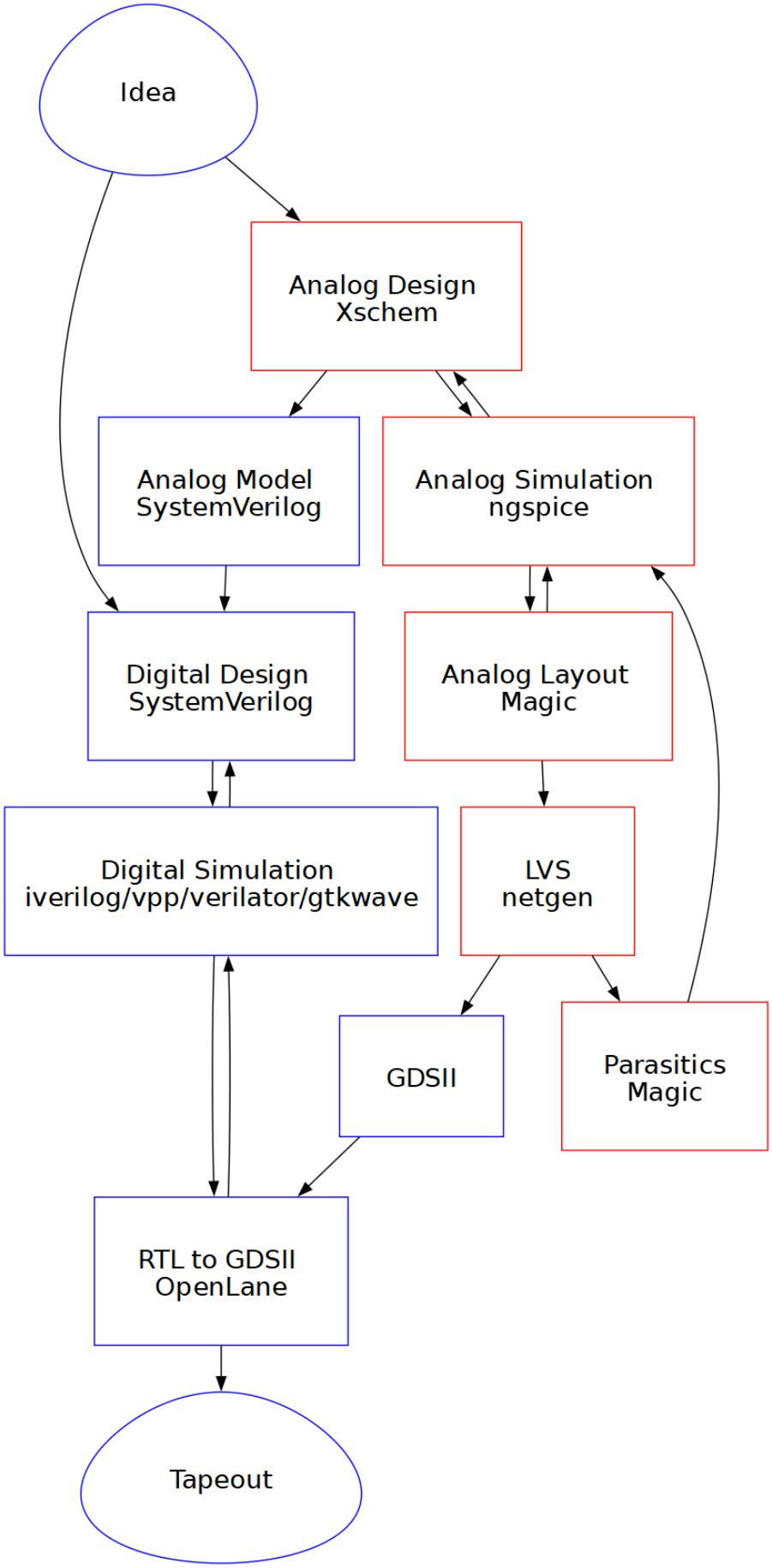
https://circuitcellar.com/insights/tech-the-future/kinget-the-world-is-analog/

The steps to make integrated circuits is split in two. We have a analog flow, and a digital flow.

It's rare to find a single human that do both flows well. Usually people choose, and I think it's based on what they like and their personality.

If you like the world to be ordered, with definite answers, then it's likely that you'll find the digital flow interesting.

If you're comfortable with not knowing, and an insatiable desire to understand how the world *really* works at a fundamental level, then it's likely that you'll find analog flow interesting.


Idea
Analog Design
Xschem
Analog Model
SystemVerilog
Analog Simulation
ngspice
Digital Design
SystemVerilog
Analog Layout
Magic
Digital Simulation
iverilog/vpp/verilator/gtkwave
LVS
netgen
GDSII
Parasitics
Magic
RTL to GDSII
OpenLane
Tapeout

2.2.1 Will you tape-out an IC?

Something that would make me really happy is if someone is able to tapeout an IC after this course.

It's now possible without signing an NDA or buying expensive software licenses.

In 2020 Google and Skywater joined forces to release a 130 nm process design kit to the public. In addition, they have fueled a renaissance of open source software tools.

Together with Efabless there are cheap alternatives, like tinytapeout, which makes it possible for a private citizen to tape-out their own integrated circuit.

Google just sponsored a GlobalFoundries 180 nm tapeout where you could tape out your circuit for free.

2.2.2 What the team needs to know to design ICs

There are a multitude of tools and skills needed to design professional ICs. It's not likely that you'll find all the skills in one human, and even if you could, one human does not have sufficient bandwidth to design ICs with all it's aspects in a reasonable timeline

That is, unless we can find a way to make ICs easier.

The skills needed are

- *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

2.2.3 Zen of IC design (stolen from Zen of Python)

When you learn something new, it's good to listen to someone that has done whatever it is before.

Here is some guiding principles that you'll likely forget.

- 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.

2.2.4 IC design mantra

To copy an old mantra I have on learning programming

> 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

And run the perl program

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

2.3 My Goal

Don't expect that I'll magically take information and put it inside your head, and you'll suddenly understand everything about making ICs.

You are the one that must teach yourself everything.

I consider my role as a guide, similar to a mountain guide. I can't carry you up the mountain, you need to walk up the mountain , but I know the safe path to take and increase the likelihood that you'll come back alive.

I want to:

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

I'm not a mind reader, I can't see inside your head. That means, you must ask questions, only by your questions can I start to understand what pieces of information is missing from your head, or maybe somehow to correct your understanding.

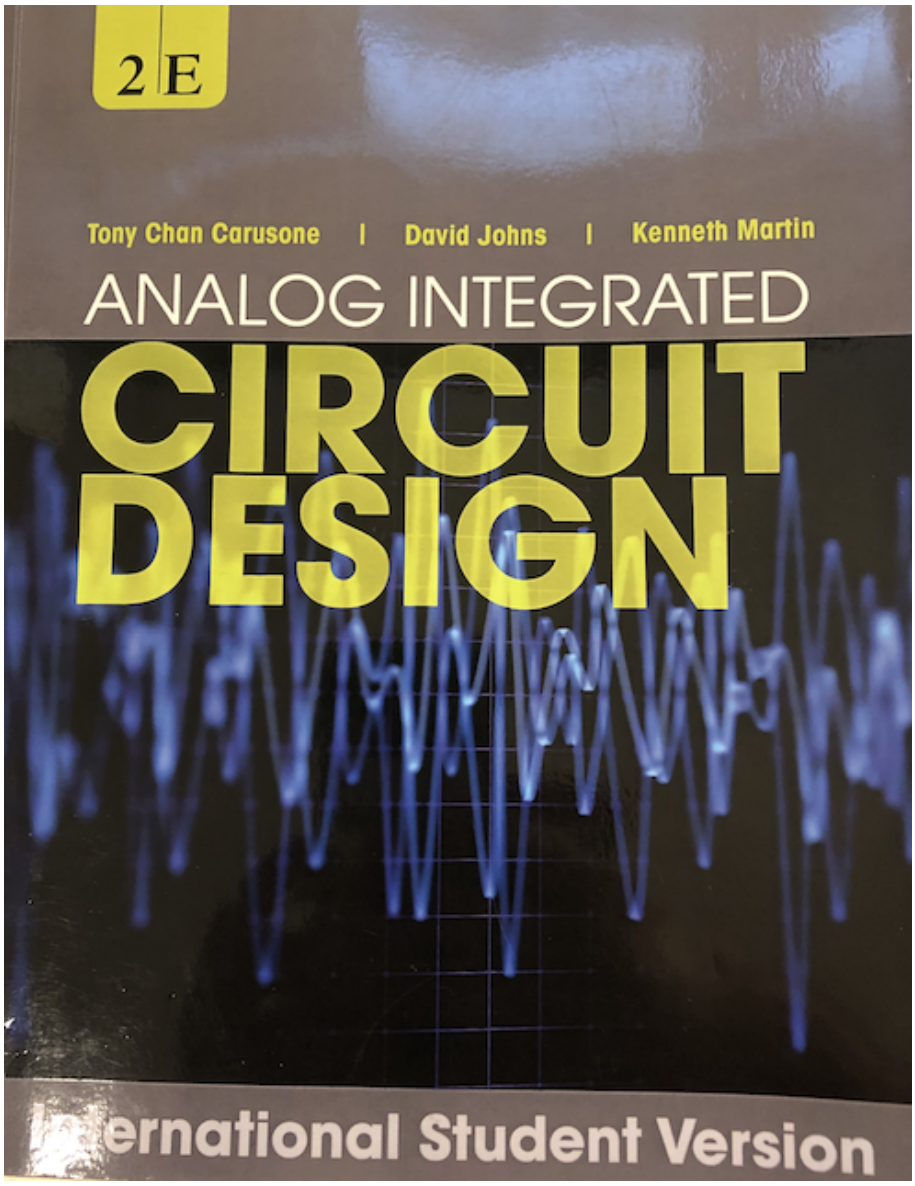
At the same time, and similar to a mountain guide, you should not assume I'm always right. I'm human, and I will make mistakes. And maybe you can correct my understanding of something. All I care about is to *really* understand how the world works, so if you think my understanding is wrong, then I'll happily discuss.

2.4 Syllabus

The syllabus will be from Analog Integrated Circuit Design (CJM) and Circuits for all seasons.

These lecture notes are a supplement to the book. I try to give some background, and how to think about electronics. It's not my goal to repeat information that you can find in the book.

Buy a hard-copy of the book if you don't have that. Don't expect to understand the book by reading the PDF.



2.5 CNR (2024)

"In an insane world, it was the sanest choice." - Sarah Connor, Terminator 2: Judgment Day

The project for 2024 is to

Design a integrated temperature sensor with digital read-out

An outline of the plan is shown below. There will be five milestones in all.

At the end of the project you will have a function that converts temperature to a digital value.

$$D = f_0(T)$$

I've broken down the challenge into three steps, first convert Temperature into a current

$$I = f_1(T)$$

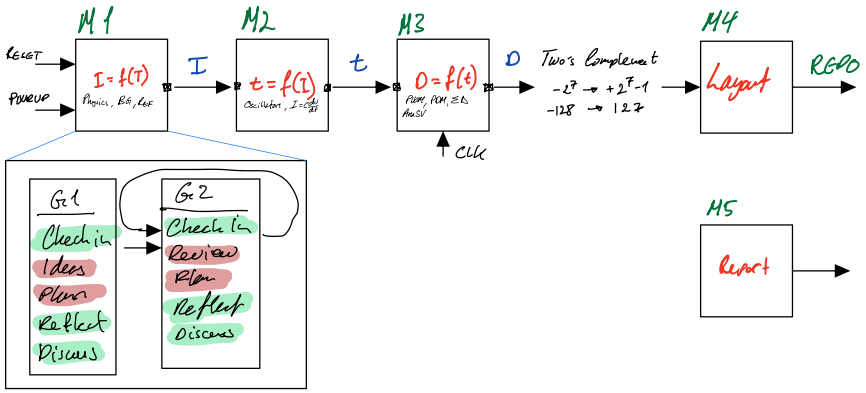
Then convert current into a time

$$t = f_2(I)$$

then time to digital

$$D = f_3(t) = f_3(f_2(f_1(T))) = f_0(T)$$

The fourth milestone is the layout, while the fifth milestone is the report.



2.5.1 Group dynamics

How you work together is important. No-one can do everything by them self. I know from experience it can be magical when bright brains come together. The collective brain can be smarter, better, faster, than anyone in the group.

That's why I think it's important not to just work in groups, but also focus on how we work in groups.

A group shall be maximum 4 members. There must be at least 3 that don't know each-other that well.

The group will meet once per week, and shall have a discussion according to the outline below.

If there is time left at the end of the group session it will be used for Q & A.

2.5.1.1 First session of milestone

During the first group session of a milestone, you will

Check-in (10 minutes)

Some example questions could be

- Share one thing that is going on in your life (personal or professional.)
- What is one thing that you are grateful for right now?
- What is something funny that happened?

Some examples answers could be: - My dog died yesterday, so I'm not feeling great today. - I woke up early, had an omelet, and went running, so I feel motivated and fantastic. - I feel *blaaah* today, motivation is lacking. - I went running yesterday and did not discover before I got home that I'd forgotten to put my pants on, even though it was -10 C.

The point of this exercise is to get to know each other a bit, and attempt to create psychological safety in the group.

Ideas (35 minutes)

Come up with ideas for how the milestone could be implemented. What circuit ideas could work?

Break (15 minutes)

Plan (20 minutes)

Sketch out who does what the next week. What's the goal for the week.

Reflect (5 minutes)

In silence, think about the group dynamics. How did it go today? What was good? What could be improved? Write down one word.

Discuss (10 minutes)

Each group member talks about their one word.

2.5.1.2 Other sessions of a milestone

You shall always Check-in, Reflect and Discuss. Although some may consider it a waste of time, it's important to improve the group dynamics.

Review (35 minutes)

Go through the plan from last week, what worked, what did not work, what should be done differently. Discuss.

Plan (20 minutes)

Sketch out who does what the next week. What's the goal for the week.

2.6 Software

We'll use professional Open source software (xschem, ngspice, sky130B PDK, Magic VLSI, netgen)

I've made a rather detailed (at least I think so myself) tutorial on how to make a current mirror with the open source tools. I strongly recommend you start with that first.

Skywater 130 nm Tutorial

I've also made some more complex examples, that can be found at the link below. There are digital logic cells, standard transistors, and few other blocks.

aicex

How to write a project report 3

3.1 Why

> Them who has a Why? in life can tolerate almost any How?

You're writing the report on the project for me to be able to see inside your head, and grade how much of the project you have understood.

- Have you learned what is to be expected?
- Do you understand what you're trying to explain?

You will work on the project in groups, however, on the report, you will write on your own.

That means, that there will be X projects reports that describe the same circuit. You shall not copy someone elses report text.

It's fine to share figures between reports, and also references.

I'm also forcing you to use a report format that matches well with what would be expected if you were to publish a paper.

Should you make a fantastic temperature sensor, and maybe even reach close to a tapeout I would strongly suggest you submit a paper to NorCas. The deadline is August 15 2024.

3.2 On writing English

Writing well is important. I would recommend that you read On writing Well.

Most of you won't buy the book, as such, a few tips.

3.2.1 Shorter is better

I can write the section title idea in many words:

> A shorter text will more elequently describe the intricacies of your thoughts than a long, distinguished, tirade of carefully, wonderfully, choosen words.

or

Shorter is better

Describe an idea with as few words as possible. The text will be better, and more readable.

3.2.2 Be careful with adjectives

Words like "very, extremely, easily, simply, . . . " don't belong in a readable text. They serve no purpose. Delete them.

3.2.3 Use paragraphs

You write a text to place ideas into anothers head. Ideas and thoughts are best communicated in chunks. I can write a dense set of text, or I can split a dense set of text into multiple paragraphs. The more I try to cram into a paragraph, for example, how magical the weather has been the last weeks, with lots of snow, and good skiing, the more difficult the paragraph is to read.

One paragraph, one thought. For example:

You write a text to place ideas into anothers head. Ideas and thoughts are best communicated in chunks.

I can write a dense set of text, or I can split a dense set of text into multiple paragraphs.

The more I try to cram into a paragraph, for example, how magical the weather has been the last weeks, with lots of snow, and good skiing, the more difficult the paragraph is to read.

3.2.4 Don't be afraid of I

If you did something, then say "I" in the text. If there were more people, then use "we".

3.2.5 Transitions are important

Sentences within a paragraph are sometimes linked. Use

- As a result,
- As such,
- Accordingly,
- Consequently,

And mix them up.

3.2.6 However, is not a start of a sentence

If you have to use "However" it should come in the middle of the sentence.

I want to go skiing, however, I cannot today due to work.

3.3 Report Structure

The sections below go through the expected structure of a report, and what the sections should contain.

3.3.1 Introduction

The purpose of the introduction is to put the reader into the right frame of mind. Introduce the problem statement, key references, the key contribution of your work, and an outline of the work presented. Think of the introduction as explaining the "Why" of the work.

Although everyone has the same assignment for the project, you have chosen to solve the problem in different ways. Explain what you consider the problem statement, and tailor the problem statement to what the reader will read.

Key references are introduced. Don't copy the paper text, write why they designed the circuit, how they chose to implement it, and what they achieved. The reason we reference other papers in the introduction is to show that we understand the current state-of-the-art. Provide a summary where state-of-the-art has moved since the original paper.

The outline should be included towards the end of the introduction. The purpose of the outline is to make this document easy to read. A reader should never be surprised by the text. All concepts should be eased into. We don't want the reader to feel like they been thrown in at the end of a long story. As such, if you chosen to solve the problem statement in a way not previously solved in a key references, then you should explain that.

A checklist for all chapters can be seen in table below.

3.3.2 Theory

It is safe to assume that all readers have read the key references, if they have not, then expect them to do so.

The purpose of the theory section is not to demonstrate that you have read the references, but rather, highlight theory that the reader probably does not know.

The theory section should give sufficient explanation to bridge the gap between references, and what you apply in this text.

3.3.3 Implementation

The purpose of the implementation is to explain what you did. How have you chosen to architect the solution, how did you split it up in analog and digital parts? Use one subsection per circuit.

For the analog, explain the design decisions you made, how did you pick the transistor sizes, and the currents. Did you make other choices than in the references? How does the circuit work?

For the digital, how did you divide up the digital? What were the design choices you made? How did you implement readout of the data? Explain what you did, and how it works. Use state diagrams and block diagrams.

Use clear figures (i.e. circuitikz), don't use pictures from schematic editors.

3.3.4 Result

The purpose of the results is to convince the reader that what you made actually works. To do that, explain testbenches and simulation results. The key to good results is to be critical of your own work. Do not try to oversell the results. Your result should speak for themself.

For analog circuits, show results from each block. Highlight key parameters, like current and delay of comparator. Demonstrate that the full analog system works.

Show simulations that demonstrate that the digital works.

3.3.5 Discussion

Explain what the circuit and results show. Be critical.

3.3.6 Future work

Give some insight into what is missing in the work. What should be the next steps?

3.3.7 Conclusion

Summarize why, how, what and what the results show.

3.3.8 Appendix

Include in appendix the necessary files to reproduce the work. One good way to do it is to make a github repository with the files, and give a link here.

3.4 Checklist

Item	Description	OK
Is the problem description clearly defined?	Describe which parts of the problem you chose to focus on. The problem description should match the results you've achieved.	
Is there a clear explanation why the problem is worth solving?	The reader might need help to understand why the problem is interesting	
Is status of state-of-the-art clearly explained?	You should make sure that you know what others have done for the same problem. Check IEEEXplore. Provide summary and references. Explain how your problem or solution is different	
Is the key contribution clearly explained?	Highlight what you've achieved. What was your contribution?	
Is there an outline of the report?	Give a short summary of what the reader is about to read	

Item	Description	OK
Is it possible for a reader skilled in the art to understand the work?	Have you included references to relevant papers	
Is the theory section too long	The theory section should be less than 10 % of the work	
Are all circuits explained?	Have you explained how every single block works?	
Are figures clear?	Remember to explain all colors, and all symbols. Explain what the reader should understand from the figure. All figures must be referenced in the text.	
Is it clear how you verified the circuit?	It's a good idea to explain what type of testbenches you used. For example, did you use dc, ac or transient to verify your circuit?	
Are key parameters simulated?	You at least need current from VDD. Think through what you would need to simulate to prove that the circuit works.	
Have you tried to make the circuit fail?	Knowing how circuits fail will increase confidence that it will work under normal conditions.	
Have you been critical of your own results?	Try to look at the verification from different perspectives. Play devil's advocate, try to think through what could go wrong, then explain how your verification proves that the circuit does work.	
Have you explained the next steps?	Imagine that someone reads your work. Maybe they want to reproduce it, and take one step further. What should that step be?	
No new information in conclusion.	Never put new information into conclusion. It's a summary of what's been done	
Story	Does the work tell a story, is it readable? Don't surprise the reader by introducing new topics without background information.	

Item	Description	OK
Chronology	Don't let the report follow the timeline of the work done. What I mean by that is don't write "first I did this, then I spent huge amount of time on this, then I did that". No one cares what the timeline was. The report does not need to follow the same timeline as the actual work.	
Too much time	How much time you spent on something should not be correlated to how much text there is in the report. No one cares how much time you spent on something. The report is about why, how, what and does it work.	
Length	A report should be concise. Only include what is necessary, but no more. Shorter is almost always better than longer.	
Template	Use IEEEtran.cls. Example can be seen from an old version of this document at https://github.com/wulffern/dic2021/tree/main/2021-10-19_project_report. Write in LaTeX. You will need LaTeX for your project and master thesis. Use http://overleaf.com if you're uncomfortable with local text editors and LaTeX.	
Spellcheck	Always use a spellchecker. Misspelled words are annoying, and may change content and context (peaked versus piqued)	

Refresher 4

4.1 There are standard units of measurement

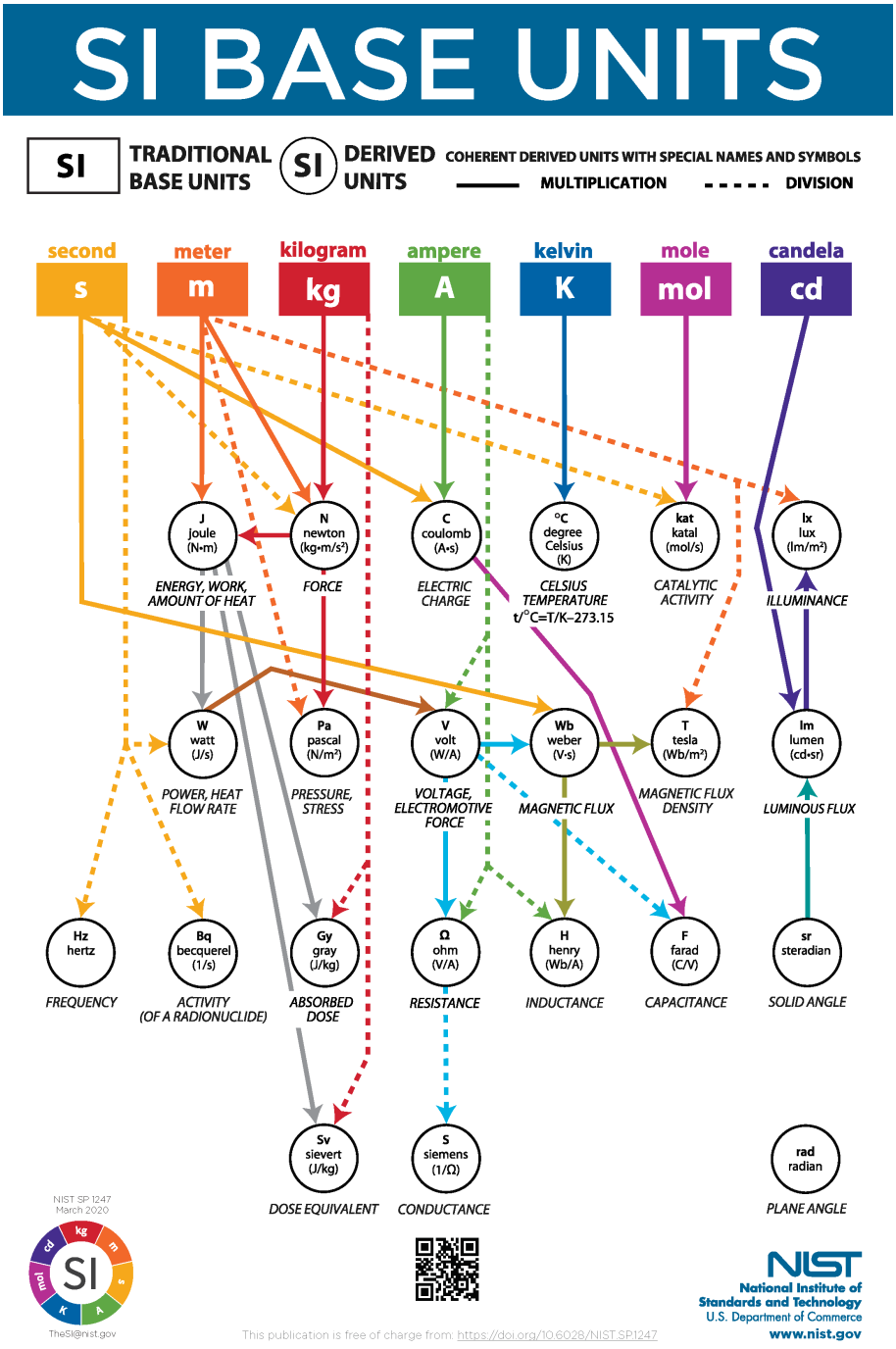
All known physical quantities are derived from 7 base units (SI units)

- second (s) : time
- meter (m) : space
- kg (kilogram) : weight
- ampere (A) : current
- kelvin (K) : temperature
- candela (cd) : luminous intensity

All other units (for example volts), are derived from the base units.

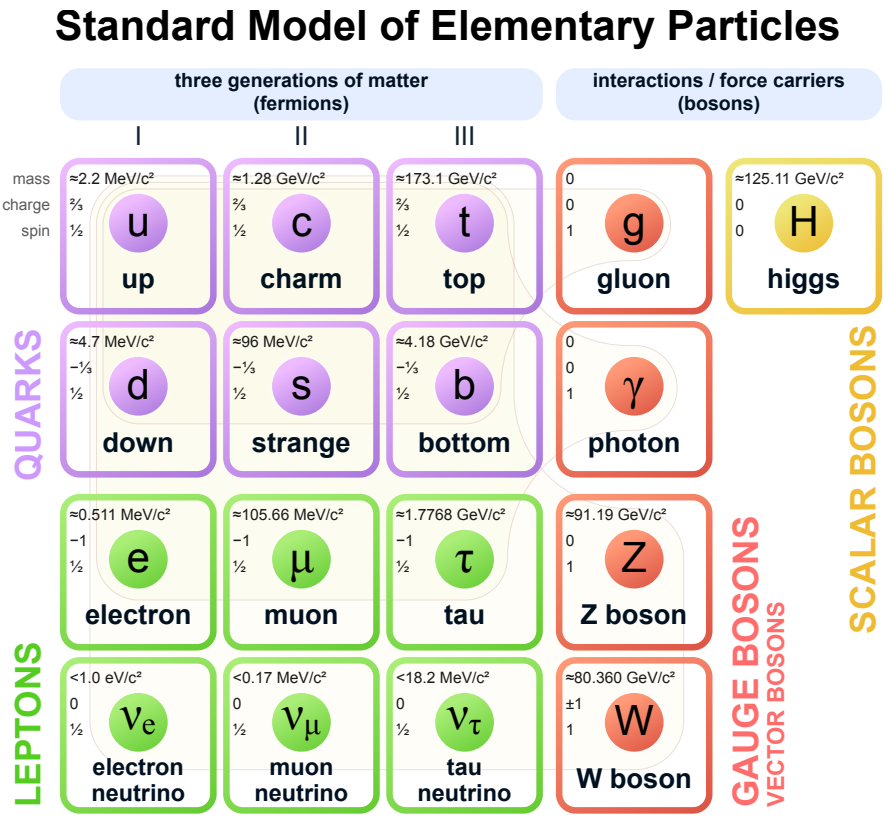
I don't go around remembering all of them, they are easily available online. When you forget the equation for charge (Q), voltage (V) and capacitance (C), look at the units below, and you can see it's $Q = CV$ *

* Although you do have to keep your symbols straight. We use "C" for Capacitance, but C can also mean Columbs. Context matters.



4.2 Electrons

Electrons are fundamental, they cannot (as far as we know), be divided into smaller parts. Explained further in the standard model of particle physics



Electrons have a negative charge of $q \approx 1.602 \times 10^{-19}$. The proton a positive charge. The two charges balance exactly! If you have a trillion electrons and a trillion protons inside a volume, the net external charge will be 0 (assuming we measure from some distance away). I find this fact absolutely incredible. There must be a fundamental connection between the charge of the proton and electron. It's insane that the charges balance out so exactly.

All electrons are the same, although the quantum state can be different.

An electron cannot occupy the same quantum state as another. This rule that applies to all Fermions (particles with spin of 1/2)

The quantum state of an electron is fully described by it's spin, momentum (p) and position in space (r).

4.3 Probability

The probability of finding an electron in a state as a function of space and time is

$$P = |\psi(r,t)|^2$$

, where ψ is named the probability amplitude, and is a complex function of space and time. In some special cases, it's

$$\psi(r,t) = Ae^{i(kr-\omega t)}$$

, where A is complex number, k is the wave number, r is the position vector from some origin, ω is the frequency and t is time.

The energy is $E = \hbar\omega$, where $\hbar = h/2\pi$ and h is Planck Constant and the momentum is $p = \hbar k$

4.4 Uncertainty principle

We cannot, with ultimate precision, determine both the position and the momentum of a particle, the precision is

$$\sigma_x \sigma_p \geq \frac{\hbar}{2}$$

From the uncertainty (Unschärfe) principle we can actually estimate the size of the atom

4.5 States as a function of time and space

The time-evolution of the probability amplitude is

$$i\hbar\frac{d}{dt}\psi(r,t) = H\psi(r,t)$$

, where H is named the Hamiltonian matrix, or the energy matrix or (if I understand correctly) the amplitude matrix of the probability amplitude to change from one state to another.

For example, if we have a system with two states, a simplified version of two electrons shared between two atoms, as in H_2, or hydrogen gas, or co-valent bonds, then the Hamiltonian is a 2 x 2 matrix. And the ψ is a vector of $[\psi_1, \psi_2]$

Computing the solution to the Schrodinger Equation can be tricky, because you must know the number of relevant states to know the vector size of ψ and the matrix size of H. In addition, the H can be a function of time and space (I think).

Compared to the equations of electric fields, however, Schrodinger is easy, it's a set of linear differential equations.

4.6 Allowed energy levels in atoms

Solutions to Schrodinger result in quantized energy levels for an electron bound to an atom.

Take hydrogen, the electron bound to the proton can only exists in quantized energy levels. The lowest energy state can have two electrons, one with spin up, and one with spin down.

From Schrodinger you can compute the energy levels, which most of us did at some-point, although now, I can't remember how it was done. That's not important. The important is to internalize that the energy levels in bound electrons are discrete.

Electrons can transition from one energy level to another by external influence, i.e temperature, light, or other.

The probability of a state transition (change in energy) can be determined from the probability amplitude and Schrodinger.

4.7 Allowed energy levels in solids

If I have two silicon atoms spaced far apart, then the electrons can have the same spin and same momentum around their respective nuclei. As I bring the atoms closer, however, the probability amplitudes start to interact (or the dimensions of the Hamiltonian matrix grow), and there can be state transitions between the two electrons.

The allowed energy levels will split. If I only had two states interacting, the Hamiltonian could be

$$H = \begin{bmatrix} A & 0 \\ 0 & -A \end{bmatrix}$$

and the new energy levels could be

$$E_1 = E_0 + A$$

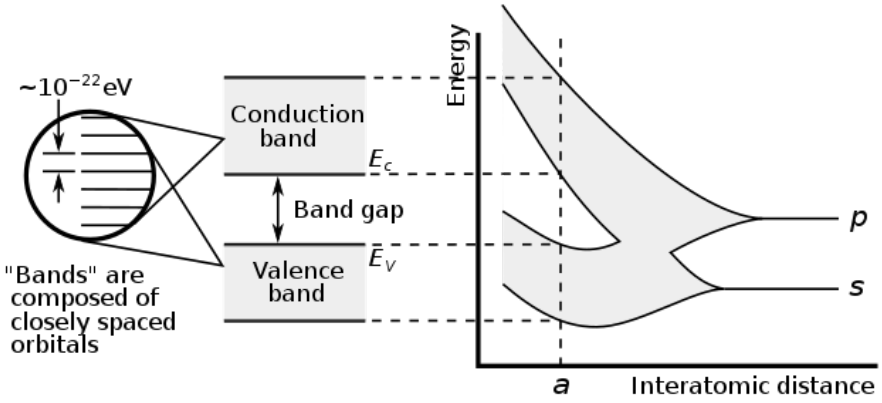
and

$$E_2 = E_0 - A$$

In a silicon crystal we can have trillions of atoms, and those that are close, have states that interact. **That's why crystals stay solids**. All chemical bonds are states of electrons interacting! Some are

strong (co-valent bonds), some are weaker (ionic bonds), but it's all quantum states interacting.

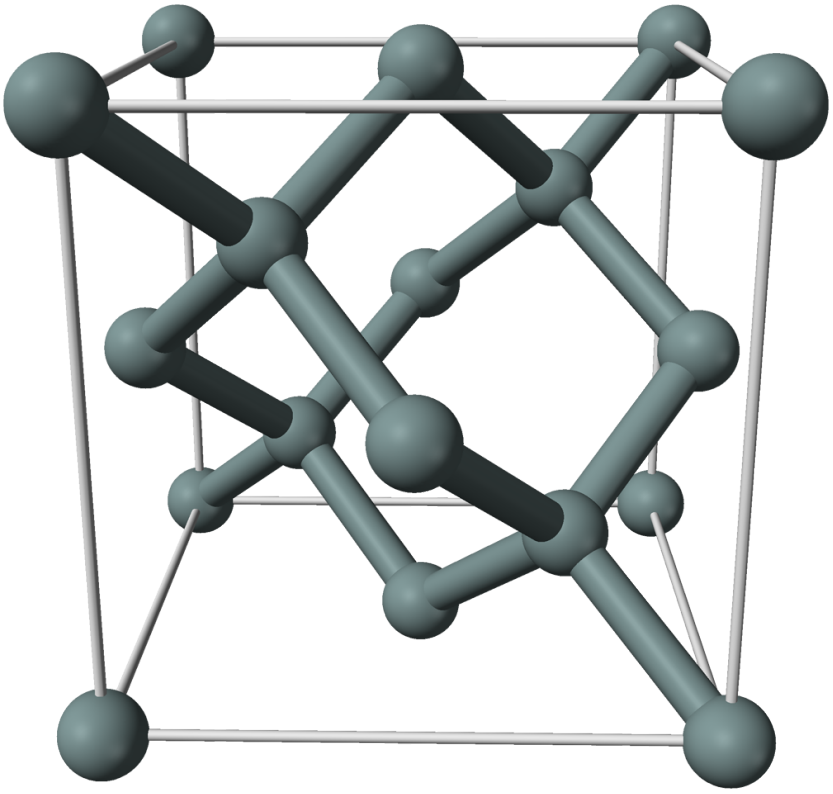
The discrete energy levels of the electron transition into bands of allowed energy states.



For a crystal, the allowed energy bands is captured in the band structure

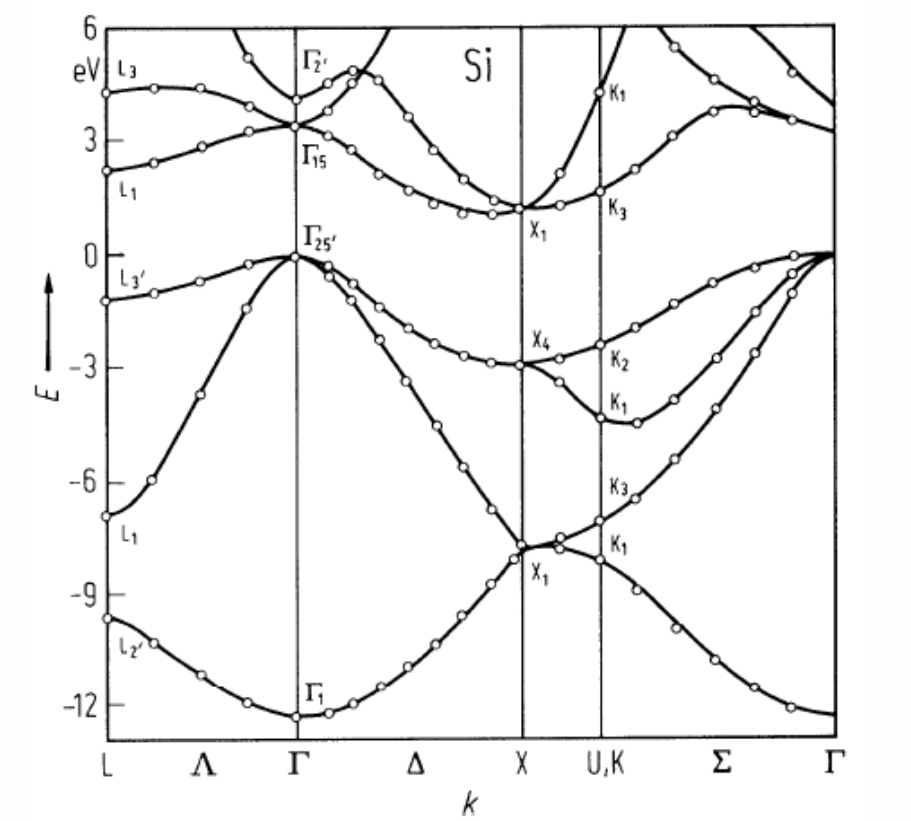
4.8 Silicon Unit Cell

A silicon crystal unit cell is a diamond faced cubic with 8 atoms in the corners spaced at 0.543 nm, 6 at the center of the faces, and 4 atoms inside the unit cell at a nearest neighbor distance of 0.235 nm.



4.9 Band structure

The full band structure of a silicon unit cell is complicated, it's a 3 dimensional concept



4.10 Valence band and Conduction band

For bulk silicon we simplify, and we think of two bands, the conduction band, and valence band

In the conduction band (E_C) is the lowest energy where electrons are free (not bound to atoms). The valence band (E_V) is the highest band where electrons are bound to silicon atoms.

The difference between E_C and E_V is a property of the material we've named the band gap.

$$E_G = E_C - E_V$$

4.11 Fermi level

From Wikipedia's Fermi level

> In band structure theory, used in solid state physics to analyze the energy levels in a solid, the Fermi level can be considered to be a hypothetical energy level of an electron, such that at thermodynamic equilibrium this

energy level would have a 50% probability of being occupied at any given time

The Fermi level is closely linked to the Fermi-Dirac distribution

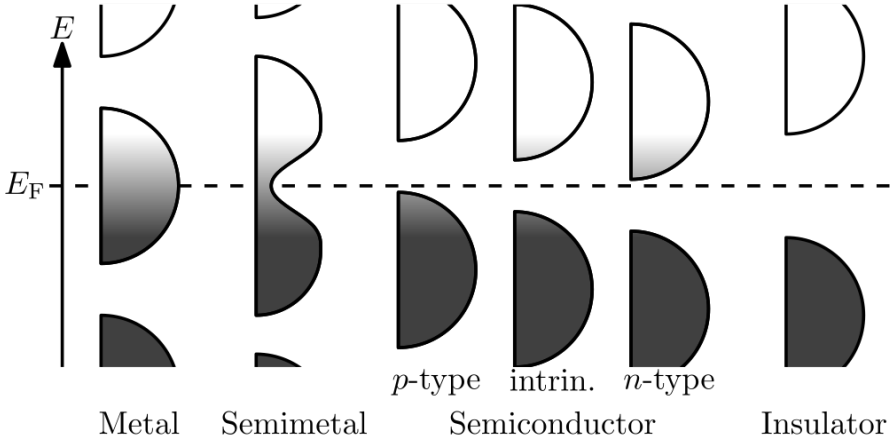
$$f(E) = \frac{1}{e^{(E-E_F)/kT} + 1}$$

If the energy of the state is more than a few kT away from the Fermi-level, then

$$f(E) \approx e^{(E_F-E)/kT}$$

4.12 Metals

In metals, the band splitting of the energy levels causes the valence band and conduction band to overlap.



As such, electrons can easily transition between bound state and free state. As such, electrons in metals are shared over large distances, and there are many electrons readily available to move under an applied field, or difference in electron density. That's why metals conduct well.

4.13 Insulators

In insulating materials the difference between the conduction band and the valence band is large. As a result, it takes a large energy to excite electrons to a state where they can freely move.

That's why glass is transparent to optical frequencies. Visible light does not have sufficient energy to excite electrons from a bound state.

That's also why glass is opaque to ultra-violet, which has enough energy to excite electrons out of a bound state.

Based on these two pieces of information you could estimate the bandgap of glass.

4.14 Semiconductors

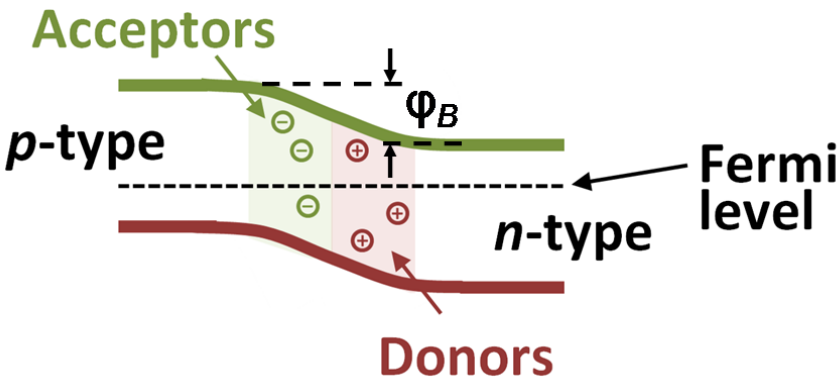
In a silicon the bandgap is lower than an insulator, approximately

$$E_G = 1.12\ eV$$

At room temperature, that allows a small number of electrons to be excited into the conduction band, leaving behind a "hole" in the valence band.

4.15 Band diagrams

A band diagram or energy level diagrams shows the conduction band energy and valence band energy as a function of distance in the material.



The horizontal axis is the distance, the vertical axis is the energy.

The figure shows a PN-junction

4.16 Density of electrons/holes

There are two components needed to determine how many electrons are in the conduction band. The density of available states, and the probability of an electron to be in that quantum state.

The probability is the Fermi-Dirac distribution. The density of available states is a complicated calculation from the band-structure of silicon. for details.

$$n_e = \int_{E_C}^{\infty} N(E) f(E) dE$$

The Fermi level is assumed to be independent of energy level, so we can write

$$n_e = e^{E_F/kT} \int_{E_C}^{\infty} N(E) e^{-E/kT} dE$$

for the density of electrons in the conduction band.

4.17 Fields

There are equations that relate electric field, magnetic field, charge density and current density to each-other.

$$\oint_{\partial\Omega} \mathbf{E} \cdot d\mathbf{S} = \frac{1}{\epsilon_0} \iiint_V \rho \cdot dV$$

,relates net electric flux to net enclosed electric charge

$$\oint_{\partial\Omega} \mathbf{B} \cdot d\mathbf{S} = 0$$

,relates net magnetic flux to net enclosed magnetic charge

$$\oint_{\partial\Sigma} \mathbf{E} \cdot d\ell = -\frac{d}{dt} \iint_{\Sigma} \mathbf{B} \cdot d\mathbf{S}$$

,relates induced electric field to changing magnetic flux

$$\oint_{\partial\Sigma} \mathbf{B} \cdot d\ell = \mu_0 \left(\iint_{\Sigma} \mathbf{J} \cdot d\mathbf{S} + \epsilon_0 \frac{d}{dt} \iint_{\Sigma} \mathbf{E} \cdot d\mathbf{S} \right)$$

,relates induced magnetic field to changing electric flux and to current

These are the Maxwell Equations, and are non-linear time dependent differential equations.

Under the best of circumstances they are fantastically hard to solve! But it's how the real world works.

The permittivity of free space is defined as

$$\epsilon_0 = \frac{1}{\mu_0 c^2}$$

, where c is the speed of light, and μ_0 is the vacuum permeability, which, in SI units, is now

$$\mu_0 = \frac{2\alpha}{q^2}\frac{h}{c}$$

, where α is the fine structure constant.

4.18 Voltage

The electric field has units voltage per meter, so the electric field is the derivative of the voltage as a function of space.

$$E = \frac{dV}{dx}$$

4.19 Current

Current has unit A and charge C has unit As, so the current is the number of charges passing through a volume per second.

The current density J has units A/m^2 and is often used, since we can multiply by the surface area of a conductor, if the current density is uniform.

$$I = A \times J$$

4.20 Drift current

Charges in an electric field will give rise to a drift current.

We know from Newtons laws that force equals mass times acceleration

$$\vec{F} = m\vec{a}$$

If we assume a zero, or constant magnetic field, the force on a particle is

$$\vec{F} = q\vec{E}$$

The current density is then

$$\vec{J} = q\vec{E} \times n \times \mu$$

where n is the charge density, and μ is the mobility (how easily the charges move) and has units $[m^2/Vs]$

Assuming

$$E = V/m$$

, we could write

$$J = \frac{C}{m^3}\frac{V}{m}\frac{m^2}{Vs} = \frac{C}{s}m^{-2}$$

So multiplying by an area

$$A = Bm^2$$

$$I = qn\mu BV$$

and we can see that the conductance

$$G = qn\mu B$$

, and since

$$G = 1/R$$

, where R is the resistance, we have

$$I = GV \Rightarrow V = RI$$

Or Ohms law

4.21 Diffusion current

A difference in charge density will give rise to a diffusion current, and the current density is

$$J = -qD_n \frac{d\rho}{dx}$$

,where D_n is a diffusion constant, and ρ is the charge density.

4.22 Why are there two currents?

I struggled with the concepts diffusion current and drift current for a long time. Why are there two types of current? It was when I read The Schrödinger Equation in a Classical Context: A Seminar on Superconductivity I realised that the two types of current come directly from the Schrodinger equation, there is one component related to the electric field (potential energy) and a component related to the momentum (kinetic energy).

In the absence of an electric field electrons will still jump from state to state set by the probabilities of the Hamiltonian. If there are more electrons in an area, then it will seem like there is an average movement of charges away from that area. That's how I think about the equation above. We can kinda see it from the Schrödinger equation below.

$$-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial^2 x}\psi(x,t) + V(x)\psi(x,t) = i\hbar\frac{\partial}{\partial t}\psi(x,t)$$

4.23 Currents in a semiconductor

Both electrons, and holes will contribute to current.

Electrons move in the conduction band, and holes move in the valence band.

Both holes and electrons can only move if there are available quantum states.

For example, if the valence band is completely filled (all states filled), then there can be no current.

To compute the total current in a semiconductor one must compute

$$I = I_{n_{drift}} + I_{n_{diffusion}} + I_{p_{drift}} + I_{p_{diffusion}}$$

where n denotes electrons, and p denote holes.

4.24 Resistors

We can make resistors with metal and silicon (a semiconductor)

In metal the dominant carrier depends on the metal, but it's usually electrons. As such, one can often ignore the hole current.

In a semiconductor the dominant carrier depends on the Fermi level in relation to the conduction band and valence band. If the Fermi level is close to the valence band the dominant carrier will be holes. If the Fermi level is close to the conduction band, the dominant carrier will be electrons.

That's why we often talk about "majority carriers" and "minority carriers", both are important in semiconductors.

4.25 Capacitors

A capacitor resists a change in voltage.

$$I = C\frac{dV}{dt}$$

and store energy in an electric field between two conductors with an insulator between.

4.26 Inductors

An inductor resist a change in current.

$$V = L\frac{dI}{dt}$$

and store energy in the magnetic fields in a loop of a conductor.

Diodes 5

5.1 Why

Diodes are a magical * semiconductor device that conduct current in one direction. It's one of the fundamental electronics components, and it's a good idea to understand how they work.

If you don't understand diodes, then you won't understand transistors, neither bipolar, or field effect transistors.

A useful feature of the diode is the exponential relationship between the forward current, and the voltage across the device.

To understand why a diode works it's necessary to understand the physics behind semiconductors.

This paper attempts to explain in the simplest possible terms how a diode works †

5.2 Silicon

Integrated circuits use single crystalline silicon. The silicon crystal is grown with the Czochralski method which forms a ingot that is cut into wafers. The wafer is a regular silicon crystal, although, it is not perfect.

A silicon crystal unit cell, as seen in Figure 1 is a diamond faced cubic with 8 atoms in the corners spaced at 0.543 nm, 6 at the center of the faces, and 4 atoms inside the unit cell at a nearest neighbor distance of 0.235 nm.

* It doesn't stop being magic just because you know how it works. Terry Pratchett, The Wee Free Men

† Simplify as much as possible, but no more. Einstein

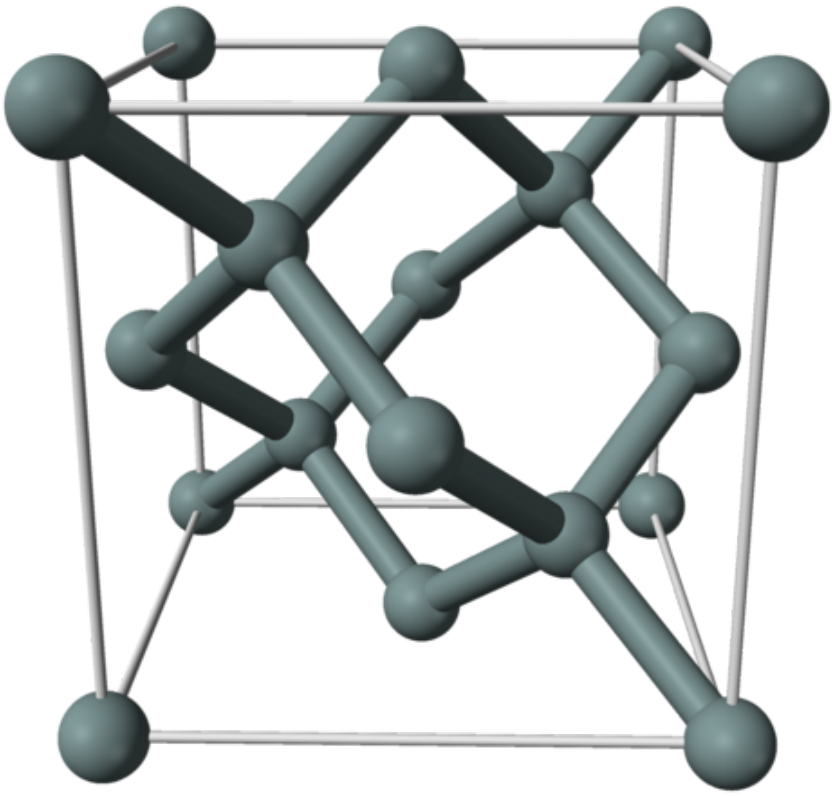


Figure 1: Silicon crystal unit cell

As you hopefully know, the energy levels of an electron around a positive nucleus are quantized, and we call them orbitals (or shells). For an atom far away from any others, these orbitals, and energy levels are distinct. As we bring atoms closer together, the orbitals start to interact, and in a crystal, the distinct orbital energies split into bands of allowed energy states. No two electrons, or any Fermion (spin of 1/2), can occupy the same quantum state. We call the outermost "shared" orbitial, or band, in a crystal the valence band. Hence covalent bonds.

If we assume the crystal is perfect, then at 0 Kelvin all electrons will be part of covalent bonds. Each silicon atom share 4 electrons with its neighbors. I think what we really mean when we say "share 4 electrons" is that the wave-functions of the outer orbitals interact, and we can no longer think of the orbitals as belonging to either of the silicon nuclei. All the neighbors atoms "share" electrons, and nowhere is there an vacant state, or a hole, in the valence band. If such a crystal were to exist, it would not conduct any current, as the charges cannot move.

In a atom, or a crystal, there are also higher energy states where the carriers are "free" to move. We call these energy levels, or bands of energy levels, conduction bands. In singular form "conduction

band", refers to the lowest available energy level where the electrons are free to move.

Due to imperfectness of the silicon crystal, and non-zero temperature, there will be some electrons that achieve sufficient energy to jump to the conduction band. The electrons in the conduction band leave vacant states, or holes, in the valence band.

Electrons can move both in the conduction band, as free electrons, and in the valence band, as a positive particle, or hole.

5.3 Intrinsic carrier concentration

The intrinsic carrier concentration of silicon, or how many free electrons and holes at a given temperature, is given by

$$n_i = \sqrt{N_c N_v} e^{-\frac{E_g}{2kT}} \tag{1}$$

where E_g is the bandgap energy of silicon (approx 1.12 eV), k is Boltzmann's constant, T is the temperature in Kelvin, N_c is the density of states in conduction band, and N_v is the density of states in the valence band.

The density of states are

$$N_c = 2\left[\frac{2\pi kTm_n^*}{h^2}\right]^{3/2} \quad N_v = 2\left[\frac{2\pi kTm_p^*}{h^2}\right]^{3/2}$$

where h is Planck's constant, m_n^* is the effective mass of electrons, and m_p^* is the effective mass of holes.

In [1] they claim the intrinsic carrier concentration is a constant, although they do mention n_i doubles every 11 degrees Kelvin.

In BSIM 4.8 [2] the intrinsic carrier concentration is

$$n_i = 1.45e10\frac{TNOM}{300.15}\sqrt{\frac{T}{300.15}}\exp^{21.5565981-\frac{E_g}{2kT}}$$

Comparing the three models in Figure 2, we see the shape of BSIM and the full equation is almost the same, while the "doubling every 11 degrees" is just wrong.

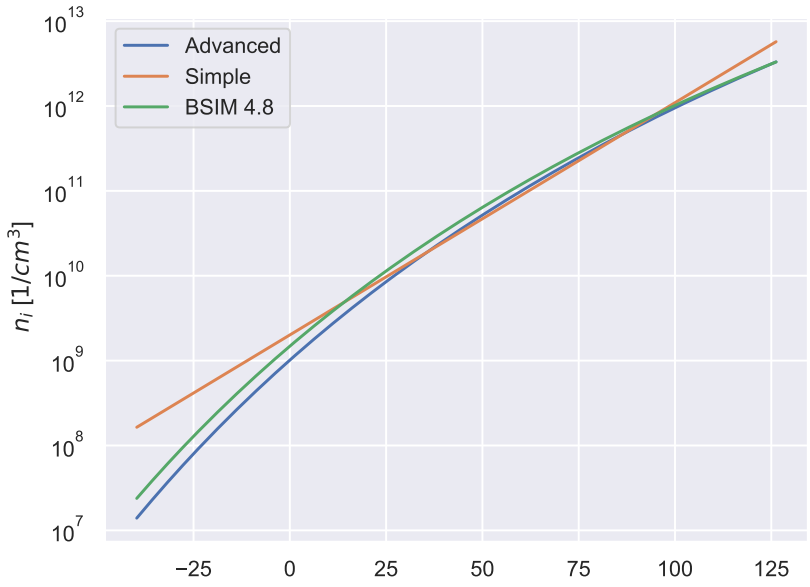


Figure 2: Intrinsic carrier concentration versus temperature

At room temperature the intrinsic carrier consentration is approximately $n_i = 1 \times 10^{16}$ carriers/m^3.

That may sound like a big number, however, if we calculate the electrons per um^3 it's $n_i = \frac{1\times10^{16}}{(1\times10^6)^3}$ carriers/μm^3 < 1, so there are really not that many free carriers in intrinsic silicon.

But where does Eq (1) come from? I find it unsatisfying if I don't understand where things come from. I like to understand why there is an exponential, or effective mass, or Planck's constant. If you're like me, then read the next section. If you don't care, and just want to memorize the equations, or indeed the number of intrinsic carrier concentration number at room temperature, then skip the next section.

5.4 It's all quantum

There are two components needed to determine how many electrons are in the conduction band. The density of available states, and the probability of an electron to be in that quantum state.

For the density of states we must turn to quantum mechanics. The probability amplitude of a particle can be described as

$$\psi = Ae^{i(\mathbf{k}\mathbf{r}-\omega t)}$$

where k is the wave number, and ω is the angular frequency, and $\mathbf{r}$ is a spatial vector.

In one dimension we could write $\psi(x,t) = Ae^{i(kx-\omega t)}$

In classical physics we described the Energy of the system as

$$\frac{1}{2m}p^2 + V = E$$

where $p = mv$, m is the mass, v is the velocity and V is the potential.

In the quantum realm we must use the Schrodinger equation to compute the time evolution of the Energy, in one space dimension

$$-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial^2 x}\psi(x,t) + V(x)\psi(x,t) = i\hbar\frac{\partial}{\partial t}\psi(x,t)$$

where m is the mass, V is the potential, $\hbar = h/2\pi$.

We could rewrite the equation above as

$$\widehat{H}\psi(x,t) = i\hbar\frac{\partial}{\partial t}\psi(x,t) = \widehat{E}\psi(x,t)$$

where $\widehat{H}$ is sometimes called the *Hamiltonian* and is an operator, or something that act on the wave-function. I recently read Feynman's Lectures on Physics, and Feynman called the Hamiltonian the *Energy Matrix* of a system. I like that better. The $\widehat{E}$ is the energy operator, something that operates on the wave-function to give the Energy.

We could re-arrange

$$[\widehat{H} - \widehat{E}]\psi(r,t) = 0$$

This is an equation with at least 5 unknowns, the space vector in three dimensions, time, and the energy matrix $\widehat{H}$. It turns out, that the energy matrix depends on the system. The energy matrix further up is for one free electron. For an atom, the energy matrix will have more dimensions to describe the possible quantum states.

I was watching Quantum computing in the 21st Century and David Jamison mentioned that the largest system we could today compute would be a system with about 30 electrons. So although we know exactly how the equations of quantum mechanics appear to be, and they've proven extremely successful, we must make simplifications before we can predict how electrons behave in complicated systems

like the silicon lattice with approximately 0.7 trillion electrons per cube micro meter. You can check the calculation

$$\left[\frac{1\ \mu\text{m}}{0.543\ \text{nm}}\right]^3 \times 8 \text{ atoms per unit cell} \times 14 \text{ electrons per atom}$$

5.4.1 Density of states

To compute "how many Energy states are there per unit volume in the conduction band", or the "density of states", we start with the three dimensional Schrodinger equation for a free electron

$$-\frac{\hbar^2}{2m}\nabla^2\psi = E\psi$$

I'm not going to repeat the computation here, but rather paraphrase the steps. You can find the full derivation in Solid State Electronic Devices.

The derivation starts by computing the density of states in the k-space, or momentum space,

$$N(dk) = \frac{2}{(2\pi)^p}dk$$

Where p is the number of dimensions (in our case 3).

Then uses the band structure $E(k)$ to convert to the density of states as a function of energy $N(E)$. The simplest band structure, and a approxmiation of the lowest conduction band is

$$E(k) = \frac{\hbar^2 k^2}{2m^*}$$

where m^* is the effective mass of the particle. It is within this effective mass that we "hide" the complexity of the actual three-dimensional crystal structure of silicon.

The effective mass when we compute the density of states is

$$m^* = \frac{\hbar^2}{\frac{d^2E}{dk^2}}$$

as such, the effective mass depends on the localized band structure of the silicon unit cell, and depends on direction of movement, strain of the silicon lattice, and probably other things.

In 3D, once we use the above equations, one can compute that the density of states per unit energy is

$$N(E)dE = \frac{2}{\pi^2}\frac{{m^*}^{3/2}}{\hbar^2}E^{1/2}dE$$

In order to find the number of electrons, we need the probability of an electron being in a quantum state, which is given by the Fermi-Dirac distribution

$$f(E) = \frac{1}{e^{(E-E_F)/kT}+1} \tag{2}$$

where E is the energy of the electron, E_F is the Fermi level or checmical potential, k is Boltzmann's constant, and T is the temperature in Kelvin.

Fun fact, the Fermi level difference between two points is what you measure with a voltmeter.

If the $E - E_F > kT$, then we can start to ignore the +1 and the probability reduces to

$$f(E) = \frac{1}{e^{(E-E_F)/kT}} = e^{(E_F-E)/kT}$$

A few observiation on the Fermi-Dirac distribution. If the Energy of a particle is at the Fermi level, then $f(E) = \frac{1}{2}$, or a 50 % probability.

In a metal, the Fermi level lies within a band, as the conduction band and valence band overlap. As a result, there are a bunch of free electrons that can move around. Metal does not have the same type of covalent bonds as silicon, but electrons are shared between a large part of the metal structure. I would also assume that the location of the Fermi level within the band structure explains the difference in conductivity of metals, as it would determined how many electrons are free to move.

In an insulator, the Fermi level lies in the bandgap between valence band and conduction band, and usually, the bandgap is large, so there is a low probability of finding electrons in the conduction band.

In a semiconductor we also have a bandgap, but much lower energy than an insulator. If we have thermal equilibrium, no external forces, and we have an un-doped (intrinsic) silicon semiconductor, then the fermi level E_F lies half way between the conduction band edge E_C and the valence band edge E_V.

The bandgap is defined as the $E_C - E_V = E_g$, and we can use that to get $E_F - E_C = E_C - E_g/2 - E_C = -E_g/2$. This is why the bandgap of silicon keeps showing up in our diode equations.

The number of electrons per delta energy will then be given by

$$N_e dE = N(E) f(E) dE$$

, which can be integrated to get

$$n_e = 2\left(\frac{2\pi m^* kT}{h^2}\right)^{3/2} e^{(E_F - E_C)/kT}$$

For intrinsic silicon at thermal equlibrium, we could write

$$n_0 = 2\left(\frac{2\pi m^* kT}{h^2}\right)^{3/2} e^{-E_g/(2kT)} \tag{3}$$

As we can see, Equation (3) has the same coefficients and form as the computation in Equation (1). The difference is that we also have to account for holes. At thermal equilibrium and intrinsic silicon $n_i^2 = n_0 p_0$.

5.4.2 How to think about electrons (and holes)

I've come to the realization that to imagine electrons as balls moving around in the silicon crystal is a bad mental image.

For example, for a metal-oxide-semiconductor field effect transistor (MOSFET) it is not the case that the electrons that form the inversion layer under strong inversion come from somewhere else. They are already at the silicon surface, but they are bound in covalent bonds (there are literaly trillions of bound electrons in a typical transistor).

What happens is that the applied voltage at the gate shifts the energy bands close to the surface (or bends the bands in relation to the Fermi level), and the density of carriers in the conduction band in that location changes, according to the type of derivations above.

Once the electrons are in the conduction band, then they follow the same equations as diffusion of a gas, Fick's law of diffusion. Any charge concentration difference will give rise to a diffusion current given by

$$J_{\text{diffusion}} = -qD_n \frac{\partial \rho}{\partial x} \tag{4}$$

where J is the current density, q is the charge, ρ is the charge density, and D is a diffusion coefficient that through the Einstein relation can be expressed as $D = \mu kT$, where mobility $\mu = v_d/F$ is the ratio of drift velocity v_d to an applied force F.

To make matters more complicated, an inversion layer of a MOSFET is not in three dimensions, but rather a two dimensional electron gas, as the density of states is confined to the silicon surface. As such, we should not expect the mobility of bulk silicon to be the same as the mobility of a MOSFET transistor.

5.5 Doping

We can change the property of silicon by introducing other elements, something we've called doping. Phosphor has one more electron than silicon, Boron has one less electron. Injecting these elements into the silicon crystal lattice changes the number of free electron/holes.

These days, we usually dope with ion implantation, while in the olden days, most doping was done by diffusion. You'd paint something containing Boron on the silicon, and then heat it in a furnace to "diffuse" the Boron atoms into the silicon.

If we have an element with more electrons we call it a donor, and the donor concentration N_D.

The main effect of doping is that it changes the location of the Fermi level at thermal equilibirum. For donors, the Fermi level will shift closer to the conduction band, and increase the probabilty of free electrons, as determined by Equation (2).

Since the crystal now has an abundance of free electrons, which have negative charge, we call it n-type.

If the element has less electrons we call it an acceptor, and the acceptor concentration N_A. Since the crystal now has an abundance of free holes, we call it p-type.

The doped material does not have a net charge, however, as it's the same number of electrons and protons, so even though we dope silicon, it does remain neutral.

The doping concentrations are larger than the intrinsic carrier concentration, from maybe 10^{21} to 10^{27} carriers/m^3. To separate between these concentrations we use $p-$, p, $p+$ or $n-$, n, $n+$.

The number of electrons and holes in a n-type material is

$$n_n = N_D \, , \, p_n = \frac{n_i^2}{N_D}$$

and in a p-type material

$$p_p = N_A \, , \, n_p = \frac{n_i^2}{N_A}$$

In a p-type crystal there is a majority of holes, and a minority of electrons. Thus we name holes majority carriers, and electrons minority carriers. For n-type it's opposite.

5.6 PN junctions

Imagine an n-type material, and a p-type material, both are neutral in charge, because they have the same number of electrons and protons. Within both materials there are free electrons, and free holes which move around constantly.

Now imagine we bring the two materials together, and we call where they meet the junction. Some of the electrons in the n-type will wander across the junction to the p-type material, and visa versa. On the opposite side of the junction they might find an opposite charge, and might get locked in place. They will become stuck.

After a while, the diffusion of charges across the junction creates a depletion region with immobile charges. Where as the two materials used to be neutrally charged, there will now be a build up of negative charge on the p-side, and positive charge on the n-side.

5.6.1 Built-in voltage

The charge difference will create a field, and a built-in voltage will develop across the depletion region.

The density of free electrons in the conduction band is

$$n = \int_{E_C}^{\infty} N(E) f(E) dE$$

, where $N(E)$ is the density of states, and $f(E)$ is a probability of a electron being in that state (Equation (2)).

We could write the density of electrons on the n-side as

$$n_n = e^{E_{F_n}/kT} \int_{E_C}^{\infty} N_n(E) e^{-E/kT} dE$$

since the Fermi level is independent of the energy state of the electrons (I think).

The density of electrons on the p-side could be written as

$$n_p = e^{E_{F_p}/kT} \int_{E_C}^{\infty} N_p(E) e^{-E/kT} dE$$

If we assume that the density of states, $N_n(E)$ and $N_p(E)$ are the same, and the temperature is the same, then

$$\frac{n_n}{n_p} = \frac{e^{E_{F_n}/kT}}{e^{E_{F_p}/kT}} = e^{(E_{F_n} - E_{F_p})/kT}$$

The difference in Fermi levels is the built-in voltage multiplied by the unit charge.

$$E_{F_n} - E_{F_p} = q\Phi$$

and by substituting for the minority carrier concentration on the p-side we get

$$\frac{N_A N_D}{n_i^2} = e^{q\Phi_0/kT}$$

or rearranged to

$$\Phi_0 = \frac{kT}{q} ln \left(\frac{N_A N_D}{n_i^2} \right)$$

5.6.2 Current

The derivation of current is a bit involved, but let's try.

The hole concentration on the p-side and n-side could be written as

$$\frac{p_p}{p_n} = e^{-q\Phi_0/kT}$$

The negative sign is because the built in voltage is positive on the n-type side

Asssume that $-x_{p0}$ is the start of the junction on the p-side, and x_{n0} is the start of the junction on the n-side.

Assume that we lift the p-side by a voltage qV

Then the hole concentration would change to

$$\frac{p(-x_{p0})}{p(x_{n0})} = e^{q(V-\Phi_0)/kT}$$

while on the n-side the hole concentration would be

$$\frac{p(x_{n0})}{p_n} = e^{qV/kT}$$

So the excess hole concentration on the n-side due to an increase of V would be

$$\Delta p_n = p(x_{n0}) - p_n = p_n \left(e^{qV/kT} - 1\right)$$

The diffusion current density, given by Equation (4) states

$$J(x_n) = -qD_p \frac{\partial \rho}{\partial x}$$

Thus we need to know the charge density as a function of x. I'm not sure why, but apparently it's

$$\partial\rho(x_n) = \Delta p_n e^{-x_n/L_p}$$

where L_p is a diffusion length. This equation smells to me like a simplified model of reality, I'm not sure how much it's based on fundamental physics.

Anyhow, we can now compute the current density, and need only compute it for $x_n = 0$, so you can show it's

$$J(0) = q\frac{D_p}{L_p} p_n \left(e^{qV/kT} - 1\right)$$

which start's to look like the normal diode equation. The p_n is the minority concentration of holes on the n-side, which we've before estimated as $p_n = \frac{n_i^2}{N_D}$

We've only computed for holes, but there will be electron transport from the p-side to the n-side also.

We also need to multiply by the area of the diode to get current from current density. The full equation thus becomes

$$I = qAn_i^2 \left(\frac{1}{N_A} \frac{D_n}{L_n} + \frac{1}{N_D} \frac{D_p}{L_p} \right) \left[e^{qV/kT} - 1 \right]$$

where A is the area of the diode, D_n,D_p is the diffusion coefficient of electrons and holes and L_n,L_p is the diffusion length of electrons and holes.

Which we usually write as

$$I_D = I_S(e^{\frac{V_D}{V_T}} - 1), \text{ where } V_T = kT/q$$

5.6.3 Forward voltage temperature dependence

We can rearrange I_D equation to get

$$V_D = V_T \ln \left(\frac{I_D}{I_S} \right)$$

and at first glance, it appears like V_D has a positive temperature coefficient. That is, however, wrong.

First rewrite

$$V_D = V_T \ln I_D - V_T \ln I_S$$

$$\ln I_S = 2 \ln n_i + \ln Aq \left(\frac{D_n}{L_n N_A} + \frac{D_p}{L_p N_D} \right)$$

Assume that diffusion coefficient [‡], and diffusion lengths are independent of temperature.

That leaves n_i that varies with temperature.

$$n_i = \sqrt{B_c B_v} T^{3/2} e^{\frac{-E_g}{2kT}}$$

where

$$B_c = 2 \left[\frac{2\pi k m_n^*}{h^2} \right]^{3/2} \quad B_v = 2 \left[\frac{2\pi k m_p^*}{h^2} \right]^{3/2}$$

[‡] From the Einstein relation $D = \mu kT$ it does appear that the diffusion coefficient increases with temperature, however, the mobility decreases with temperature. I'm unsure of whether the mobility decreases with the same rate though.

$$2 \ln n_i = 2 \ln \sqrt{B_c B_v} + 3 \ln T - \frac{V_G}{V_T}$$

with $V_G = E_G/q$ and inserting back into equation for V_D

$$V_D = \frac{kT}{q}(\ell - 3 \ln T) + V_G$$

Where ℓ is temperature independent, and given by

$$\ell = \ln I_D - \ln \left(Aq \frac{D_n}{L_n N_A} + \frac{D_p}{L_p N_D} \right) - 2 \ln \sqrt{B_c B_v}$$

From equations above we can see that at 0 K, we expect the diode voltage to be equal to the bandgap of silicon. Diodes don't work at 0 K though.

Although it's not trivial to see that the diode voltage has a negative temperature coefficient, if you do compute it as in vd.py, then you'll see it decreases.

The slope of the diode voltage can be seen to depend on the area, the current, doping, diffusion constant, diffusion length and the effective masses.

Figure 3 shows the V_D and the deviation of V_D from a straight line. The non-linear component of V_D is only a few mV. If we could combine V_D with a voltage that increased with temperature, then we could get a stable voltage across temperature to within a few mV.

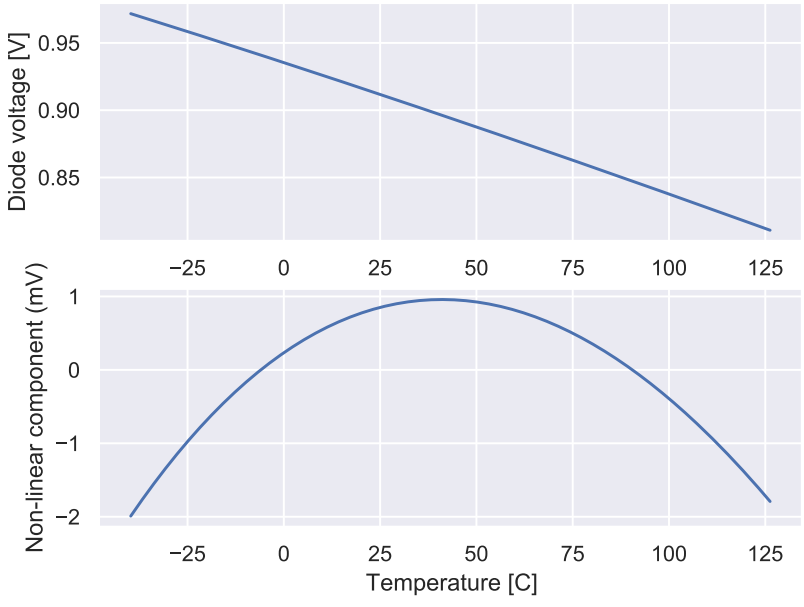


Figure 3: Diode forward voltage as a function of temperature

5.6.4 Current proportional to temperature

Assume we have a circuit like Figure 4.

Here we have two diodes, biased at different current densities. The voltage on the left diode V_{D1} is equal to the sum of the voltage on the right diode V_{D2} and voltage across the resistor R_1. The current in the two diodes are the same due to the current mirror. A such, we have that

$$I_S e^{\frac{qV_{D1}}{kT}} = N I_S e^{\frac{qV_{D2}}{kT}}$$

Taking logarithm of both sides, and rearranging, we see that

$$V_{D1} - V_{D2} = \frac{kT}{q} \ln N$$

Or that the difference between two diode voltages biased at different current densities is proportional to absolute temperature.

In the circuit above, this ΔV_D is across the resistor R_1, as such, the $I_D = \Delta V_D / R_1$. We have a current that is proportional to temperature.

If we copied the current, and sent it into a series combination of a resistor R_2 and a diode, we could scale the R_2 value to give us the exactly right slope to compensate for the negative slope of the V_D voltage.

The voltage across the resistor and diode would be constant over temperature, with the small exception of the non-linear component of V_D.

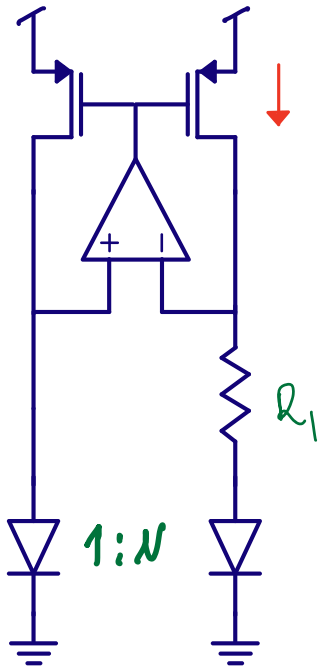


Figure 4: Circuit to generate a current proportional to kT

5.7 Equations aren't real

> Nature does not care about equations. It just is.

We know, at the fundamental level, nature appears to obey the mathematics on quantum mechanics, however, due to the complexity of nature, it's not possible today (which is not the same as impossible), to compute exactly how the current in a diode works. We can get close, by measuring a diode we know well, and hope that the next time we make the same diode, the behavior will be the same.

As such, I want to warn you about the "lies" or "simplifications" we tell you. Take the diode equation above, some parts, like the intrinsic carrier concentration n_i has roots directly from quantum mechanics, with few simplifications, which means it's likely solid truth, at least for a single unit cell.

But there is no reason nature should make all unit cells the same, and infact, we know they are not the same, we put in dopants. As we scale down to a few nano-meter transistors the simplification that "all unit cells of silicon are the same, and extend to infinity" is

no longer true, and must be taken into account in how we describe reality.

Other parts, like the exact value of the bandgap E_g, the diffusion constant D_p or diffusion length L_p are macroscopic phenomena, we can't expect them to be 100 true. The values would be based on measurement, but not always exact, and maybe, if you rotate your diode, they would be different.

You should realize that the consequence of our imperfection is that the equations in electronics should always be taken with a grain of salt.

Nature does not care about your equations. Nature will easily have the superposition of trillions of electrons, and they don't have to agree with your equations.

But most of the time, the behavior is similar.

References

[1] T. C. Carusone, D. Johns, and K. Martin, *Analog integrated circuit design*. Wiley, 2011 [Online]. Available: https://books.google.no/books?id=1OIJZzLvVhcC

[2] Berkeley, "Berkeley short-channel IGFET model." [Online]. Available: http://bsim.berkeley.edu/models/bsim4/

6 Mosfets

Spice 7

8 ESD and IC

9 References and bias

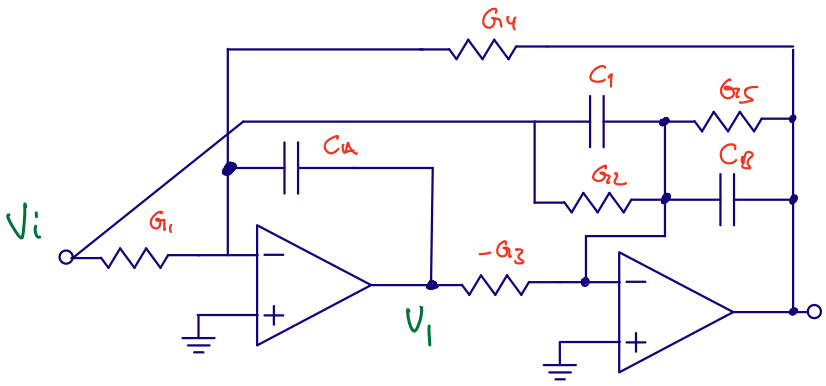
Analog frontend and filter 10

Switched capacitor circuits 11

Keywords: SC DAC, SC FUND, DT, Alias, Subsample, Z Domain, FIR, IIR, SC MDAC, SC INT, Switch, Non-Overlap, VBE SC, Nyquist

11.1 Active-RC

A general purpose Active-RC bi-quadratic (two-quadratic equations) filter is shown below



If you want to spend a bit of time, then try and calculate the transfer function below.

$$H(s) = \frac{\left[\frac{C_1}{C_B}s^2 + \frac{G_2}{C_B}s + (\frac{G_1 G_3}{C_A C_B})\right]}{\left[s^2 + \frac{G_5}{C_B}s + \frac{G_3 G_4}{C_A C_B}\right]}$$

Active resistor capacitor filters are made with OTAs (high output impedance) or OPAMP (low output impedance). Active amplifiers will consume current, and in Active-RC the amplifiers are always on, so there is no opportunity to reduce the current consumption by duty-cycling (turning on and off).

Both resistors and capacitors vary on an integrated circuit, and the 3-sigma variation can easily be 20 %.

The pole or zero frequency of an Active-RC filter is proportional to the inverse of the product between R and C

$$\omega_{p|z} \propto \frac{G}{C} = \frac{1}{RC}$$

As a result, the total variation of the pole or zero frequency is can have a 3-sigma value of

$$\sigma_{RC} = \sqrt{\sigma_R^2 + \sigma_C^2} = \sqrt{0.02^2 + 0.02^2} = 0.028 = 28\ \%$$

On an IC we sometimes need to calibrate the R or C in production to get an accurate RC time constant.

We cannot physically change an IC, every single one of the 100 million copies of an IC is from the same Mask set. That's why ICs are cheap. To make the Mask set is incredibility expensive (think 5 million dollars), but a copy made from the Mask set can cost one dollar or less. To calibrate we need additional circuits.

Imagine we need a resistor of 1 kOhm. We could create that by parallel connection of larger resistors, or series connection of smaller resistors. Since we know the maximum variation is 0.02, then we need to be able to calibrate away +- 20 Ohms. We could have a 980 kOhm resistor, and then add ten 4 Ohm resistors in series that we can short with a transistor switch.

But is a resolution of 4 Ohms accurate enough? What if we need a precision of 0.1%? Then we would need to tune the resistor within +-1 Ohm, so we might need 80 0.5 Ohm resistors.

But how large is the on-resistance of the transistor switch? Would that also affect our precision?

But is the calibration step linear with addition of the transistors? If we have a non-linear calibration step, then we cannot use gradient decent calibration algorithms, nor can we use binary search.

Analog designers need to deal with an almost infinite series of "But".

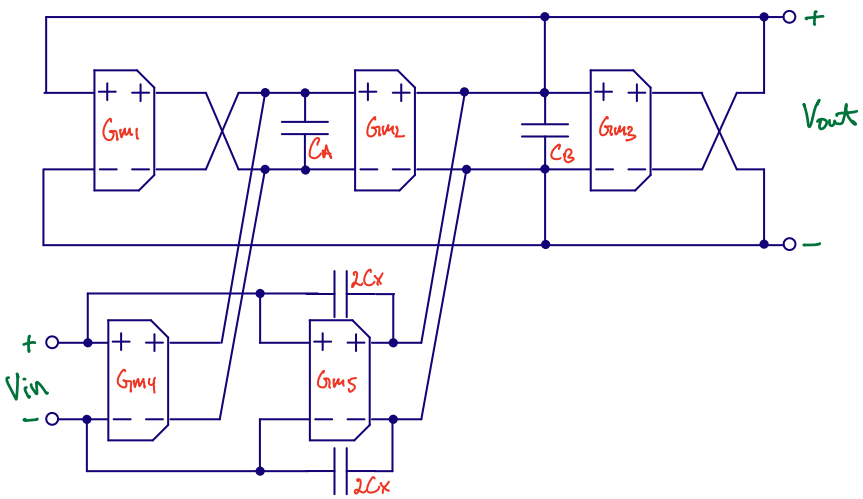
The experienced designer will know when to stop, when is the "But what if" not a problem anymore.

The most common error in analog integrated circuit design is a "I did not imagine that my circuit could fail in this manner" type of problem. Or, not following the line of "But" 's far enough.

But if we follow all the "But" 's we will never tapeout!

Active-RC filters are great for linearity, but if we need accurate time constant, there are better alternatives.

11.2 Gm-C



$$H(s) = \frac{\left[s^2 \frac{C_X}{C_X+C_B} + s\frac{G_{m5}}{C_X+C_B} + \frac{G_{m2}G_{m4}}{C_A(C_X+C_B)}\right]}{\left[s^2 + s\frac{G_{m2}}{C_X+C_B} + \frac{G_{m1}G_{m2}}{C_A(C_X+C_B)}\right]}$$

The pole and zero frequency of a Gm-C filter is

$$\omega_{p|z} \propto \frac{G_m}{C}$$

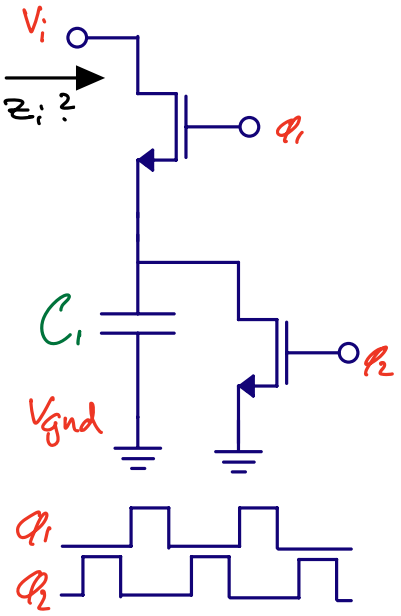
The transconductance accuracy depends on the circuit, and the bias circuit, so we can't give a general, applies for all circuits, sigma number. Capacitors do have 3-sigma 20 % variation, usually.

Same as Active-RC, Gm-C need calibration to get accurate pole or zero frequency.

11.3 Switched capacitor

The first time you encounter Switched Capacitor (SC) circuits, they do require some brain training. So let's start simple.

Consider the circuit below. Assume that the two transistors are ideal (no-charge injection, no resistance).



For SC circuits, we need to consider the charge on the capacitors, and how they change with time.

The charge on the capacitor at the end * of phase 2 is

$$Q_{\phi 2\$} = C_1 V_{GND} = 0$$

while at the end of phase 1

$$Q_{\phi 1\$} = C_1 V_I$$

The impedance, from Ohm's law is

$$Z_I = (V_I - V_{GND})/I_I$$

And from SI units units we can see current is

$$I_I = \frac{Q}{dt} = Q f_\phi$$

Charge cannot disappear, charge is conserved. As such, the charge going out from the input must be equal to the difference of charge at the end of phase 1 and phase 2.

$$Z_I = \frac{V_I - V_{GND}}{(Q_{\phi 1\$} - Q_{\phi 2\$})\, f_\phi}$$

* I use the $ to mark the end of the period. It comes from Regular Expressions.

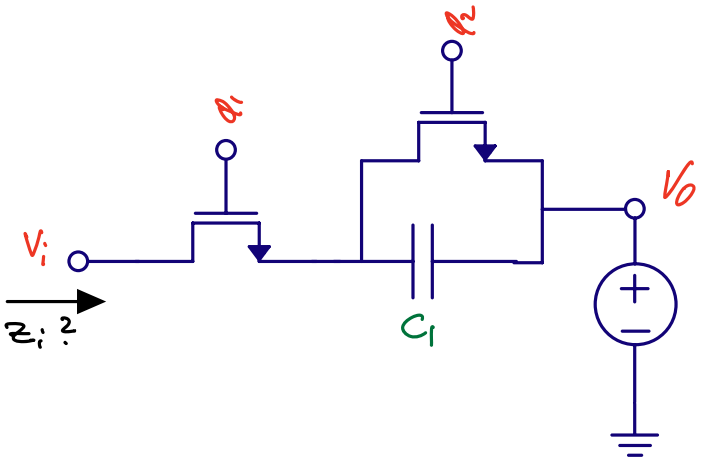
Inserting for the charges, we can see that the impedance is

$$Z_I = \frac{V_I}{(V_I C - 0)\, f_\phi} = \frac{1}{C_1 f_\phi}$$

A common confusion with SC circuits is to confuse the impedance of a capacitor $Z = 1/sC$ with the impedance of a SC circuit $Z = 1/fC$. The impedance of a capacitor is complex (varies with frequency and time), while the SC circuit impedance is real (a resistance).

The main difference between the two is that the impedance of a capacitor is continuous in time, while the SC circuit is a discrete time circuit, and has a discrete time impedance.

The circuit below is drawn slightly differently, but the same equation applies.



If we compute the impedance.

$$Z_I = \frac{V_I - V_O}{\left(Q_{\phi 1\$} - Q_{\phi 2\$}\right) f_\phi}$$

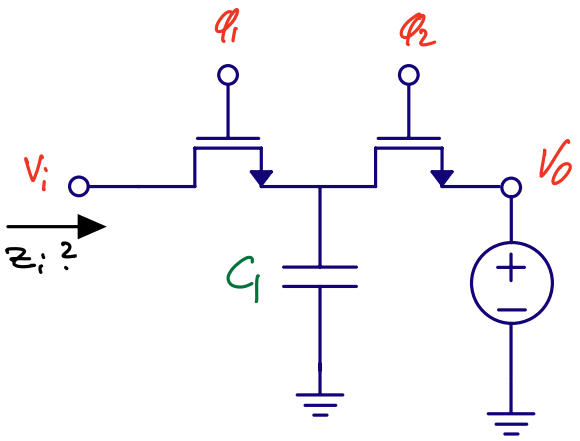
$$Q_{\phi 1\$} = C_1(V_I - V_O)$$

$$Q_{\phi 2\$} = 0$$

$$Z_I = \frac{V_I - V_O}{(C_1(V_I - V_O))\, f_\phi} = \frac{1}{C_1 f_\phi}$$

Which should not be surprising, as all I've done is to rotate the circuit and call $V_{GND} = V_0$.

Let's try the circuit below.



$$Z_I = \frac{V_I - V_O}{(Q_{\phi 1\$} - Q_{\phi 2\$})\, f_\phi}$$

$$Q_{\phi 1\$} = C_1 V_I)$$

$$Q_{\phi 2\$} = C_1 V_O$$

Inserted into the impedance we get the same result.

$$Z_I = \frac{V_I - V_O}{(C_1 V_I - C_1 V_O))\, f_\phi} = \frac{1}{C_1 f_\phi}$$

The first time I saw the circuit above it was not obvious to me that the impedance still was $Z = 1/Cf$. It's one of the cases where mathematics is a useful tool. I could follow a set of rules (charge conservation), and as long as I did the mathematics right, then from the equations, I could see how it worked.

11.3.1 An example SC circuit

An example use of an SC circuit is

A pipelined 5-Msample/s 9-bit analog-to-digital converter

Shown in the figure below. You should think of the switched capacitor circuit as similar to a an amplifier with constant gain. We can use two resistors and an opamp to create a gain. Imagine we create a circuit without the switches, and with a resistor of R from input to virtual ground, and $4R$ in the feedback. Our Active-R would have a gain of $A = 4$.

The switches disconnect the OTA and capacitors for half the time, but for the other half, at least for the latter parts of ϕ_2 the gain is four.

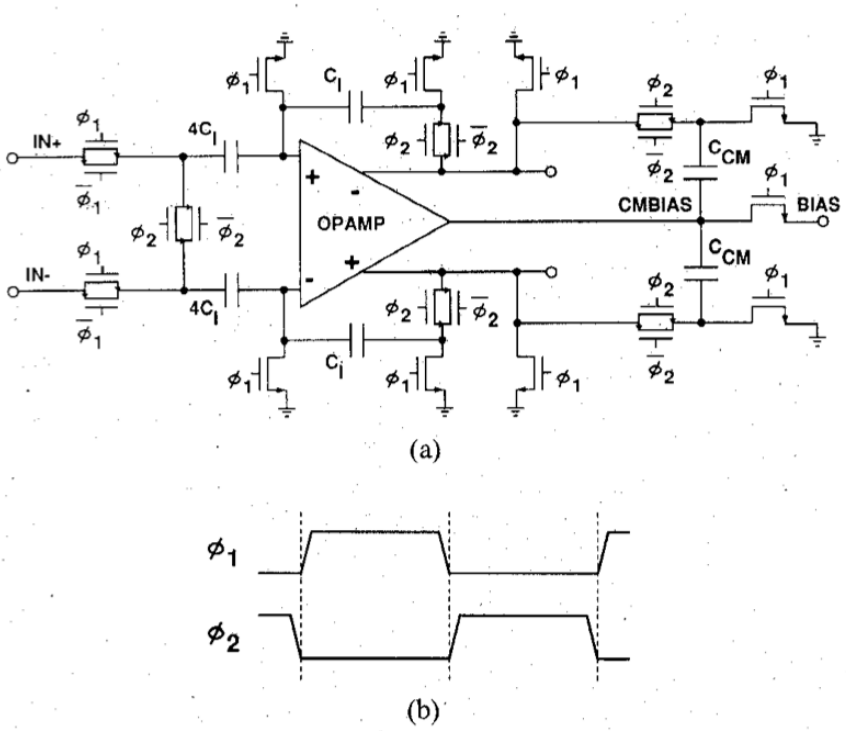


Fig. 6. (a) Schematic of S/H amplifier. (b) Timing diagram of a two-phase nonoverlapping clock.

The output is only correct for a finite, but periodic, time interval. The circuit is discrete time. As long as all circuits afterwards also have a discrete-time input, then it's fine. An ADC can sample the output from the amplifier at the right time, and never notice that the output is shorted to a DC voltage in ϕ_1

We charge the capacitor $4C$ to the differential input voltage in ϕ_1

$$Q_1 = 4CV_{in}$$

Then we turn off ϕ_1, which opens all switches. The charge on $4C$ will still be Q_1 (except for higher order effects like charge injection from switches).

After a short time (non-overlap), we turn on ϕ_2, closing some of the switches. The OTA will start to force its two inputs to be the same voltage, and we short the left side of $4C$. After some time we would have the same voltage on the left side of $4C$ for the two capacitors, and another voltage on the right side of the $4C$ capacitors. The two capacitors must now have the same charge, so the difference in charge, or differential charge must be zero.

Physics tell us that charge is conserved, so our differential charge Q_1 cannot vanish into thin air. The difference in electrons that made Q_1 must be somewhere in our circuit.

Assume the designer of the circuit has done a proper job, then the Q_1 charge will be found on the feedback capacitors.

We now have a Q_1 charge on smaller capacitors, so the differential output voltage must be

$$Q_1 = 4CV_{in} = Q_2 = CV_{out}$$

The gain is

$$A = \frac{V_{out}}{V_{in}} = 4$$

Why would we go to all this trouble to get a gain of 4?

In general, we can sum up with the following equation.

$$\omega_{p|z} \propto \frac{C_1}{C_2}$$

We can use these "switched capacitor resistors" to get pole or zero frequency or gain proportional to a the relative size of capacitors, which is a fantastic feature. Assume we make two identical capacitors in our layout. We won't know the absolute size of the capacitors on the integrated circuit, whether the C_1 is 100 fF or 80 fF, but we can be certain that if C_1 = 80 fF, then C_2 = 80 fF to a precision of around 0.1 %.

With switched capacitor amplifiers we can set an accurate gain, and we can set an accurate pole and zero frequency (as long as we have an accurate clock and a high DC gain OTA).

The switched capacitor circuits do have a drawback. They are discrete time circuits. As such, we must treat them with caution, and they will always need some analog filter before to avoid a phenomena we call aliasing.

11.4 Discrete-Time Signals

An random, Gaussian, continuous time, continuous value, signal has infinite information. The frequency can be anywhere from zero to infinity, the value have infinite levels, and the time division is infinitely small. We cannot store such a signal. We have to quantize.

If we quantize time to $T = 1$ ns, such that we only record the value of the signal every 1 ns, what happens to all the other information? The stuff that changes at 0.5 ns or 0.1 ns, or 1 ns.

We can always guess, but it helps to know, as in absolutely know, what happens. That's where mathematics come in. With mathematics we can prove things, and know we're correct.

11.4.1 The mathematics

Define

$$x_c$$

as a continuous time, continuous value signal

Define

$$\ell(t) = \begin{cases} 1 & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}$$

Define

$$x_{sn}(t) = \frac{x_c(nT)}{\tau}[\ell(t - nT) - \ell(t - nT - \tau)]$$

where $x_{sn}(t)$ is a function of the continuous time signal at the time interval nT.

Define

$$x_s(t) = \sum_{n=-\infty}^{\infty} x_{sn}(t)$$

where $x_s(t)$ is the sampled, continuous time, signal.

Think of a sampled version of an analog signal as an infinite sum of pulse trains where the area under the pulse train is equal to the analog signal.

Why do this?

With a exact definition of a sampled signal in the time-domain it's sometimes possible to find the Laplace transform, and see how the frequency spectrum looks.

If

$$x_s(t) = \sum_{n=-\infty}^{\infty} x_{sn}(t)$$

Then

$$X_{sn}(s) = \frac{1}{\tau}\frac{1 - e^{-s\tau}}{s}x_c(nT)e^{-snT}$$

And

$$X_s(s) = \frac{1}{\tau}\frac{1 - e^{-s\tau}}{s}\sum_{n=-\infty}^{\infty} x_c(nT)e^{-snT}$$

Thus

$$\lim_{\tau \to 0} \rightarrow X_s(s) = \sum_{n=-\infty}^{\infty} x_c(nT) e^{-snT}$$

Or

$$X_s(j\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c \left(j\omega - \frac{jk2\pi}{T} \right)$$

The spectrum of a sampled signal is an infinite sum of frequency shifted spectra

or equivalently

When you sample a signal, then there will be copies of the input spectrum at every

$$n f_s$$

However, if you do an FFT of a sampled signal, then all those infinite spectra will fold down between

$$0 \rightarrow f_{s1}/2$$

or

$$-f_{s1}/2 \rightarrow f_{s1}/2$$

for a complex FFT

11.4.2 Python discrete time example

If your signal processing skills are a bit thin, now might be a good time to read up on FFT, Laplace transform and But what is the Fourier Transform?

In python we can create a demo and see what happens when we "sample" an "continuous time" signal. Hopefully it's obvious that it's impossible to emulate a "continuous time" signal on a digital computer. After all, it's digital (ones and zeros), and it has a clock!

We can, however, emulate to any precision we want.

The code below has four main sections. First is the time vector. I use Numpy, which has a bunch of useful features for creating ranges, and arrays.

Secondly, I create continuous time signal. The time vector can be used in numpy functions, like `np.sin()`, and I combine three sinusoid plus some noise. The sampling vector is a repeating pattern of 11001100, so our sample rate should be 1/2'th of the input sample rate. FFT's can be unwieldy beasts. I like to use coherent

sampling, however, with multiple signals and samplerates I did not bother to figure out whether it was possible.

The alternative to coherent sampling is to apply a window function before the FFT, that's the reason for the Hanning window below.

dt.py

```
#- Create a time vector
N = 2**13
t = np.linspace(0,N,N)

#- Create the "continuous time" signal with multiple
#- "sinusoidal signals and some noise
f1 = 233/N
fd = 1/N*119
x_s = np.sin(2*np.pi*f1*t) + 1/1024*np.random.randn(N) + \
    0.5*np.sin(2*np.pi*(f1-fd)*t) + 0.5*np.sin(2*np.pi*(f1+fd)*t)

#- Create the sampling vector, and the sampled signal
t_s_unit = [1,1,0,0,0,0,0,0]
t_s = np.tile(t_s_unit,int(N/len(t_s_unit)))
x_sn = x_s*t_s

#- Convert to frequency domain with a hanning window to avoid FFT bin
#- energy spread
Hann = True
if(Hann):
    w = np.hanning(N+1)
else:
    w = np.ones(N+1)
X_s = np.fft.fftshift(np.fft.fft(np.multiply(w[0:N],x_s)))
X_sn = np.fft.fftshift(np.fft.fft(np.multiply(w[0:N],x_sn)))
```

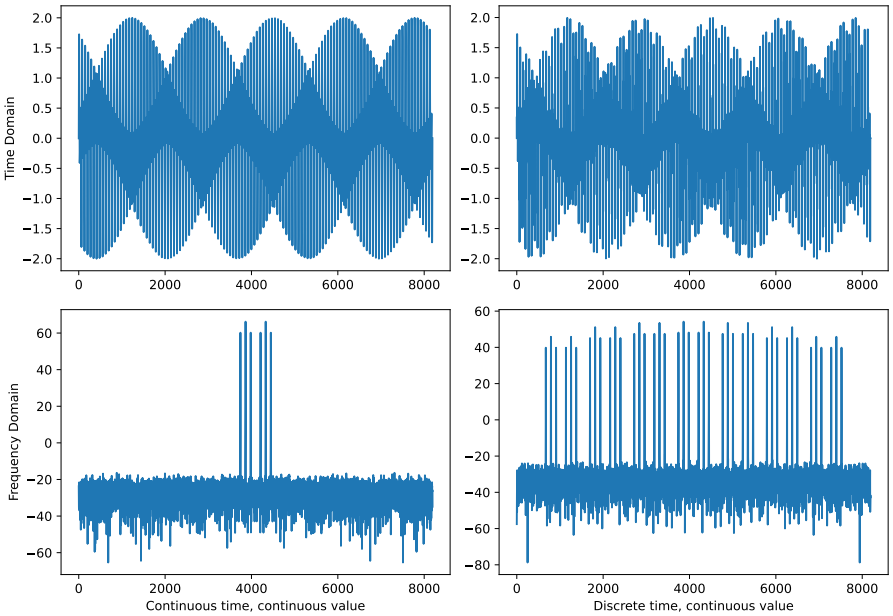
Try to play with the code, and see if you can understand what it does.

Below are the plots. On the left side is the "continuous value, continuous time" emulation, on the right side "discrete time, continuous value".

The top plots are the time domain, while the bottom plots is frequency domain.

The FFT is complex, so that's why there are six sinusoids bottom left. The "0 Hz" would be at x-axis index 4096 ($2^{13}/2$).

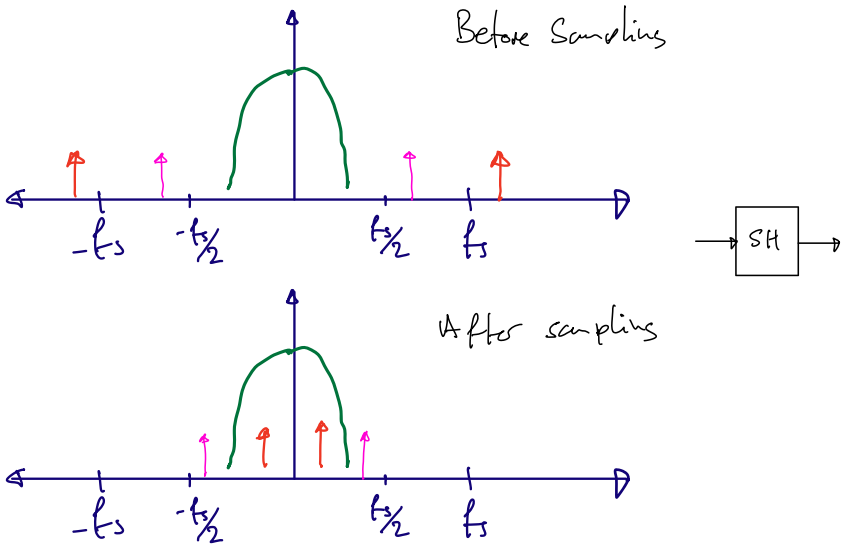
The spectral copies can be seen bottom right. How many spectral copies, and the distance between them will depend on the sample rate (length of `t_s_unit`). Try to play around with the code and see what happens.



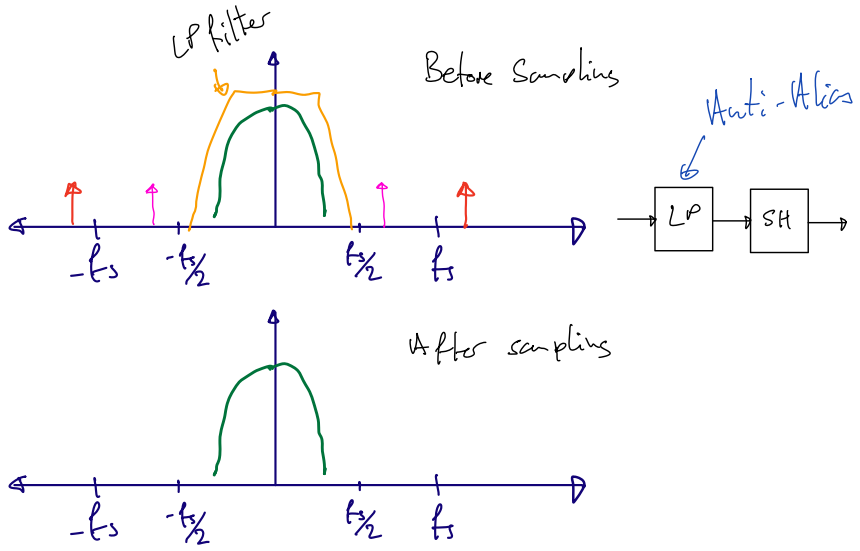
11.4.3 Aliasing, bandwidth and sample rate theory

I want you to internalize that the spectral copies are real. They are not some "mathematical construct" that we don't have to deal with.

They are what happens when we sample a signal into discrete time. Imagine a signal with a band of interest as shown below in Green. We sample at f_s. The pink and red unwanted signals do not disappear after sampling, even though they are above the Nyquist frequency ($f_s/2$). They fold around $f_s/2$, and in may appear in-band. That's why it's important to band limit analog signals before they are sampled.



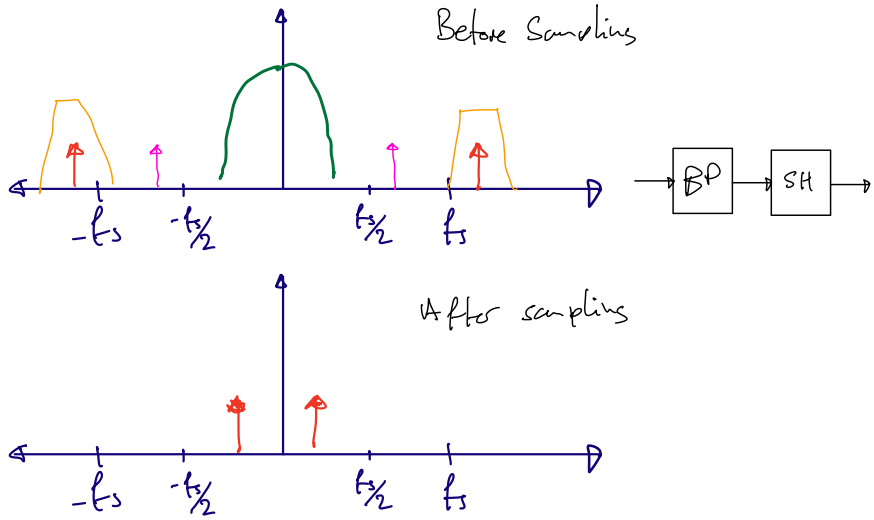
With an anti-alias filter (yellow) we ensure that the unwanted components are low enough before sampling. As a result, our wanted signal (green) is undisturbed.



Assume that we we're interested in the red signal. We could still use a sample rate of f_s. If we bandpass-filtered all but the red signal the red signal would fold on sampling, as shown in the figure below.

Remember that the Nyquist-Shannon states that a sufficient no-loss condition is to sample signals with a sample rate of twice the bandwidth of the signal.

Nyquist-Shannon has been extended for sparse signals, compressed sensing, and non-uniform sampling to demonstrate that it's sufficient for the average sample rate to be twice the bandwidth. One 2009 paper Blind Multiband Signal Reconstruction: Compressed Sensing for Analog Signal is a good place to start to delve into the latest on signal reconstruction.



11.4.4 Z-transform

Someone got the idea that writing

$$X_s(s) = \sum_{n=-\infty}^{\infty} x_c(nT)e^{-snT}$$

was cumbersome, and wanted to find something better.

$$X_s(z) = \sum_{n=-\infty}^{\infty} x_c[n]z^{-n}$$

For discrete time signal processing we use Z-transform

If you're unfamiliar with the Z-transform, read the book or search https://en.wikipedia.org/wiki/Z-transform

The nice thing with the Z-transform is that the exponent of the z tell's you how much delayed the sample $x_c[n]$ is. A block that delays a signal by 1 sample could be described as $x_c[n]z^{-1}$, and an accumulator

$$y[n] = y[n-1] + x[n]$$

in the Z domain would be

$$Y(z) = z^{-1}Y(z) + X(z)$$

With a Z-domain transfer function of

$$\frac{Y(z)}{X(z)} = \frac{1}{1 - z^{-1}}$$

11.4.5 Pole-Zero plots

If you're not comfortable with pole/zero plots, have a look at

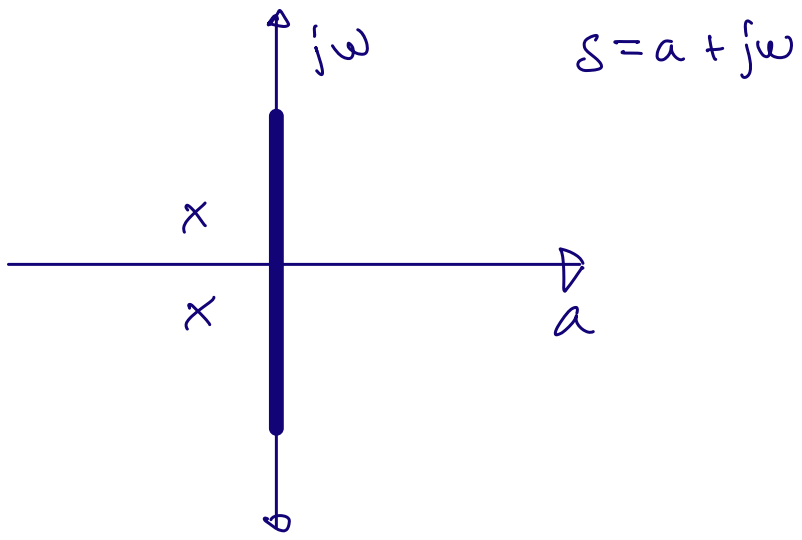
What does the Laplace Transform really tell us

Think about the pole/zero plot as a surface your looking down onto. At $a = 0$ we have the steady state fourier transform. The "x" shows the complex frequency where the fourier transform goes to infinity.

Any real circuit will have complex conjugate, or real, poles/zeros. A combination of two real circuits where one path is shifted 90 degrees in phase can have non-conjugate complex poles/zeros.

If the "x" is $a < 0$, then any perturbation will eventually die out. If the "x" is on the $a = 0$ line, then we have a oscillator that will ring forever. If the "x" is $a > 0$ then the oscillation amplitude will grow without bounds, although, only in Matlab. In any physical circuit an oscillation cannot grow without bounds forever.

Growing without bounds is the same as "being unstable".



11.4.6 Z-domain

Spectra repeat every

$$2\pi$$

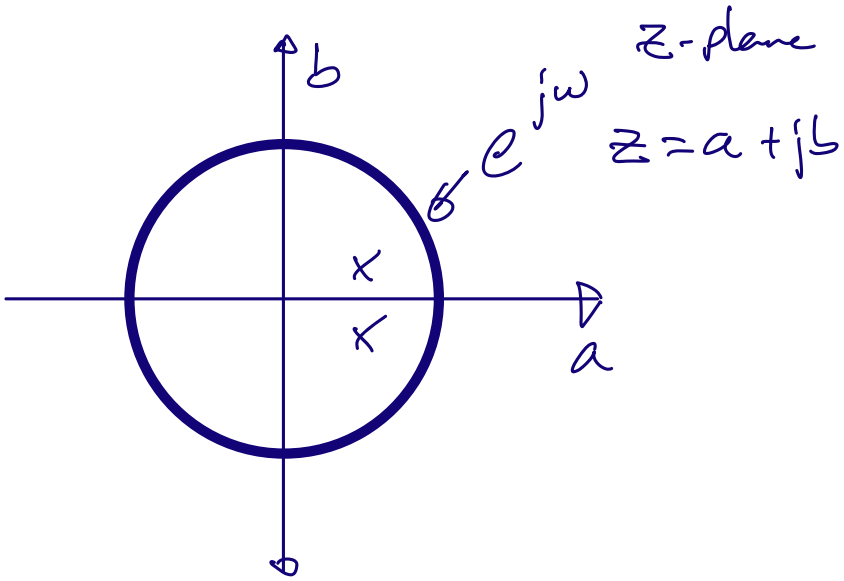
As such, it does not make sense to talk about a plane with a a and a $j\omega$. Rather we use the complex number $z = a + jb$.

As long as the poles ("x") are within the unit circle, oscillations will die out. If the poles are on the unit-circle, then we have an oscillator. Outside the unit circle the oscillation will grow without bounds, or in other words, be unstable.

We can translate between Laplace-domain and Z-domain with the Bi-linear transform

$$s = \frac{z-1}{z+1}$$

Warning: First-order approximation https://en.wikipedia.org/wiki/Bilinear_transform



11.4.7 First order filter

Assume a first order filter given by the discrete time equation.

$$y[n+1] = bx[n] + ay[n] \Rightarrow Yz = bX + aY$$

The "n" index and the "z" exponent can be chosen freely, which sometimes can help the algebra.

$$y[n] = bx[n-1] + ay[n-1] \Rightarrow Y = bXz^{-1} + aYz^{-1}$$

The transfer function can be computed as

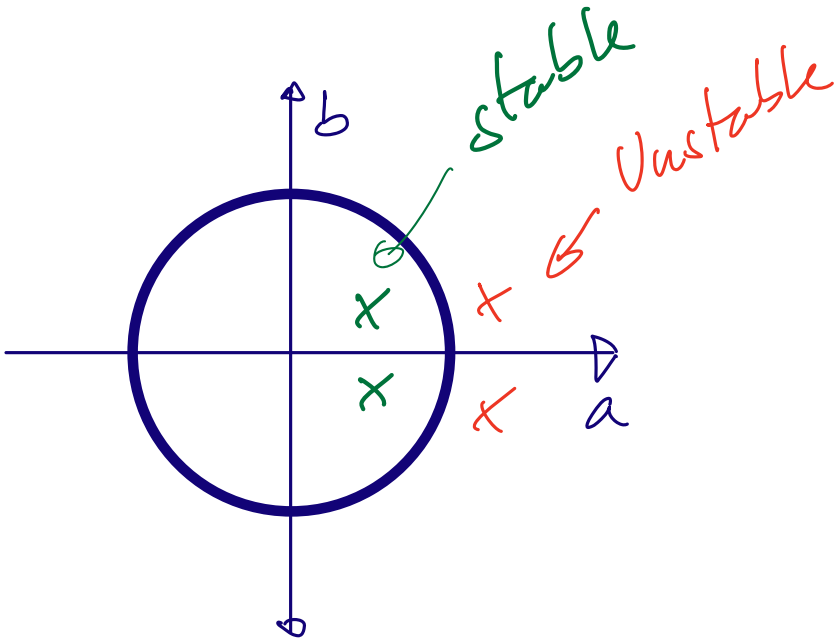
$$H(z) = \frac{b}{z - a}$$

From the discrete time equation we can see that the impulse will never die out. We're adding the previous output to the current input. That means the circuit has infinite memory. Accordingly, filters of this type are known as. Infinite-impulse response (IIR)

$$h[n] = \begin{cases} k & \text{if } n < 1 \\ a^{n-1}b + a^n k & \text{if } n \geq 1 \end{cases}$$

Head's up: Fig 13.12 in AIC is wrong

From the impulse response it can be seen that if $a > 1$, then the filter is unstable. Same if $b > 1$. As long as $|a + jb| < 1$ the filter should be stable.

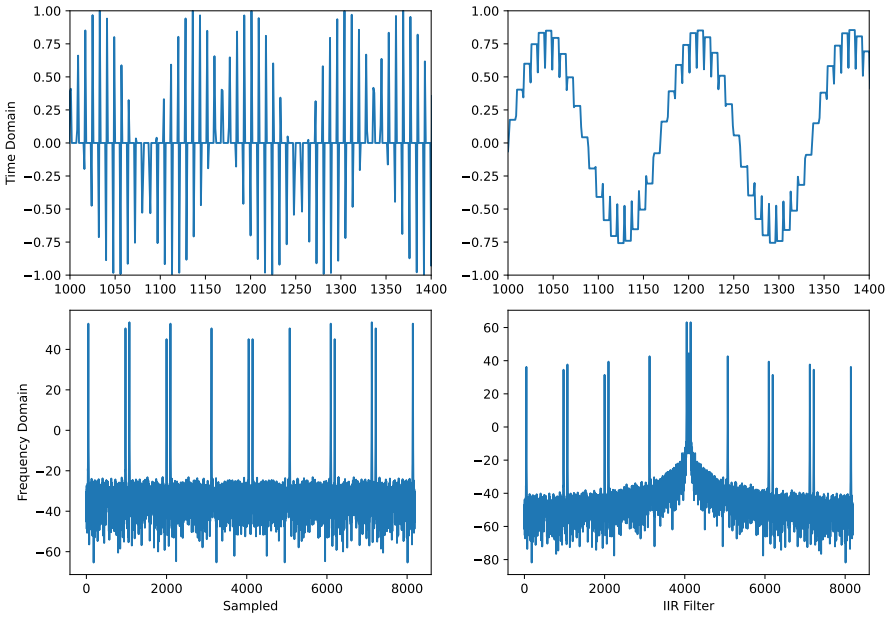


The first order filter can be implemented in python, and it's really not hard. See below. The $x_s n$ vector is from the previous python example.

There are smarter, and faster ways to do IIR filters (and FIR) in python, see scipy.signal.iirfilter

From the plot below we can see the sampled time domain and spectra on the left, and the filtered time domain and spectra on the right.

iir.py




```
#- IIR filter
b = 0.3
a = 0.25
z = a + 1j*b
z_abs = np.abs(z)
print("|z| = " + str(z_abs))
y = np.zeros(N)
y[0] = a
for i in range(1,N):
    y[i] = b*x_sn[i-1] + y[i-1]
```

The IIR filter we implemented above is a low-pass filter, and the filter partially rejects the copied spectra, as expected.

11.4.8 Finite-impulse response(FIR)

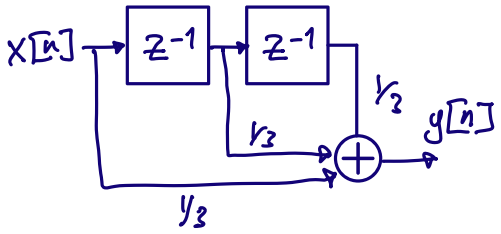
FIR filters are unconditionally stable, since the impulse response will always die out. FIR filters are a linear sum of delayed inputs.

In my humble opinion, there is nothing wrong with an IIR. Yes, the could become unstable, however, they can be designed safely. I'm not sure there is a theological feud on IIR vs FIR, I suspect there could be. Talk to someone that knows digital filters better than me.

But be wary of rules like "IIR are always better than FIR" or visa versa. Especially if statements are written in books. Remember that the book was probably written a decade ago, and based on papers two decades old, which were based on three decades old state of

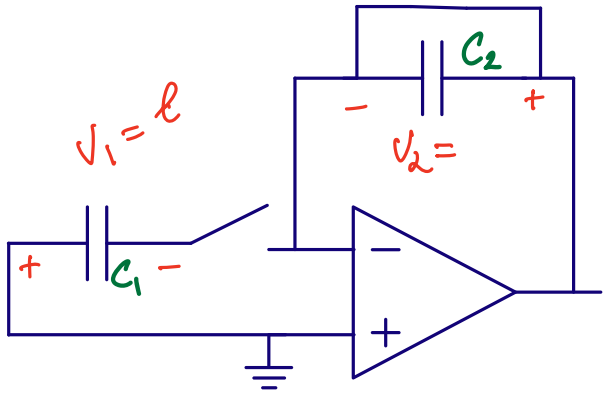
the art. Our abilities to use computers for design has improved a bit the last three decades.

$$H(z) = \frac{1}{3}\sum_{i=0}^{2} z^{-1}$$



11.5 Switched-Capacitor

Below is an example of a switched-capacitor circuit during phase 1. Think of the two phases as two different configurations of a circuit, each with a specific purpose.



This is the SC circuit during the sampling phase. Imagine that we somehow have stored a voltage $V_1 = \ell$ on capacitor C_1 (the switches for that sampling or storing are not shown). The charge on C_1 is

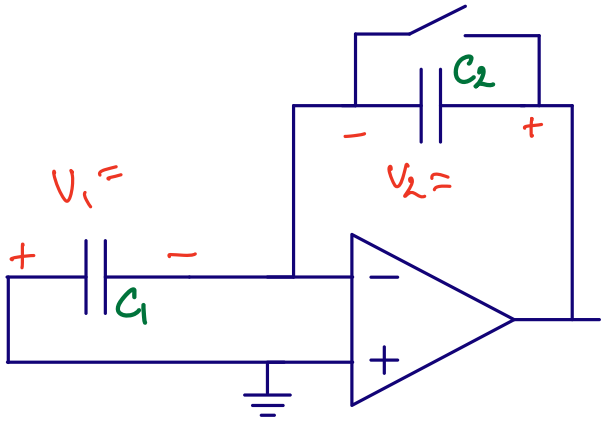
$$Q_{1\phi_1\$} = C_1 V_1$$

The C_2 capacitor is shorted, as such, $V_2 = 0$, which must mean that the charge on C_2 given by

$$Q_{2\phi_1\$} = 0$$

The voltage at the negative input of the OTA must be 0 V, as the positive input is 0 V, and we assume the circuit has settled all transients.

Imagine we (very carefully) open the circuit around C_2 and close the circuit from the negative side of C_1 to the OTA negative input, as shown below.



It's the OTA that ensures that the negative input is the same as the positive input, but the OTA cannot be infinitely fast. At the same time, the voltage across C_1 cannot change instantaneously. Neither can the voltage across C_2. As such, the voltage at the negative input must immediately go to $-V_1$ (ignoring any parasitic capacitance at the negative input).

The OTA does not like it's inputs to be different, so it will start to charge C_2 to increase the voltage at the negative input to the OTA. When the negative input reaches 0 V the OTA is happy again. At that point the charge on C_1 is

$$Q_{1\phi_2\$} = 0$$

A key point is, that even the voltages now have changed, there is zero volt across C_1, and thus there cannot be any charge across C_1 the charge that was there cannot have disappeared. The negative input of the OTA is a high impedance node, and cannot supply charge. The charge must have gone somewhere, but where?

In process of changing the voltage at the negative input of the OTA we've changed the voltage across C_2. The voltage change must exactly match the charge that was across C_1, as such

$$Q_{2\phi_2\$} = Q_{1\phi_1\$} = C_1V_1 = C_2V_2$$

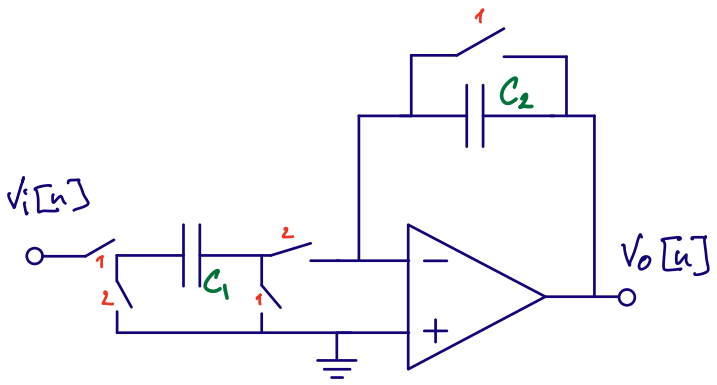
thus

$$\frac{V_2}{V_1} = \frac{C_1}{C_2}$$

11.5.1 Switched capacitor gain circuit

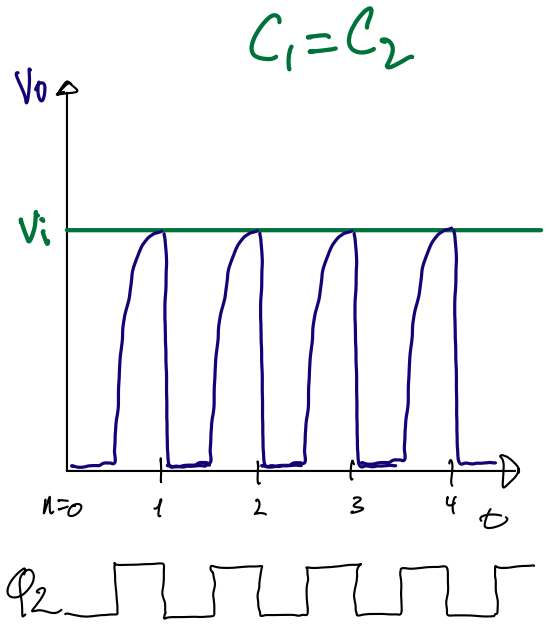
Redrawing the previous circuit, and adding a few more switches we can create a switched capacitor gain circuit.

There is now a switch to sample the input voltage across C_1 during phase 1 and reset C_2. During phase 2 we configure the circuit to leverage the OTA to do the charge transfer from C_1 to C_2.



The discrete time output from the circuit will be as shown below. It's only at the end of the second phase that the output signal is valid. As a result, it's common to use the sampling phase of the next circuit close to the end of phase 2.

For charge to be conserved the clocks for the switch phases must never be high at the same time.



The discrete time, Z-domain and transfer function is shown below. The transfer function tells us that the circuit is equivalent to a gain, and a delay of one clock cycle. The cool thing about switch capacitor circuits is that the precision of the gain is set by the relative size between two capacitors. In most technologies that relative sizing can be better than 0.1 %.

Gain circuits like the one above find use in most Pipelined ADCs, and are common, with some modifications, in Sigma-Delta ADCs.

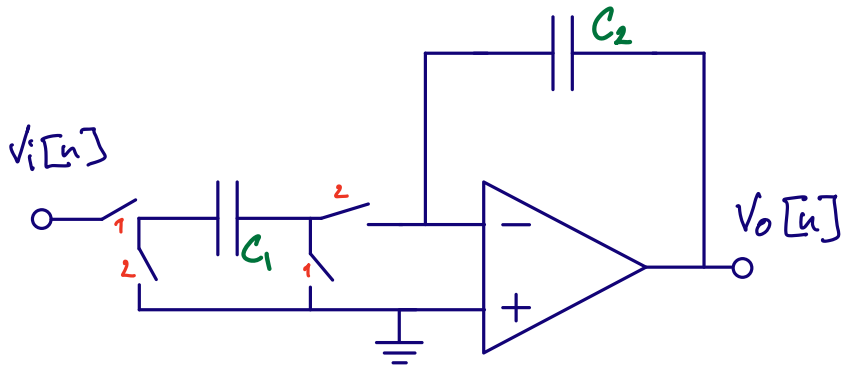
$$V_o[n+1] = \frac{C_1}{C_2} V_i[n]$$

$$V_o z = \frac{C_1}{C_2} V_i$$

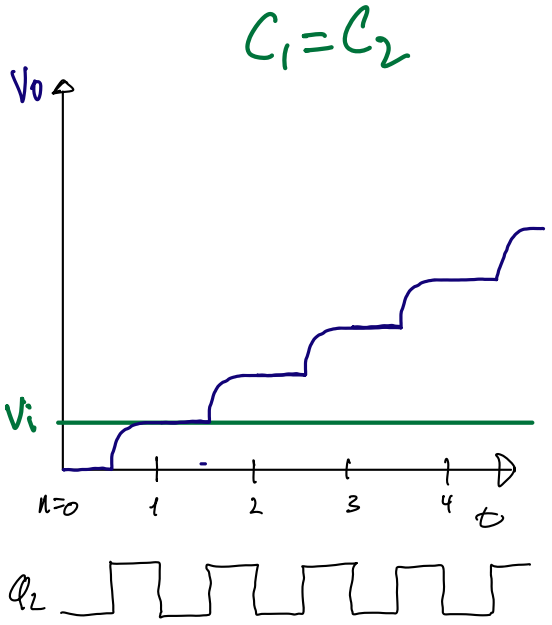
$$\frac{V_o}{V_i} = H(z) = \frac{C_1}{C_2} z^{-1}$$

11.5.2 Switched capacitor integrator

Removing one switch we can change the function of the switched capacitor gain circuit. If we don't reset C_2 then we accumulate the input charge every cycle.



The output now will grow without bounds, so integrators are most often used in filter circuits, or sigma-delta ADCs where there is feedback to control the voltage swing at the output of the OTA.



Make sure you read and understand the equations below, it's good to realize that discrete time equations, Z-domain and transfer functions in the Z-domain are actually easy.

$$V_o[n] = V_o[n-1] + \frac{C_1}{C_2} V_i[n-1]$$

$$V_o - z^{-1}V_o = \frac{C_1}{C_2} z^{-1} V_i$$

Maybe one confusing thing is that multiple transfer functions can mean the same thing, as below.

$$H(z) = \frac{C_1}{C_2}\frac{z^{-1}}{1-z^{-1}} = \frac{C_1}{C_2}\frac{1}{z-1}$$

11.5.3 Noise

Capacitors don't make noise, but switched-capacitor circuits do have noise. The noise comes from the thermal, flicker, burst noise in the switches and OTA's. Both phases of the switched capacitor circuit contribute noise. As such, the output noise of a SC circuit is usually

$$V_n^2 > \frac{2kT}{C}$$

I find that sometimes it's useful with a repeat of mathematics, and since we're talking about noise.

The mean, or average of a signal is defined as

Mean

$$\overline{x(t)} = \lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{+T/2} x(t)dt$$

Define

Mean Square

$$\overline{x^2(t)} = \lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{+T/2} x^2(t)dt$$

How much a signal varies can be estimated from the Variance

$$\sigma^2 = \overline{x^2(t)} - \overline{x(t)}^2$$

where

$$\sigma$$

is the standard deviation. If mean is removed, or is zero, then

$$\sigma^2 = \overline{x^2(t)}$$

Assume two random processes,

$$x_1(t)$$

and

$$x_2(t)$$

with mean of zero (or removed).

$$x_{tot}(t) = x_1(t) + x_2(t)$$

$$x_{tot}^2(t) = x_1^2(t) + x_2^2(t) + 2x_1(t)x_2(t)$$

Variance (assuming mean of zero)

$$\sigma_{tot}^2 = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{+T/2} x_{tot}^2(t)dt$$

$$\sigma_{tot}^2 = \sigma_1^2 + \sigma_2^2 + \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{+T/2} 2x_1(t)x_2(t)dt$$

Assuming uncorrelated processes (covariance is zero), then

$$\sigma_{tot}^2 = \sigma_1^2 + \sigma_2^2$$

In other words, if two noises are uncorrelated, then we can sum the variances. If the noise sources are correlated, for example, noise comes from the same transistor, but takes two different paths through the circuit, then we cannot sum the variances. We must also add the co-variance.

11.5.4 Sub-circuits for SC-circuits

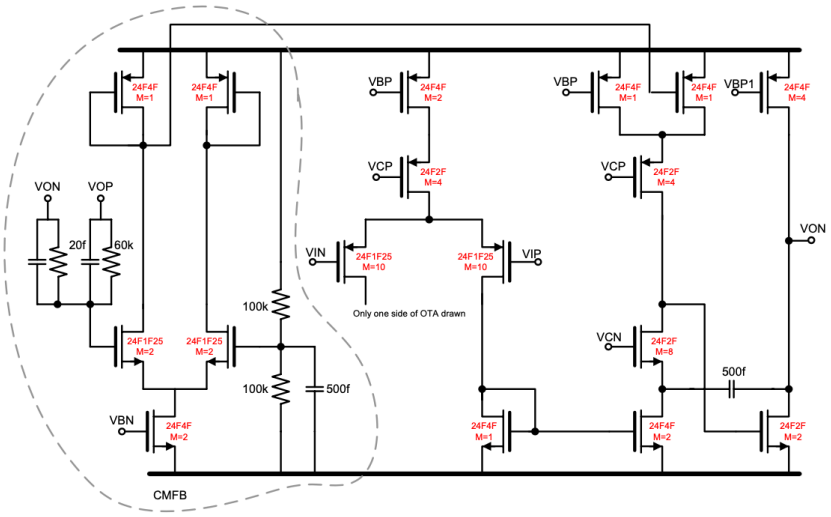
Switched-capacitor circuits are so common that it's good to delve a bit deeper, and understand the variants of the components that make up SC circuits.

11.5.4.1 OTA

At the heart of the SC circuit we usually find an OTA. Maybe a current mirror, folded cascode, recycling cascode, or my favorite: a fully differential current mirror OTA with cascoded, gain boosted, output stage using a parallel common mode feedback.

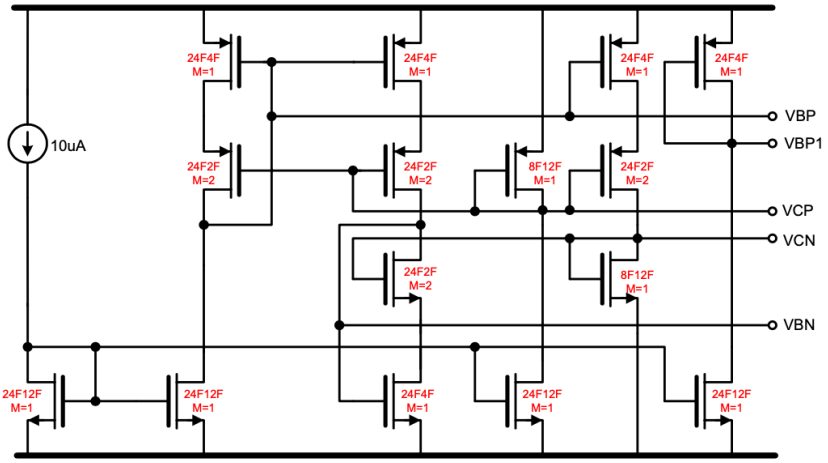
Not all SC circuits use OTAs, there are also comparator based SC circuits.

Below is a fully-differential two-stage OTA that will work with most SC circuits. The notation "24F1F25" means "the width is 24 F" and "length is 1.25 F", where "F" is the minimum gate length in that technology.



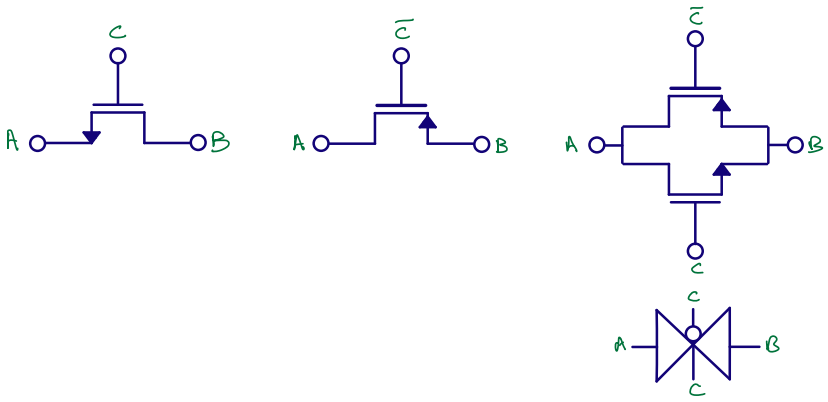
F = minimum transistor gate length. For example 24F4F => W = 24 x min gate, L= 4 x min gate

As bias circuit to make the voltages the below will work



11.5.4.2 Switches

If your gut reaction is "switches, that's easy", then you're very wrong. Switches can be incredibly complicated. All switches will be made of transistors, but usually we don't have enough headroom to use a single NMOS or PMOS. We may need a transmission gate



The challenge with transmission gates is that when the voltage at the input is in the middle between VDD and ground then both PMOS and NMOS, although they are on , they might not be that on. Especially in nano-scale CMOS with a 0.8 V supply and 0.5 V threshold voltage. The resistance mid-rail might be too large.

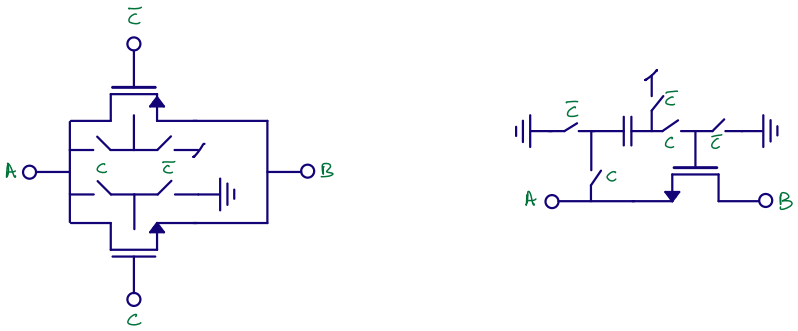
For switched-capacitor circuits we must settle the voltages to the required accuracy. In general

$$t > -\log(\text{error})\tau$$

For example, for a 10-bit ADC we need $t > -\log(1/1024)\tau = 6.9\tau$. This means we need to wait at least 6.9 time constants for the voltage to settle to 10-bit accuracy in the switched capacitor circuit.

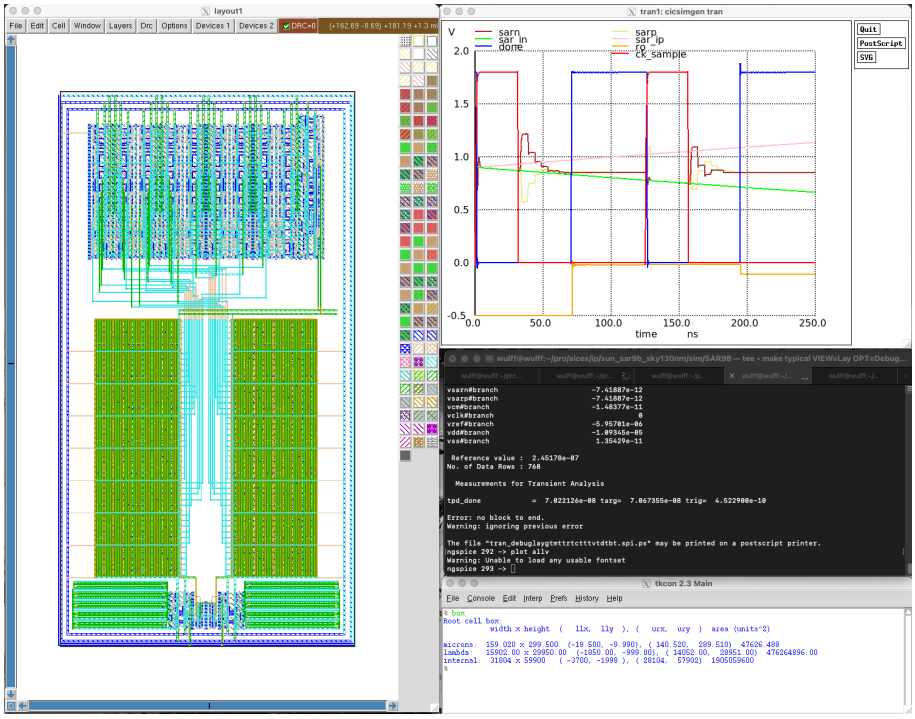
Assume the capacitors are large due to noise, then the switches must be low resistance for a reasonable time constant. Larger switches have smaller resistance, however, they also have more charge in the inversion layer, which leads to charge injection when the switches are turned of. Accordingly, larger switches are not always the solution.

Sometimes it may be sufficient to switch the bulks, as shown on the left below. But more often that one would like, we have to implement bootstrapped switches as shown on the right.

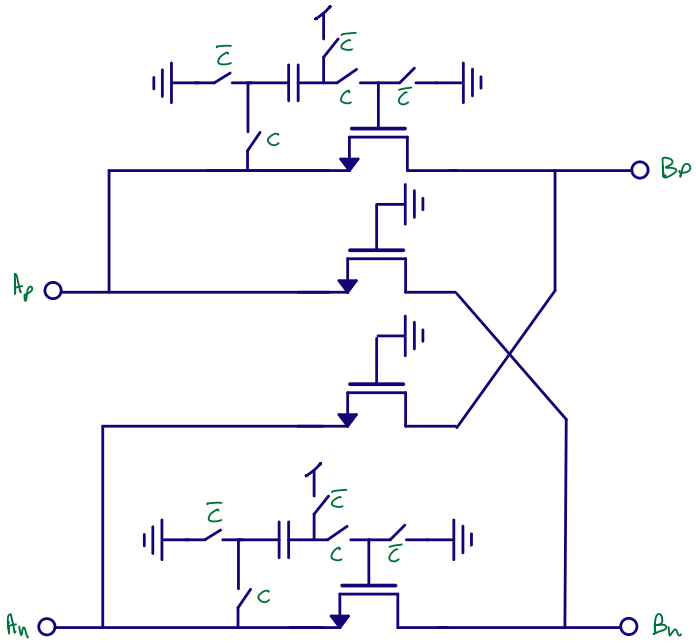


The switch I used in my JSSC SAR is a fully differential boostrapped switch with cross coupled dummy transistors. The JSSC SAR I've also ported to GF130NM, as shown below. The switch is at the bottom.

wulffern/sun_sar9b_sky130nm

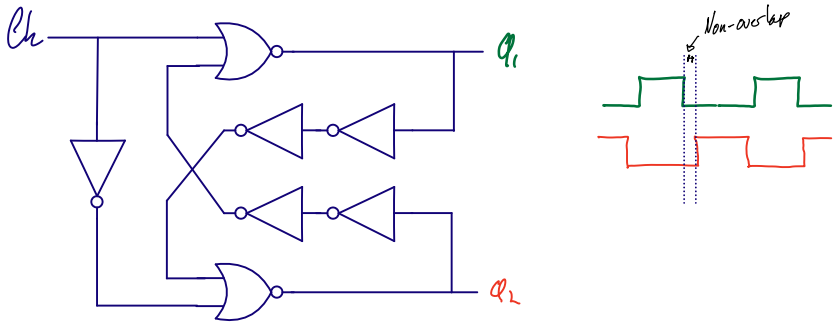


looks like the one below.



11.5.4.3 Non-overlapping clocks

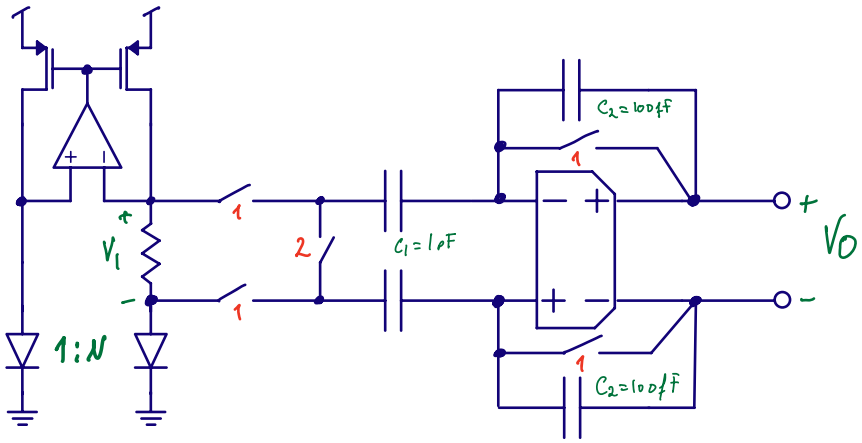
The non-overlap generator is standard. Use the one shown below. Make sure you simulate that the non-overlap is sufficient in all corners.



11.5.5 Example

In the circuit below there is an example of a switched capacitor circuit used to increase the ΔV_D across the resistor. We can accurately set the gain, and thus the equation for the differential output will be

$$V_O(z) = 10\frac{kT}{q}\ln(N)z^{-1}$$



11.6 Want to learn more?

Blind Multiband Signal Reconstruction: Compressed Sensing for Analog Signal

Comparator-based switched-capacitor pipelined analog-to-digital converter with comparator preset, and comparator delay compensation

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

A 10-bit 50-MS/s SAR ADC With a Monotonic Capacitor Switching Procedure

Low Voltage, Low Power, Inverter-Based Switched-Capacitor Delta-Sigma Modulator

Ring Amplifiers for Switched Capacitor Circuits

A Switched-Capacitor RF Power Amplifier

Design of Active N-Path Filters

Oversampling and Sigma-Delta ADCs

12

Keywords: Quantization, OSR, NEG FB, STF, NTF, SAR, First Order, SC SD, CT SD, INCR, FOM

12.1 ADC state-of-the-art

The performance of an analog-to-digital converter is determined by the effective number of bits (ENOB), the power consumption, and the maximum bandwidth. The effective number of bits contain information on the linearity of the ADC. The power consumption shows how efficient the ADC is. The maximum bandwidth limits what signals we can sample and reconstruct in digital domain.

Many years ago, Robert Walden did a study of ADCs, one of the plot's is shown below.

1999, R. Walden: Analog-to-digital converter survey and analysis

There are obvious trends, the faster an ADC is, the less precise the ADC is (lower SNDR). There are also fundamental limits, Heisenberg tells us that a 20-bit 10 GS/s ADC is impossible, according to Walden.

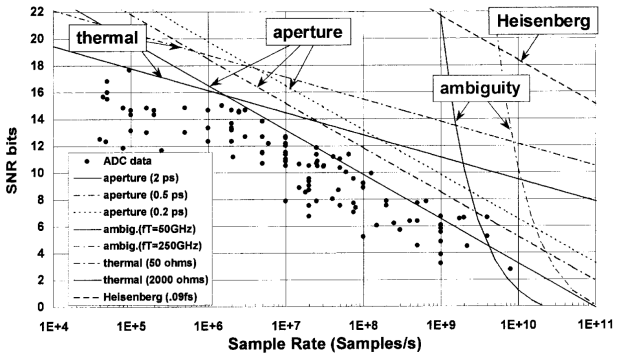


Fig. 7. Signal-to-noise ratio according to SNR-bits = (SNR(dB) − 1.76)/6.02. Three sets of curves show performance limiters due to thermal noise, aperture uncertainty, and comparator ambiguity. The Heisenberg limit is also displayed.

The uncertainty principle states that the precision we can determine position and the momentum of a particle is

$$\sigma_x \sigma_p \geq \frac{\hbar}{2}$$

. There is a similar relation of energy and time, given by

$$\Delta E \Delta t > \frac{h}{2\pi}$$

where ΔE is the difference in energy, and Δt is the difference in time.

You should take these limits with a grain of salt. The plot assumes 50 Ohm and 1 V full-scale. As a result, the "Heisenberg" line that appears to be unbreakable certainly is breakable. Just change the voltage to 100 V, and the number of bits can be much higher. Always check the assumptions.

A more recent survey of ADCs comes from Boris Murmann. He still maintains a list of the best ADCs from ISSCC and VLSI Symposium.

B. Murmann, ADC Performance Survey 1997-2023

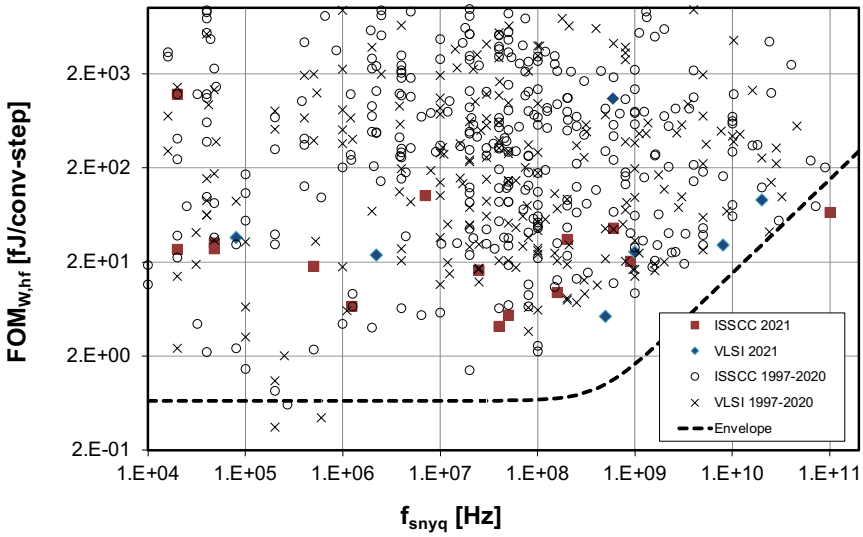
A common figure of merit for low-to-medium resolution ADCs is the Walden figure of merit, defined as

$$FOM_W = \frac{P}{2^B f_s}$$

Below 10 fJ/conv.step is good.

Below 1 fJ/conv.step is extreme.

In the plot below you can see the ISSCC and VLSI ADCs.



12.1.1 What makes a state-of-the-art ADC

People from NTNU have made some of the worlds best ADCs

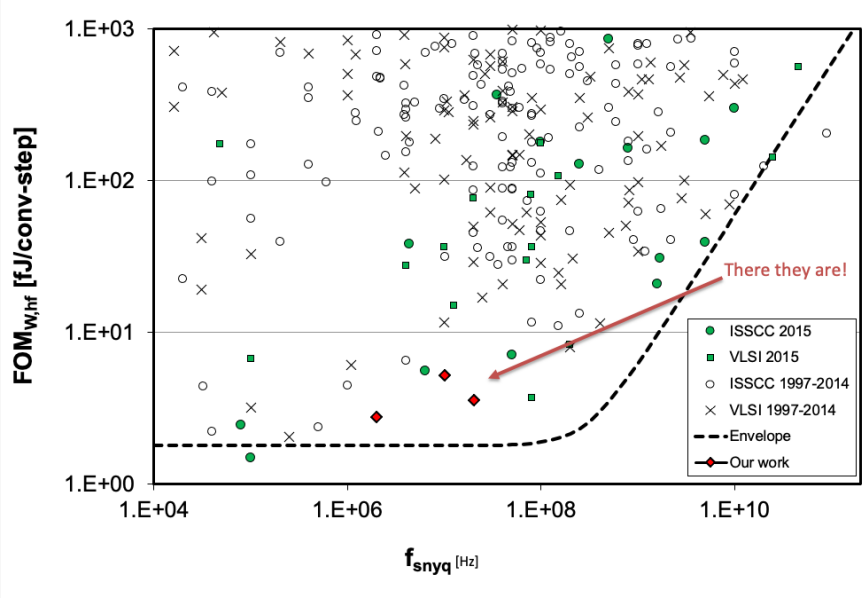
If you ever want to make an ADC, and you want to publish the measurements, then you must be better than most. A good algorithm for state-of-the-art ADC design is to first pick a sample rate with low number of data (blank spaces in the plot above), then

read the papers in the vicinity of the blank space to understand the application, then set a target FOM which is best in world, then try and find a ADC architecture that can achieve that FOM.

That's pretty much the algorithm I, and others, have followed to make state-of-the-art ADCs. A few of the NTNU ADCs are:

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

[2] A 68 dB SNDR Compiled Noise-Shaping SAR ADC With On-Chip CDAC Calibration



In order to publish, there must be something new. Preferably a new circuit. Below is the circuit from [1]. It's a standard successive-approximation register (SAR) analog-to-digital converter.

The differential input signal is sampled on a capacitor array where the bottom plate is connected to either VSS or VREF. Once the voltage is sampled, the comparator will decide whether the differential voltage is larger, or smaller than 0. Depending on the decision, the MSB capacitors (left-most) in figure will switch the bottom plate in order to effectively subtract a voltage equivalent to half the VREF voltage.

The comparator makes another decision, and 1/4'th the VREF voltage is subtracted or added. Then 1/8'th and so on implementing a binary search to find the input voltage.

The "bit-cycling" (binary-search) loop is self-timed, as such, when the comparator has made a decision, the next cycle starts.

In (b) we can see the enable flip-flop for the next stage. The CK bar is the sample clock, as such, A is high during sampling. The output of the comparator (P and N) is low.

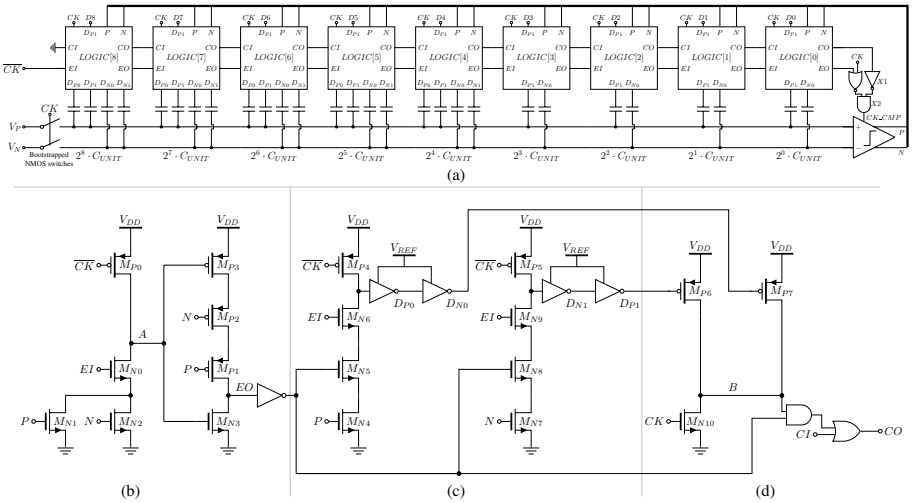
As soon as the comparator makes a decision, P or N goes high, A will be pulled low, if EI is enabled.

In (c) we can see that the bottom plate of the capacitors D_{P0}, D_{P1}, D_{N0}, and D_{N1}, are controlled by P and N.

In (d) we can see that the bottom plate of the capacitors also used to set the comparator clock low again (CO), resetting the comparator, and pulling P and N low, which in (b) enables the next SAR logic state.

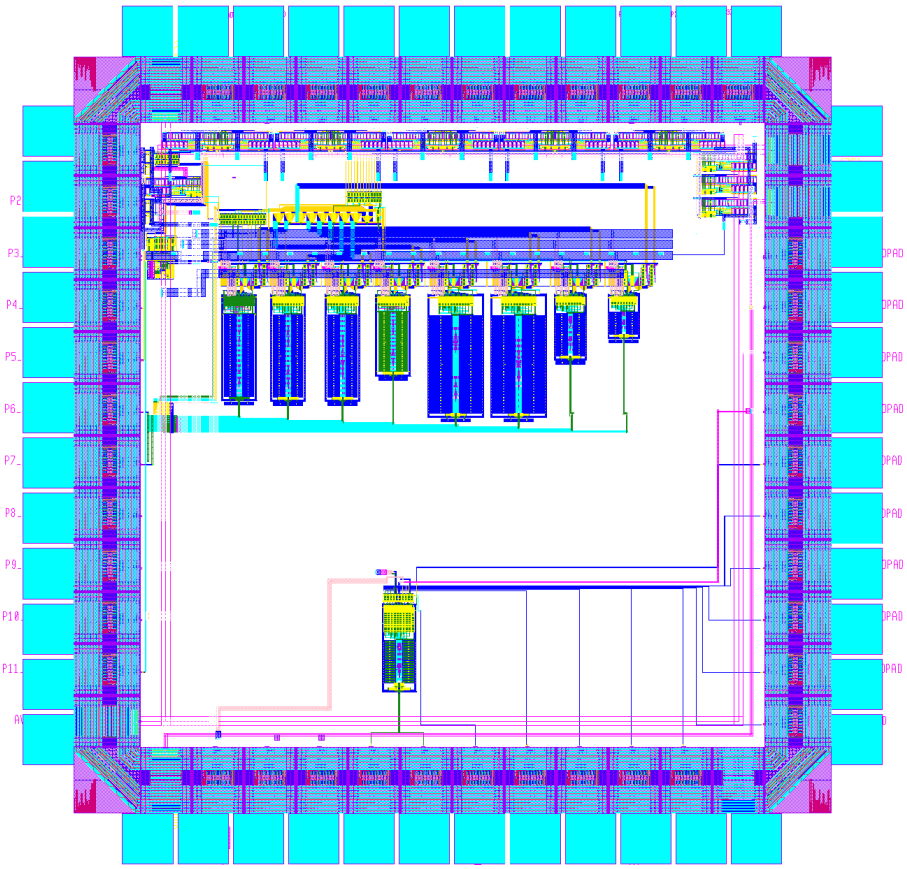
How fast the D_{XX} settle depend on the size of the capacitors, as such, the comparator clock will be slow for the MSB, and very fast for the LSB. This was my main circuit contribution in the paper. I think it's quite clever, because both the VDD and the capacitor corner will change the settling time. It's important that the capacitor values fully settle before the next comparator decision, and as a result of the circuit in (c,d) the delay is automatically adjusted.

For further details see the paper.

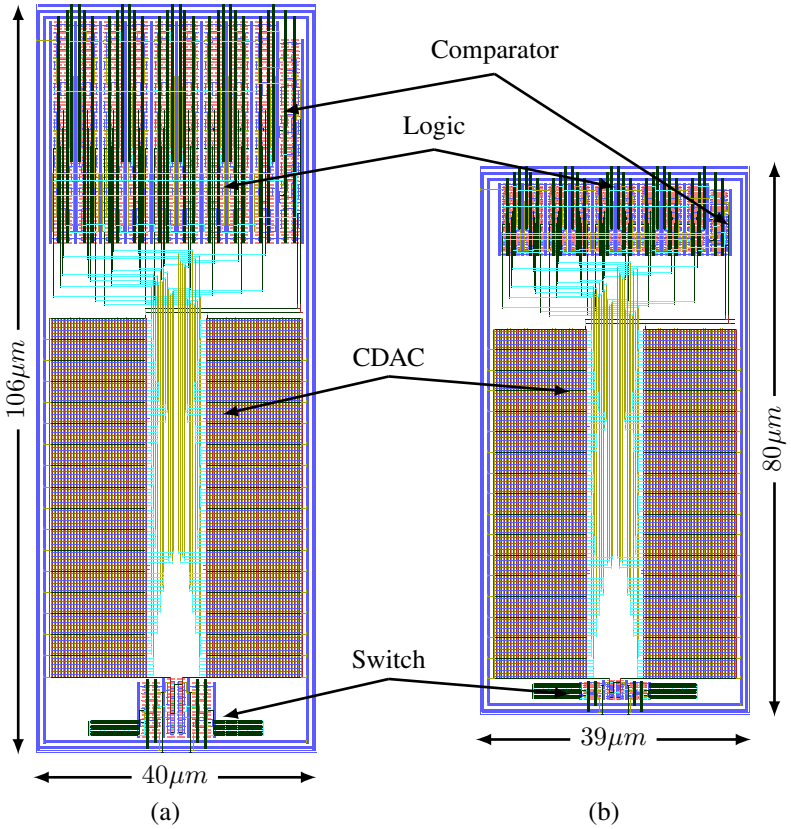


For state-of-the-art ADC papers it's not sufficient with the idea, and simulation. There must be proof that it actually works. No-one will really believe that the ADC works until there is measurements of an actual taped out IC.

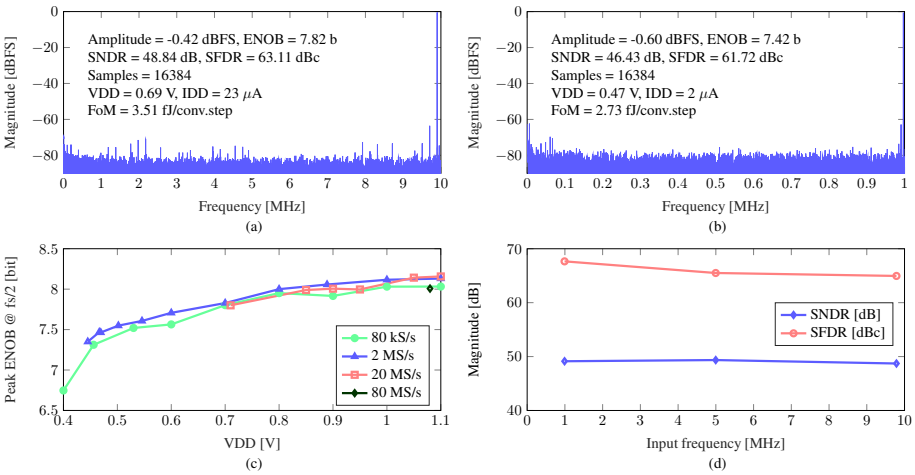
Below you can see the layout of the IC I made for the paper. Notice that there are 9 ADCs. I had many ideas that I wanted to try out, and I was not sure what would actually be state of the art. As a result, I taped out multiple ADCS.



The two ADCs that I ended up using in the paper is shown below. The one on the left was made with 180 nm IO transistors, while the one on the right was made with core-transistors. Notice that the layout of the two is quite similar.



Once taped out, and many months of waiting, a few months of measurement in the lab, I had some results that would be good enough to qualify for the best conference, and luckily the best journal.

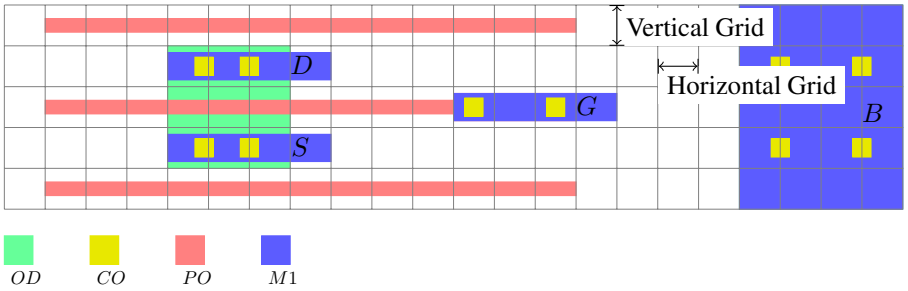


Comparing my ADCs to others, we can see that the FOM is similar to others. Based on the FOM it might not be clear why the paper was considered state-of-the-art.

The circuit technique mentioned above would not have been enough to qualify. The big thing was the “Compiled” line. Compared to the other “Compiled” mine was 300 times better, and on par with other state-of-the-art.

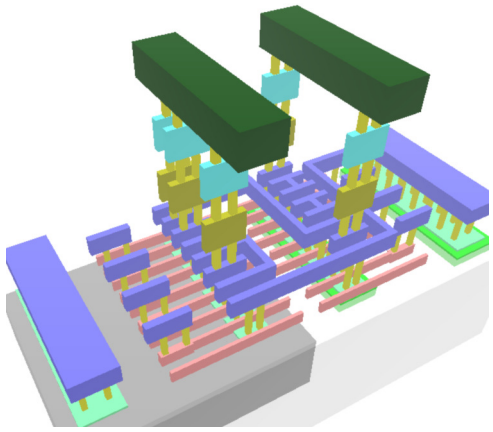
	Weaver [5]	Harpe [9]	Patil [10]	Liu [11]	This work	
Technology (nm)	90	90	28 FDSOI	28	28 FDSOI	
Fsample (MS/s)	21	2	No sampling	100	2	20
Core area (mm^2)	0.18	0.047	0.0032	0.0047	0.00312	
SNDR (dB)	34.61	57.79	40	64.43	46.43	48.84
SFDR (dBc)	40.81	72.33	30	75.42	61.72	63.11
ENOB (bits)	5.45	6.7 - 9.4	6.35	10.41	7.42	7.82
Supply (V)	0.7	0.7	0.65	0.9	0.47	0.69
Pwr (μW)	1110	1.64 -3.56	24	350	0.94	15.87
Compiled	Yes	No	No	No	Yes	
FoM (fJ/c.step)	838	2.8 - 6.6	3.7	2.6	2.7	3.5

The big thing was how I made the ADC. I started with a definition of a transistor, as shown below



And then wrote a compiler (in Perl, later C++ ciccreator) to compile a object definition file, a SPICE netlist and a technology rule file into the full ADC layout.

In (a) you can see one of the cells in the SAR logic, (b) is the spice file, and (c) is the definition of the routing. The numbers to the right in the routing creates the paths shown in (d).



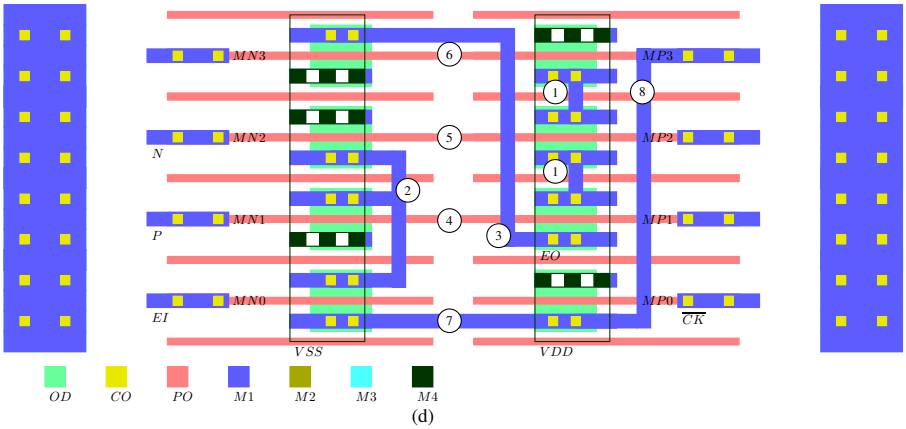
(a)

```
.SUBCKT SAREMX1_CV P  N EI EO CK_N AVDD AVSS
  MN0 N3   EI   A    AVSS  NCHDL
  MN1 N3   P    AVSS AVSS  NCHDL
  MN2 AVSS N    N3   AVSS  NCHDL
  MN3 EO   A    AVSS AVSS  NCHDL
  MP0 AVDD CK_N A    AVSS  PCHDL
  MP1 N2   P    EO   AVSS  PCHDL
  MP2 N1   N    N2   AVSS  PCHDL
  MP3 AVDD A    N1   AVSS  PCHDL
.ENDS
```

(b)

```
{ "name": "SAREMX1_CV",
  "class" : "Layout::LayoutDigitalCell",
  "addConnectivityRoutes": [
       ["M1","N1|N2","||",""],       1
       ["M1","N3","-|",""],          2
       ["M1","EO","--|-","onTopR"]   3
         ],
    "addDirectedRoutes" : [
      ["PO","P","MN1:G-MP1:G"],      4
      ["PO","N","MN2:G-MP2:G"],      5
      ["PO","A","MN3:G-MP3:G"],      6
      ["M1","A","MN0:S-MP0:S"],      7
      ["M1","A","MP0:S-|--MP3:G"]    8
      ]
    }
}
```

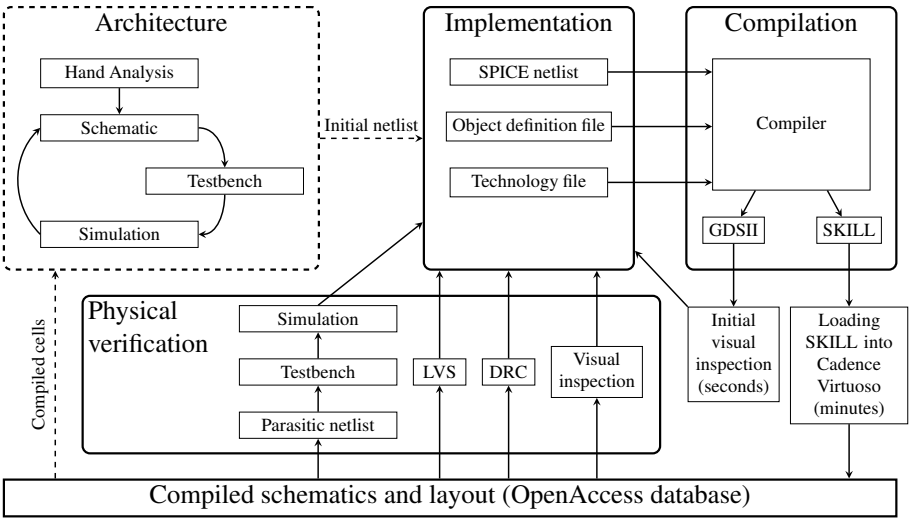
(c)



(d)

The implementation is the SPICE netlist, and the object definition file (JSON) and the rule file.

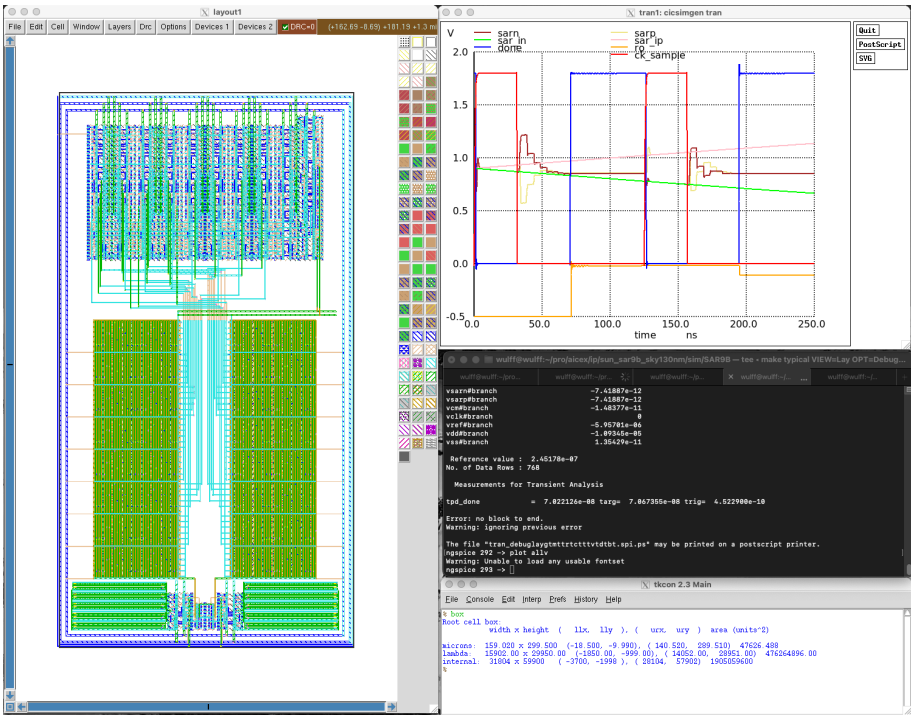
What I really like is the fact that the compilation could generate GDSII or SKILL, or these days, Xschem schematics and Magic layout.



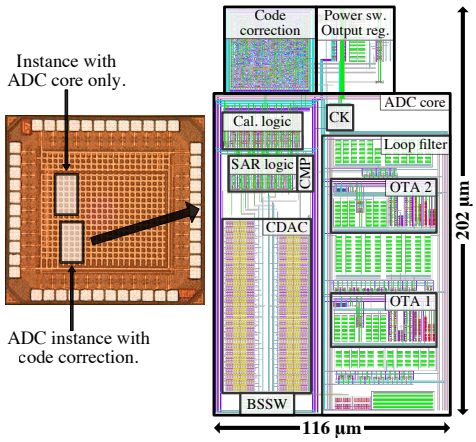
The cool thing with a compiled ADC is that it's easy to port between technologies. Since the original ADC, I've ported the ADC to multiple closed PDKs (22 nm FDSOI, 22 nm, 28 nm, 55 nm, 65

nm and 130nm). In the summer of 2022 I made an open source port to skywater 130nm.

SUN_SAR9B_SKY130NM



One of my Ph.D students built on-top on my work, and made a noise-shaped compiled SAR ADC, shown below, more on that later.



12.1.2 High resolution FOM

For high-resolution ADCs, it's more common to use the Schreier figure of merit, which can also be found in

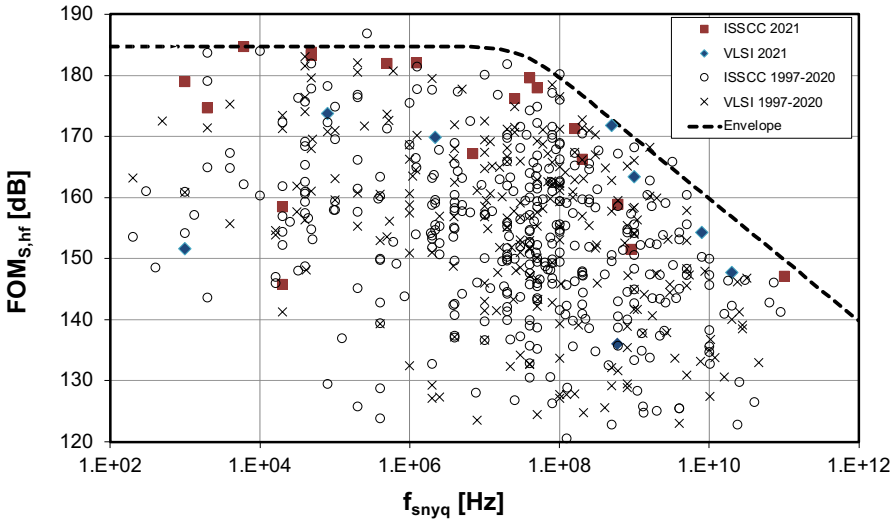
B. Murmann, ADC Performance Survey 1997-2022 (ISSCC & VLSI Symposium)

The Walden figure of merit assumes that thermal noise does not constrain the power consumption of the ADC, which is usually true for low-to-medium resolution ADCs. To keep the Walden FOM you can double the power for a one-bit increase in ENOB. If the ADC is limited by thermal noise, however, then you must quadruple the capacitance (reduce kT/C noise power) for each 1-bit ENOB increase. Accordingly, the power must also go up four times.

For higher resolution ADC the power consumption is set by thermal noise, and the Schreier FOM allows for a 4x power consumption increase for each added bit.

$$FOM_S = SNDR + 10\log\left(\frac{f_s/2}{P}\right)$$

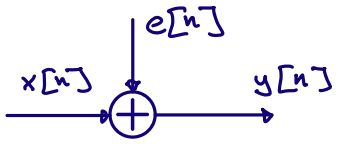
Above 180 dB is extreme



12.2 Quantization

Sampling turns continuous time into discrete time. Quantization turns continuous value into discrete value. Any complete ADC is always a combination of sampling and quantization.

In our mathematical drawings of quantization we often define $y[n]$ as the output, the quantized signal, and $x[n]$ as the discrete time, continuous value input, and we add some "noise", or "quantization noise" $e[n]$, where $x[n] = y[n] - e[n]$.



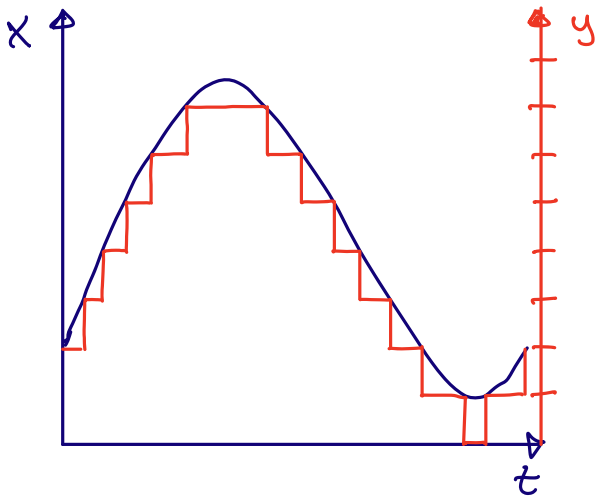
Maybe you've even heard the phrase "Quantization noise is white" or "Quantization noise is a random Gaussian process"?

I'm here to tell you that you've been lied to. Quantization noise is not white, nor is it a Gaussian process. Those that have lied to you may say "yes, sure, but for high number of bits it can be considered white noise". I would say that's similar to saying "when you look at the earth from the moon, the surface looks pretty smooth without bumps, so let's say the earth is smooth with no mountains".

I would claim that it's an unnecessary simplification. It's obvious to most that the earth would appear smooth from really far away, but they would not be surprised by Mount Everest, since they know it's not smooth. An Alien that has been told that the earth is smooth, would be surprised to see Mount Everest.

But if Quantization noise is not white, what is it?

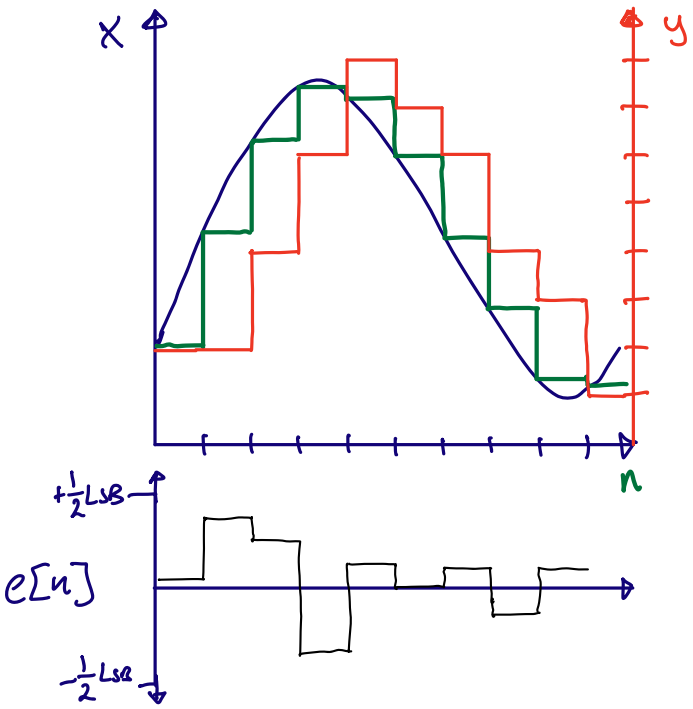
The figure below shows the input signal x and the quantized signal y.



To see the quantization noise, first take a look at the sample and held version of x in green in the figure below. The difference between the green (x at time n) and the red (y) would be our quantization noise e

The quantization noise is contained between $+\frac{1}{2}$ Least Significant Bit (LSB) and $-\frac{1}{2}$ LSB.

This noise does not look random to me, but I can't see what it is, and I'm pretty sure I would not be able to work it out either.



Luckily, there are people in this world that love mathematics, and that can delve into the details and figure out what $e[n]$ is. A guy called Blachman wrote a paper back in 1985 on quantization noise.

See The intermodulation and distortion due to quantization of sinusoids for details

In short, quantization noise is defined as

$$e_n(t) = \sum_{p=1}^{\infty} A_p \sin p\omega t$$

where p is the harmonic index, and

$$A_p = \begin{cases} \delta_{p1} A + \sum_{m=1}^{\infty} \frac{2}{m\pi} J_p(2m\pi A) & , p = \text{odd} \\ 0 & , p = \text{even} \end{cases}$$

$$\delta_{p1} \begin{cases} 1 & , p = 1 \\ 0 & , p \neq 1 \end{cases}$$

and

$$J_p(x)$$

is a Bessel function of the first kind, A is the amplitude of the input signal.

If we approximate the amplitude of the input signal as

$$A = \frac{2^n - 1}{2} \approx 2^{n-1}$$

where n is the number of bits, we can rewrite as

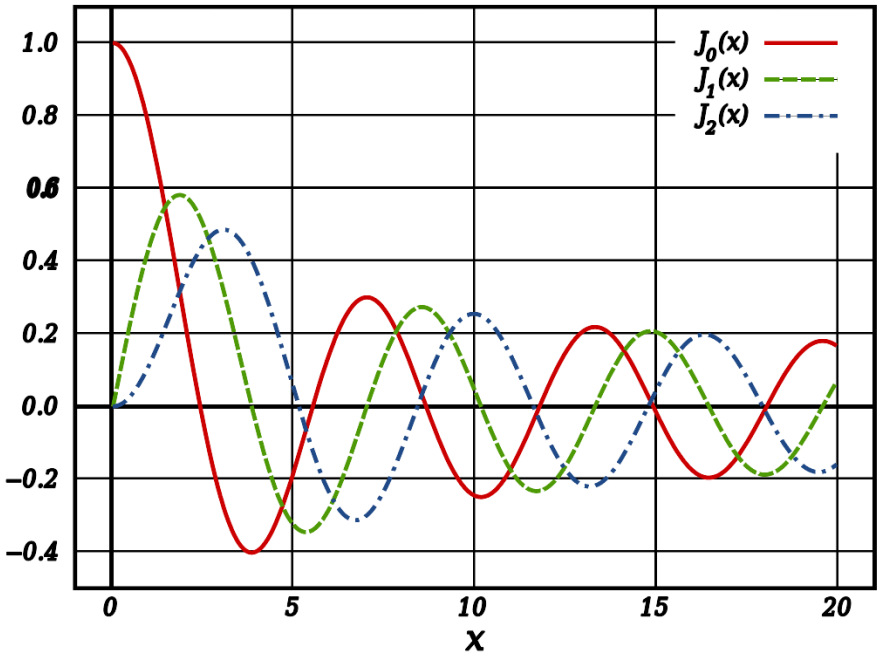
$$e_n(t) = \sum_{p=1}^{\infty} A_p \sin p\omega t$$

$$A_p = \delta_{p1} 2^{n-1} + \sum_{m=1}^{\infty} \frac{2}{m\pi} J_p(2m\pi 2^{n-1}), p = odd$$

Obvious, right?

I must admit, it's not obvious to me. But I do understand the implications. The quantization noise is an infinite sum of input signal odd harmonics, where the amplitude of the harmonics is determined by a sum of a Bessel function.

A Bessel function of the first kind looks like this



So I would expect the amplitude to show signs of oscillatory behavior for the harmonics. That's the important thing to remember. The quantization noise is **odd harmonics of the input signal**

The mean value is zero

$$\overline{e_n(t)} = 0$$

and variance (mean square, since mean is zero), or noise power, can be approximated as

$$\overline{e_n(t)^2} = \frac{\Delta^2}{12}$$

12.2.1 Signal to Quantization noise ratio

Assume we wanted to figure out the resolution, or effective number of bits for an ADC limited by quantization noise. A power ratio, like signal-to-quantization noise ratio (SQNR) is one way to represent resolution.

Take the signal power, and divide by the noise power

$$SQNR = 10\log\left(\frac{A^2/2}{\Delta^2/12}\right) = 10\log\left(\frac{6A^2}{\Delta^2}\right)$$

$$\Delta = \frac{2A}{2^B}$$

$$SQNR = 10\log\left(\frac{6A^2}{4A^2/2^B}\right) = 20B\log 2 + 10\log 6/4$$

$$SQNR \approx 6.02B + 1.76$$

You may have seen the last equation before, now you know where it comes from.

12.2.2 Understanding quantization

Below I've tried to visualize the quantization process q.py.

The left most plot is a sinusoid signal and random Gaussian noise. The signal is not a continuous time signal, since that's not possible on a digital computer, but it's an approximation.

The plots are FFTs of a sinusoidal signal combined with noise. These are complex FFTs, so they show both negative and positive frequencies. The x-axis is the FFT bin (not the frequency). Notice that there are two spikes, which should not be surprising, since a sinusoidal signal is a combination of two frequencies.

$$sin(x) = \frac{e^{ix} - e^{-ix}}{2i}$$

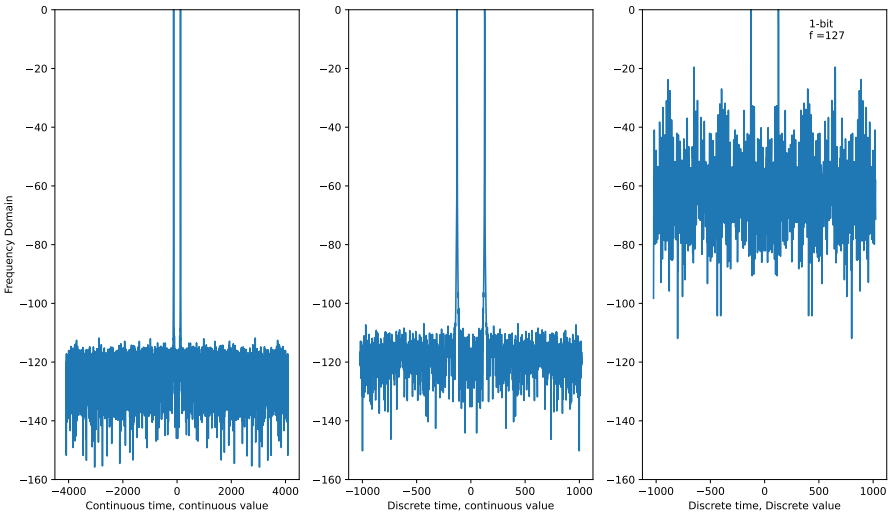
The second plot from the left is after sampling, notice that the noise level increases. The increase in the noise level should be due to noise folding, and reduced number of points in the FFT, but I have not confirmed (maybe you could confirm?).

The right plot is after quantization, where I've used the function below.

```
def adc(x,bits):
    levels = 2**bits
    y = np.round(x*levels)/levels
    return y
```

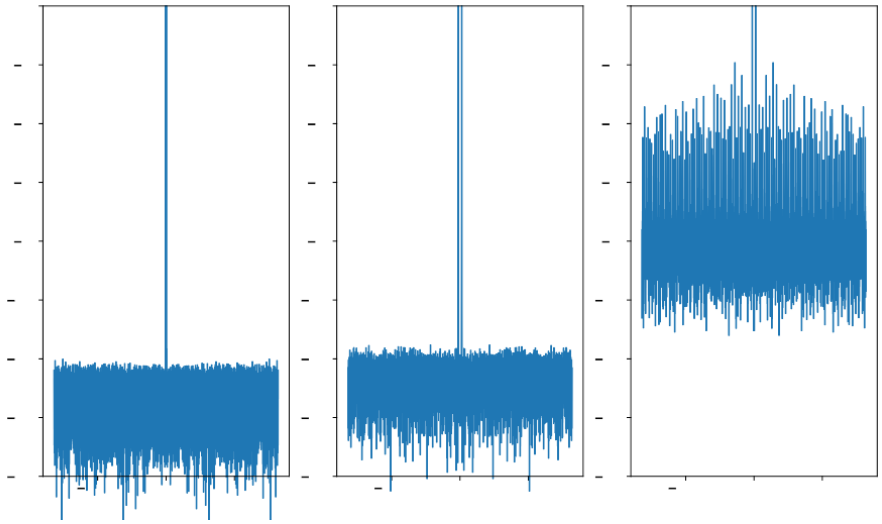
I really need you to internalize a few things from the right most plot. Really think through what I'm about to say.

Can you see how the noise (what is not the two spikes) is not white? White noise would be flat in the frequency domain, but the noise is not flat.



If you run the python script you can zoom in and check the highest spikes. The fundamental is at 127, so odd harmonics would be 381, 635, 889, and from the function of the quantization noise we would expect those to be the highest harmonics (at least when we look at the Bessel function), however, we can see that it's close, but that bin 396 is the highest. Is the math's wrong?

No, the math is correct. Never bet against mathematics. If you change the python script to reduce the frequency, `fdivide=2**9`, and increase number of points, `N=2**16`, as in the plot below, you'll see it's the 11'th harmonic that is highest.



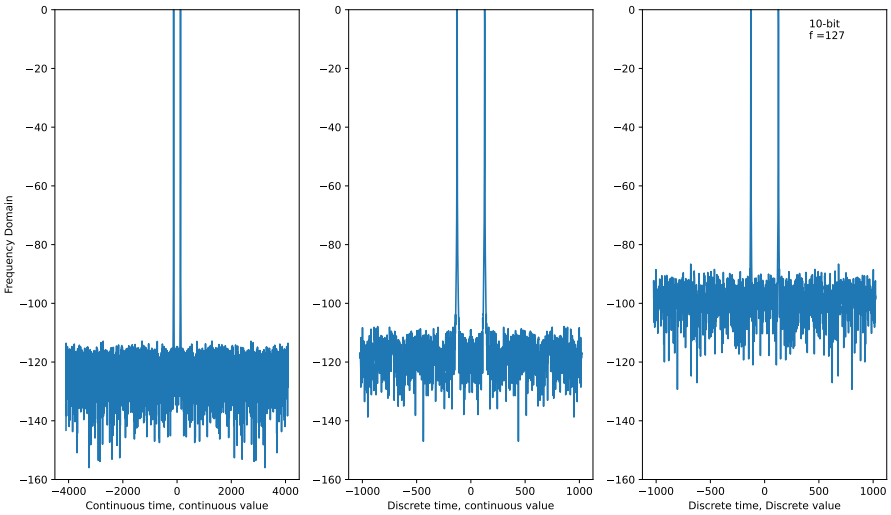
All the other spikes are the odd harmonics above the sample rate that fold. The infinite sum of harmonics will fold, some in-phase, some out of phase, depending on the sign of the Bessel function.

From the function for the amplitude of the quantization noise for harmonic indices higher than $p = 1$

$$A_p = \sum_{m=1}^{\infty} \frac{2}{m\pi} J_p(2m\pi 2^{n-1}), \text{ p=odd}$$

we can see that the input to the Bessel function increases faster for a higher number of bits n. As such, from the Bessel function figure above, I would expect that the sum of the Bessel function is a lower value. Accordingly, the quantization noise reduces at higher number of bits.

A consequence is that the quantization noise becomes more and more uniform, as can be seen from the plot of a 10-bit quantizer below. That's why people say "Quantization noise is white", because for a high number of bits, it looks white in the FFT.



12.2.3 Why you should care about quantization noise

So why should you care whether the quantization noise looks white, or actually is white? A class of ADCs called oversampling and sigma-delta modulators rely on the assumption that quantization noise **is** white. In other words, the cross-correlation between noise components at different time points is zero. As such the noise power sums as a sum of variance, and we can increase the signal-to-noise ratio.

We know that assumption to be wrong though, **quantization noise is not white**. For noise components at harmonic frequencies the cross-correlation will be high. As such, when **we** design oversampling or sigma-delta based ADC **we** will include some form of dithering (making quantization noise whiter). For example, before the actual quantizer we inject noise, or we make sure that the thermal noise is high enough to dither the quantizer.

Everybody that thinks that quantization noise **is** white will design non-functioning (or sub-optimal) oversampling and sigma-delta ADCs. That's why you should care about the details around quantization noise.

12.3 Oversampling

Assume a signal $x[n] = a[n] + b[n]$ where a is a sampled sinusoid and b is a random process where cross-correlation is zero for any time except for $n = 0$. Assume that we sum two (or more) equally spaced signal components, for example

$$y = x[n] + x[n + 1]$$

What would the signal to noise ratio be for y?

12.3.1 Noise power

Our mathematician friends have looked at this, and as long the noise signal b **is random** then the noise power for the oversampled signal $b_{osr} = b[n] + b[n+1]$ will be

$$\overline{b_{osr}^2} = OSR \times \overline{b^2}$$

where OSR is the oversampling ratio. If we sum two time points the $OSR = 2$, if we sum 4 time points the $OSR = 4$ and so on.

For fun, let's go through the mathematics

Define $b_1 = b[n]$ and $b_2 = b[n+1]$ and compute the noise power

$$\overline{(b_1 + b_2)^2} = \overline{b_1^2 + 2b_1b_2 + b_2^2}$$

Let's replace the mean with the actual function

$$\frac{1}{N}\sum_{n=0}^{N}\left(b_1^2 + 2b_1b_2 + b_2^2\right)$$

which can be split up into

$$\frac{1}{N}\sum_{n=0}^{N} b_1^2 + \frac{1}{N}\sum_{n=0}^{N} 2b_1b_2 + \frac{1}{N}\sum_{n=0}^{N} b_2^2$$

we've defined the cross-correlation to be zero, as such

$$\overline{(b_1 + b_2)^2} = \frac{1}{N}\sum_{n=0}^{N} b_1^2 + \frac{1}{N}\sum_{n=0}^{N} b_2^2 = \overline{b_1^2} + \overline{b_2^2}$$

but the noise power of each of the b's must be the same as b, so

$$\overline{(b_1 + b_2)^2} = 2\overline{b^2}$$

12.3.2 Signal power

For the signal a we need to calculate the increase in signal power as OSR increases.

I like to think about it like this. a is low frequency, as such, samples n and $n+1$ is pretty much the same value. If the sinusoid has an amplitude of 1, then the amplitude would be 2 if we sum two samples. As such, the amplitude must increase with the OSR.

The signal power of a sinusoid is $A^2/2$, accordingly, the signal power of an oversampled signal must be $(OSR \times A)^2/2$.

12.3.3 Signal to Noise Ratio

Take the signal power to the noise power

$$\frac{(OSR \times A)^2/2}{OSR \times \overline{b^2}} = OSR \times \frac{A^2/2}{\overline{b^2}}$$

We can see that the signal to noise ratio increases with increased oversampling ratio, **as long as the cross-correlation of the noise is zero**

12.3.4 Signal to Quantization Noise Ratio

The in-band quantization noise for a oversampling ratio (OSR)

$$\overline{e_n(t)^2} = \frac{\Delta^2}{12OSR}$$

And the improvement in SQNR can be calculated as

$$SQNR = 10\log\left(\frac{6A^2}{\Delta^2/OSR}\right) = 10\log\left(\frac{6A^2}{\Delta^2}\right) + 10\log(OSR)$$

$$SQNR \approx 6.02B + 1.76 + 10\log(OSR)$$

For an OSR of 2 and 4 the SQNR improves by

$$10\log(2) \approx 3dB$$

and for OSR=4

$$10\log(4) \approx 6dB$$

which is roughly equivalent to a 0.5-bit per doubling of OSR

12.3.5 Python oversample

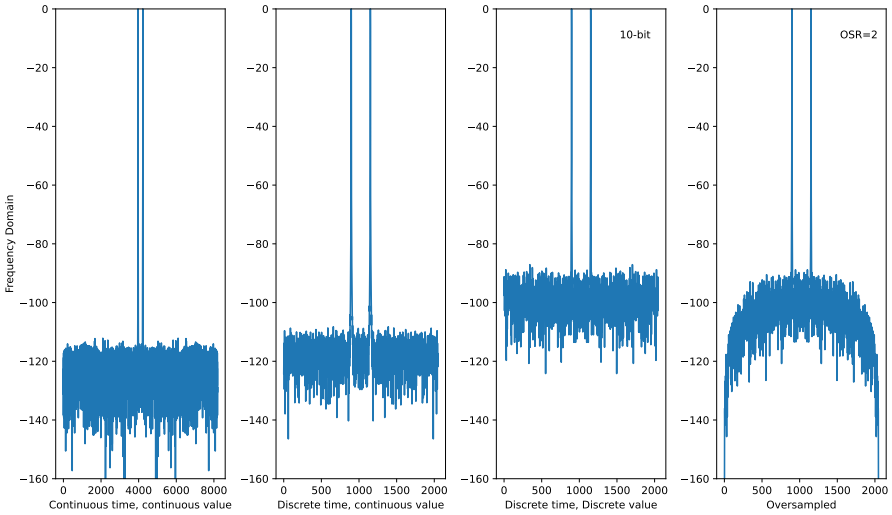
There are probably more elegant (and faster) ways of implementing oversampling in python, but I like to write the dumbest code I can, simply because dumb code is easy to understand.

Below you can see an example of oversampling. The `oversample` function takes in a vector and the OSR. For each index it sums OSR future values.

```
def oversample(x,OSR):
    N = len(x)
    y = np.zeros(N)

    for n in range(0,N):
        for k in range(0,OSR):
            m = n+k
            if (m < N):
                y[n] += x[m]
    return y
```

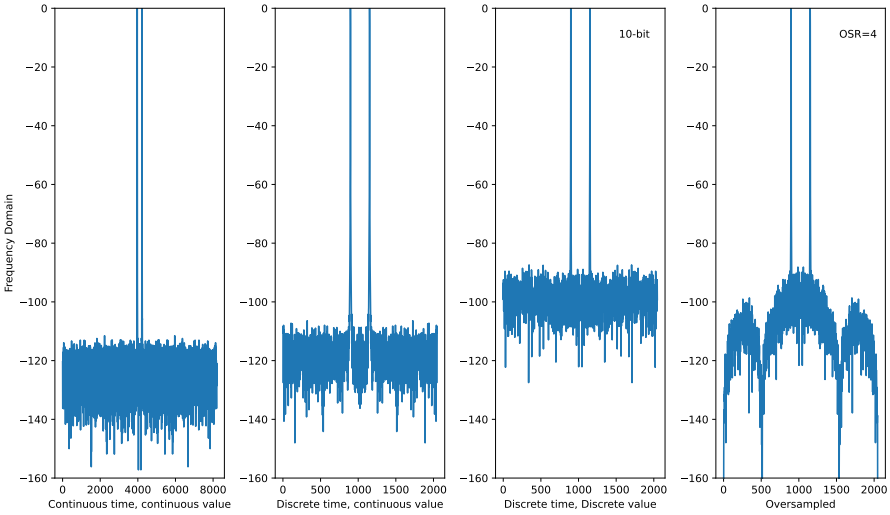
Below we can see the plot for OSR=2, the right most plot is the oversampled version.

The noise has all frequencies, and it's the high frequency components that start to cancel each other. An average filter (sometimes called a sinc filter due to the shape in the frequency domain) will have zeros at $\pm fs/2$ where the noise power tends towards zero.



The low frequency components will add, and we can notice how the noise power increases close to the zero frequency (middle of the x-axis).

For an OSR of 4 we can notice how the noise floor has 4 zero's.



The code for the plots is osr.py. I would encourage you to play a bit with the code, and make sure you understand oversampling.

12.4 Noise Shaping

Look at the OSR=4 plot above. The OSR=4 does decrease the noise compared to the discrete time discrete value plot, however, the noise level of the discrete time continuous value is much lower.

What if we could do something, add some circuitry, before the quantization such that the quantization noise was reduced?

That's what noise shaping is all about. Adding circuits such that we can "shape" the quantization noise. We can't make the quantization noise disappear, or indeed reduce the total noise power of the quantization noise, but we can reduce the quantization noise power for a certain frequency band.

But what circuitry can we add?

12.4.1 The magic of feedback

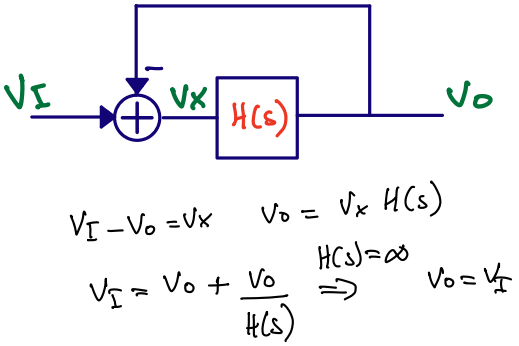
A generalized feedback system is shown below, it could be a regulator, a unity-gain buffer, or something else.

The output V_o is subtracted from the input V_i, and the error V_x is shaped by a filter $H(s)$.

If we make $H(s)$ infinite, then $V_o = V_i$. If you've never seen such a circuit, you might ask "Why would we do this? Could we not just use V_i directly?". There are many reasons for using a circuit like this, let me explain one instance.

Imagine we have a VDD of 1.8 V, and we want to make a 0.9 V voltage for a CPU. The CPU can consume up to 10 mA. One way to make a divide by two circuit is with two equal resistors connected between VDD and ground. We don't want the resistive divider to consume a large current, so let's choose 1 MOhm resistors. The current in the resistor divider would then be about 1 μA. We can't connect the CPU directly to the resistor divider, the CPU can draw 10 mA. As such, we need a copy of the voltage at the mid-point of the resistor divider that can drive 10 mA.

Do you see now why a circuit like the one below is useful? If not, you should really come talk to me so I can help you understand.

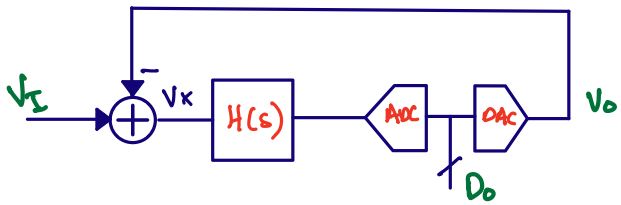


12.4.2 Sigma-delta principle

Let's modify the feedback circuit into the one below. I've added an ADC and a DAC to the feedback loop, and the D_o is now the output we're interested in. The equation for the loop would be

$$D_o = adc\left[H(s)\left(dac(D_o) - V_i\right)\right]$$

But how can we now calculate the transfer function $\frac{D_o}{V_i}$? Both adc and dac could be non-linear functions, so we can't disentangle the equation. Let's make assumptions.



12.4.2.1 The DAC assumption

Assumption 1: the dac is linear, such that $V_o = dac(D_o) = AD_o + B$, where A and B are scalar values.

The DAC must be linear, otherwise our noise-shaping ADC will not work.

One way to force linearity is to use a 1-bit DAC, which has only two points, so should be linear. For example

$$V_o = A \times D_o$$

, where $D_o \in (0, 1)$. Even a 1-bit DAC could be non-linear if A is time-variant, so $V_o[n] = A(t) \times D_o[n]$, this could happen if the reference voltage for the DAC changed with time.

I've made a couple noise shaping ADCs, and in the first one I made I screwed up the DAC. It turned out that the DAC current had a signal dependent component which lead to a non-linear behavior.

12.4.2.2 The ADC assumption

Assumption 2: the adc can be modeled as a linear function $D_o = adc(x) = x + e$, where e is **white noise source**

We've talked about this, the e is not white, especially for low-bit ADCs, so we usually have to add noise. Sometimes it's sufficient with thermal noise, but often it's necessary to add a random, or pseudo-random noise source at the input of the ADC.

12.4.2.3 The modified equation

With the assumptions we can change the equation into

$$D_o = adc\left[H(s)\left(V_i - dac(D_o)\right)\right] = H(s)\left(V_i - AD_o\right) + e$$

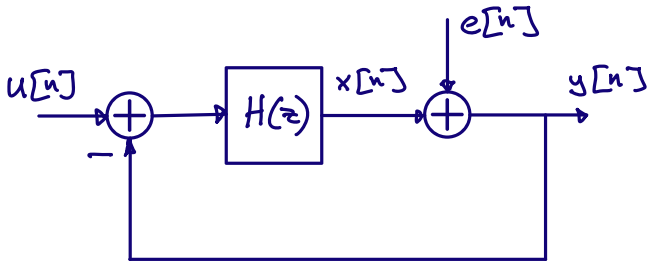
In noise-shaping texts it's common to write the above equation as

$$y = H(s)(u - y) + e$$

or in the sample domain

$$y[n] = e[n] + h * (u[n] - y[n])$$

which could be drawn in a signal flow graph as below.



in the Z-domain the equation would turn into

$$Y(z) = E(z) + H(z)\,[U(z) - Y(z)]$$

The whole point of this exercise was to somehow shape the quantization noise, and we're almost at the point, but to show how it works we need to look at the transfer function for the signal U and for the noise E.

12.4.3 Signal transfer function

Assume U and E are uncorrelated, and E is zero

$$Y = HU - HY$$

$$STF = \frac{Y}{U} = \frac{H}{1+H} = \frac{1}{1+\frac{1}{H}}$$

Imagine what will happen if H is infinite. Then the signal transfer function (STF) is 1, and the output Y is equal to our input U. That's exactly what we wanted from the feedback circuit.

12.4.4 Noise transfer function

Assume U is zero

$$Y = E + HY \rightarrow NTF = \frac{1}{1+H}$$

Imagine again what happens when H is infinite. In this case the noise-transfer function becomes zero. In other words, there is no added noise.

12.4.5 Combined transfer function

In the combined transfer function below, if we make $H(z)$ infinite, then $Y = U$ and there is **no added quantization noise**. I don't know how to make $H(z)$ infinite everywhere, so we have to choose at what frequencies it's "infinite".

$$Y(z) = STF(z)U(z) + NTF(z)E(z)$$

There are a large set of different $H(z)$ and I'm sure engineers will invent new ones. We usually classify the filters based on the number of zeros in the NTF, for example, first-order (one zero), second order (two zeros) etc. There are books written about sigma-delta modulators, and I would encourage you to read those to get a deeper understanding. I would start with Delta-Sigma Data Converters: Theory, Design, and Simulation.

12.5 First-Order Noise-Shaping

We want an infinite $H(z)$. One way to get an infinite function is an accumulator, for example

$$y[n+1] = x[n] + y[n]$$

or in the Z-domain

$$zY = X + Y \rightarrow Y(z-1) = X$$

which has the transfer function

$$H(z) = \frac{1}{z-1}$$

The signal transfer function is

$$STF = \frac{1/(z-1)}{1+1/(z-1)} = \frac{1}{z} = z^{-1}$$

and the noise transfer function

$$NFT = \frac{1}{1+1/(z-1)} = \frac{z-1}{z} = 1 - z^{-1}$$

In order calculate the Signal to Quantization Noise Ratio we need to have an expression for how the NTF above filters the quantization noise.

In the book they replace the z with the continuous time variable

$$z = e^{sT} \overset{s=j\omega}{\rightarrow} e^{j\omega T} = e^{j2\pi f/f_s}$$

inserted into the NTF we get the function below.

$$NTF(f) = 1 - e^{-j2\pi f/f_s}$$

$$= \frac{e^{j\pi f/f_s} - e^{-j\pi f/f_s}}{2j} \times 2j \times e^{-j\pi f/f_s}$$

$$= \sin\frac{\pi f}{f_s} \times 2j \times e^{-j\pi f/f_s}$$

The arithmetic magic is really to extract the $2j \times e^{-j\pi f/f_s}$ from the first expression such that the initial part can be translated into a sinusoid.

When we take the absolute value to figure out how the NTF changes with frequency the complex parts disappears (equal to 1)

$$|NFT(f)| = \left|2\sin\left(\frac{\pi f}{f_s}\right)\right|$$

The signal power for a sinusoid is

$$P_s = A^2/2$$

The in-band noise power for the shaped quantization noise is

$$P_n = \int_{-f_0}^{f_0} \frac{\Delta^2}{12} \frac{1}{f_s} \left[2 \sin \left(\frac{\pi f}{f_s} \right) \right]^2 dt$$

and with a bunch of tedious maths, we can get to the conclusion

$$\vdots$$

$$SQNR = 6.02B + 1.76 - 5.17 + 30 \log(OSR)$$

If we compare to pure oversampling, where the SQNR improves by $10 \log(OSR)$, a first order sigma-delta improves by $30 \log(OSR)$. That's a significant improvement.

12.5.1 SQNR and ENOB

Below is the signal-to-quantization noise ratio's for Nyquist up to second order sigma-delta.

$$SQNR_{nyquist} \approx 6.02B + 1.76$$

$$SQNR_{oversample} \approx 6.02B + 1.76 + 10 \log(OSR)$$

$$SQNR_{\Sigma\Delta 1} \approx 6.02B + 1.76 - 5.17 + 30 \log(OSR)$$

$$SQNR_{\Sigma\Delta 2} \approx 6.02B + 1.76 - 12.9 + 50 \log(OSR)$$

We could compute an effective number of bits, as shown below.

$$ENOB = (SQNR - 1.76)/6.02$$

The table below shows the effective number of bits for oversampling, and sigma-delta modulators. For a 1-bit quantizer, pure oversampling does not make sense at all. For first-order and second-order sigma delta modulators, and a OSR of 1024 we can get high resolution ADCs.

Assume 1-bit quantizer, what would be the maximum ENOB?

OSR	Oversampling	First-Order	Second Order
4	2	3.1	3.9

OSR	Oversampling	First-Order	Second Order
64	4	9.1	13.9
1024	6	15.1	23.9

12.6 Examples

12.6.1 Python noise-shaping

I want to demystify noise-shaping modulators. I think one way to do that is to show some code. You can find the code at sd_1st.py

Below we can see an excerpt. Again pretty stupid code, and I'm sure it's possible to make a faster version (for loops in python are notoriously slow).

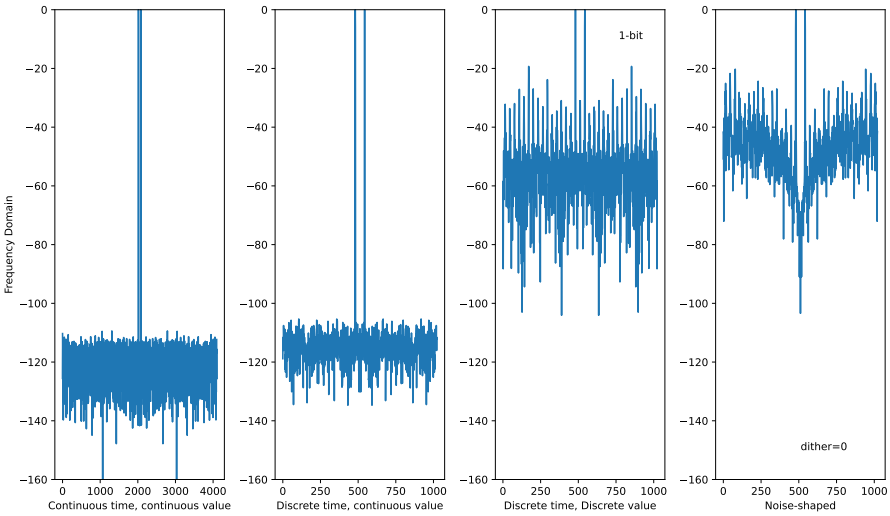
For each sample in the input vector u I compute the input to the quantizer x, which is the sum of the previous input to the quantizer and the difference between the current input and the previous output y_{sd}.

The quantizer generates the next y_{sd} and I have the option to add dither.

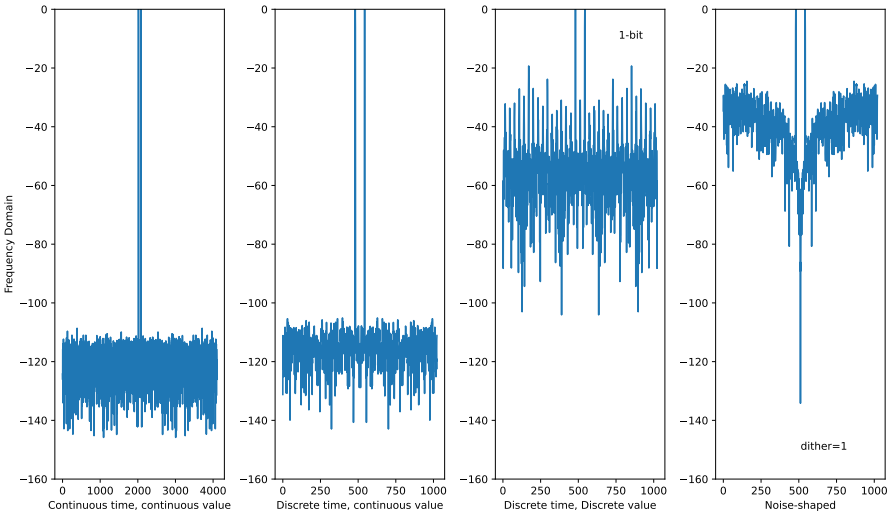
```
# u is discrete time, continuous value input
M = len(u)
y_sd = np.zeros(M)
x = np.zeros(M)
for n in range(1,M):
    x[n] = x[n-1] + (u[n]-y_sd[n-1])
    y_sd[n] = np.round(x[n]*2**bits
    + dither*np.random.randn()/4)/2**bits
```

The right-most plot is the one with noise-shaping. We can observe that the noise seems to tend towards zero at zero frequency, as we would expect. The accumulator above would have an infinite gain at infinite time (it's the sum of all previous values), as such, the NTF goes towards zero at 0 frequency.

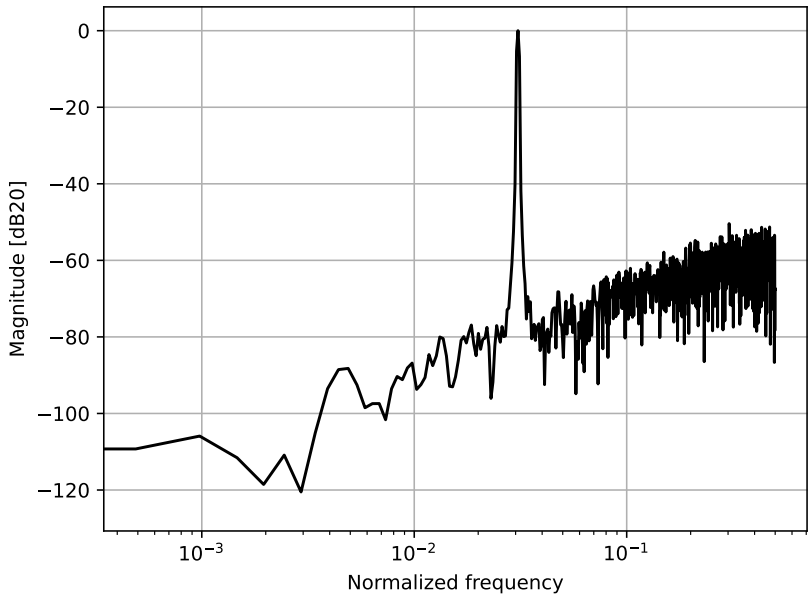
If we look at the noise we can also see the non-white quantization noise, which will degrade our performance. I hope by now, you've grown tired of me harping on the point that **quantization noise is not white**



In the figure below I've turned on dither, and we can see how the noise looks "better", which I know is not a qualitative statement, but ask anyone that's done 1-bit quantizers. It's important to have enough random noise.



In papers it's common to use a logarithmic x-axis for the power spectral density, as shown below. In the plot I only show the positive frequencies of the FFT. From the shape of the quantization noise we can also see the first order behavior.



12.6.2 The wonderful world of SD modulators

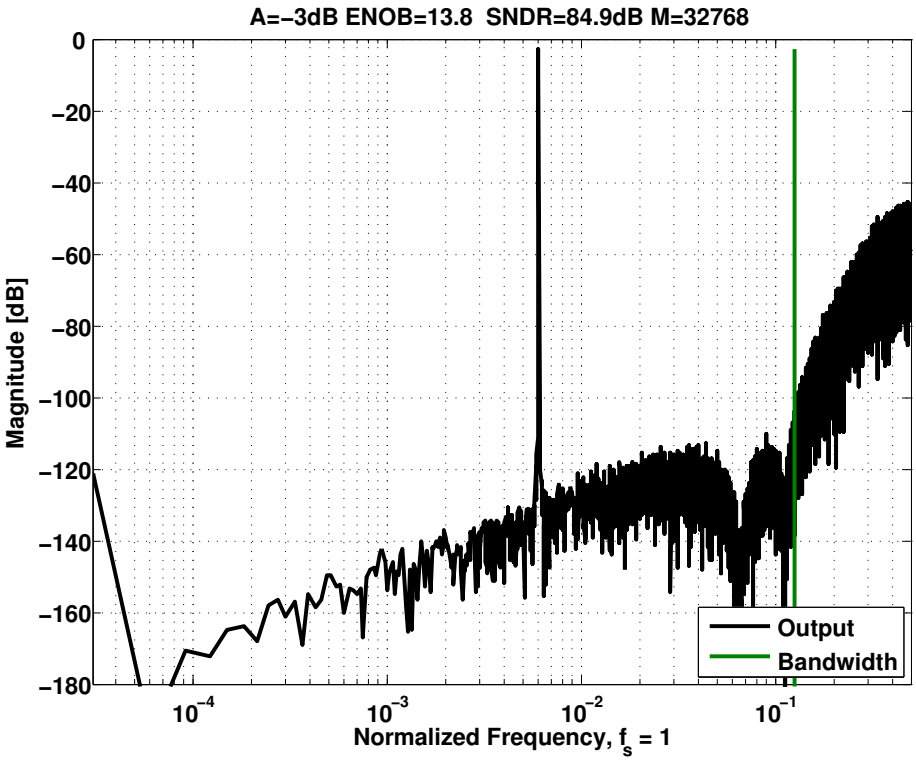
12.6.2.1 Open-Loop Sigma-Delta

On my Ph.D I did some work on

Resonators in Open-Loop Sigma-Delta Modulators

which was a pure theoretical work. The idea was to use modulo integrators (local control of integrator output swing) in front of large latency multi-bit quantizers to achieve a high SNR.

The plot below shows a fifth order NFT where there are two complex conjugate zeros, and a zero at zero frequency. With a higher order filter one can use a lower OSR, and still achieve high ENOB.



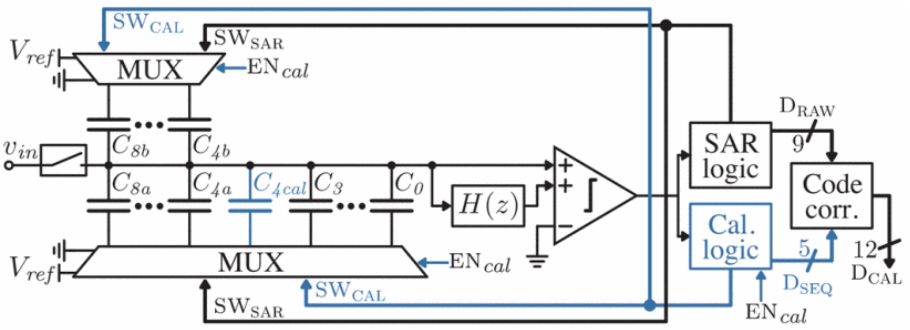
12.6.2.2 Noise Shaped SAR

One of my Ph.d students made a

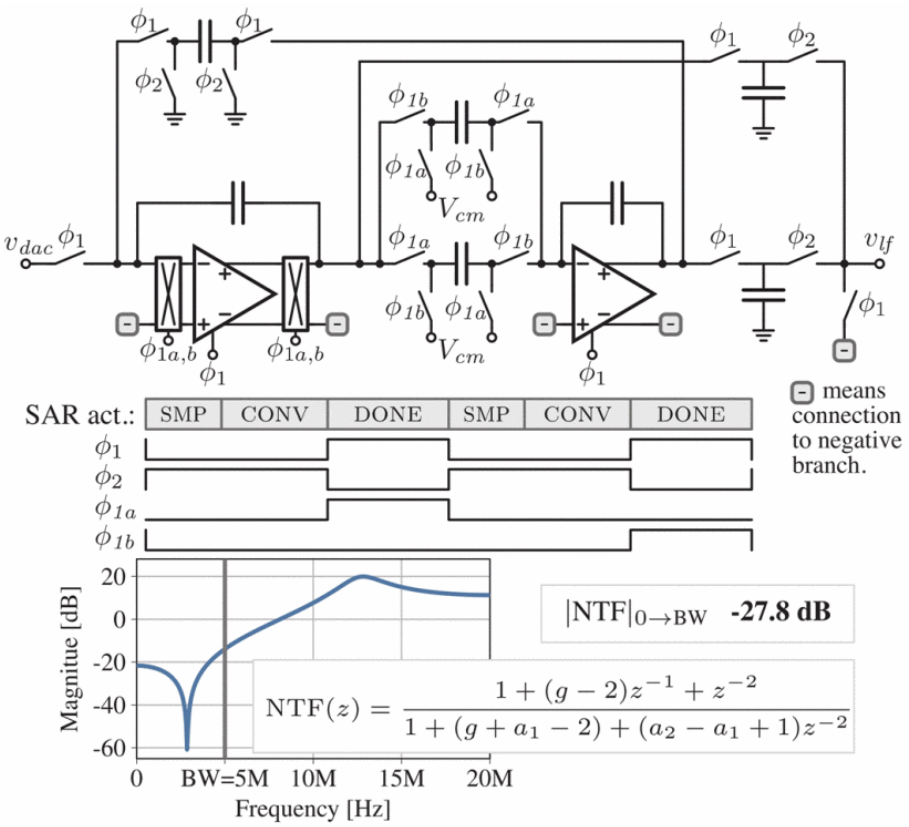
A 68 dB SNDR Compiled Noise-Shaping SAR ADC With On-Chip CDAC Calibration

In a SAR ADC, once the bit-cycling is complete, the analog value on the capacitors is the actual quantization error. That error can be fed to a loop filter, H(z), and amplified in the next conversion, accordingly a combination of SAR and noise-shaping.

In the paper the SD modulator was also used to calibrate the non-linearity in the CDAC, as the MSB capacitor won't be exactly N times larger than the smallest capacitor.



The loop filter was a switched cap loop filter, and we can see the NTF below. The first OTA made use of chopping to reduce the offset.



12.6.2.3 Control-Bounded ADCs

One of my current Ph.D students is working an even more advanced type of sigma-delta ADC. Actually, it's more a super-set of SD ADCs called control-bounded ADCs.

Design Considerations for a Low-Power Control-Bounded A/D Converter

A block diagram of a Leapfrog ADC version of a control-bounded ADC is shown below.

Here we're walking into advanced maths territory, but to simplify, I think it's correct to say that a control-bounded ADC seeks to control the local analog state, $x_n(t)$ such that no voltage is saturated. The digital control signals $s_n(t)$ are used to infer the state of the input $u(t)$ using a form of Bayesian Statistics.

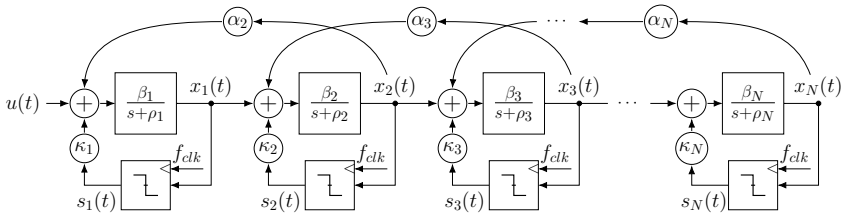
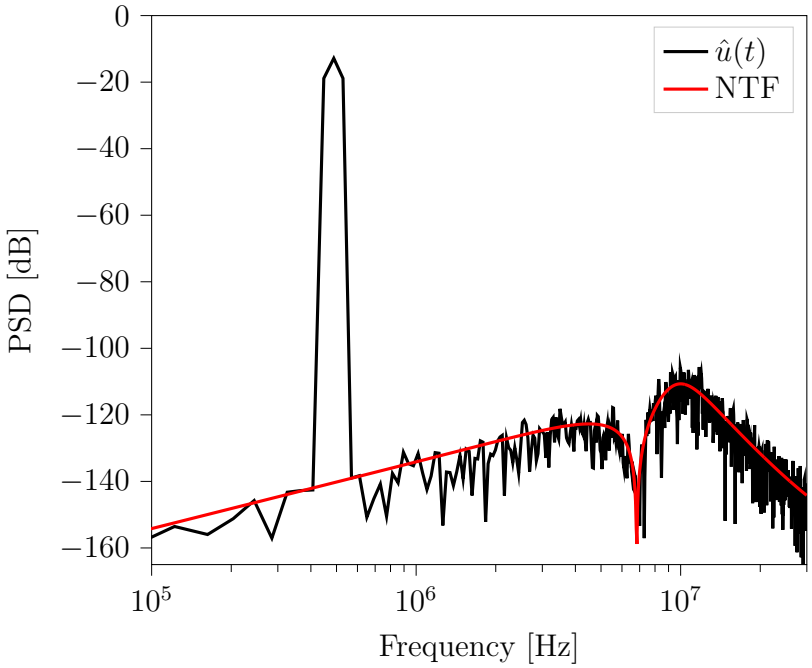


Figure 3.1: The general structure of the Leapfrog ADC

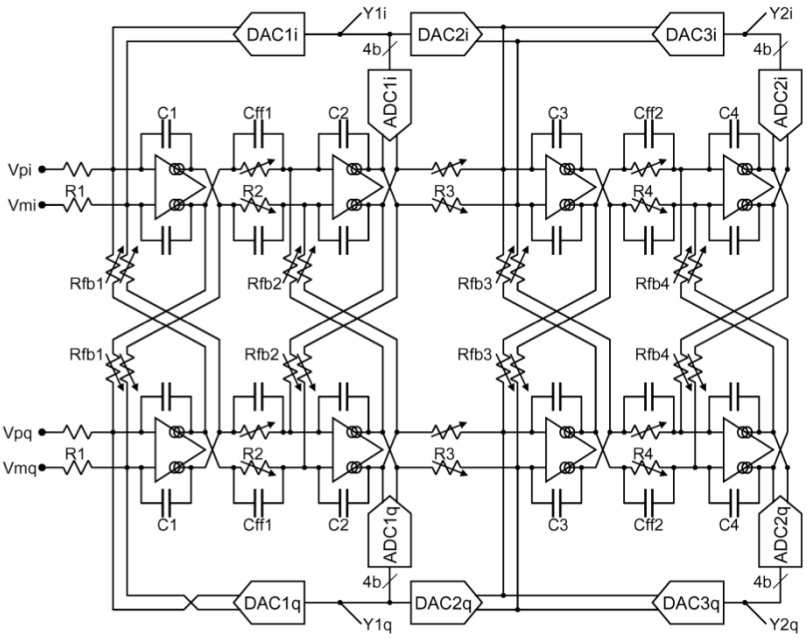
Below we can see a power spectral density plot of the ADC, and we can observe how the quantization noise is shaped. I think it's a third order NTF with a zero at zero frequency and a complex conjugate pole at 8 MHzish.



12.6.2.4 Complex Sigma-Delta

There are cool sigma-delta modulators with crazy configurations and that may look like an exercise in "Let's make something complex", however, most of them have a reasonable application. One example is the one below for radio recievers

A 56 mW Continuous-Time Quadrature Cascaded Sigma-Delta Modulator With 77 dB DR in a Near Zero-IF 20 MHz Band



igma-delta modulator design.

12.6.2.5 My first Sigma-Delta

The first sigma-delta modulator I made in "real-life" was similar to the one shown below.

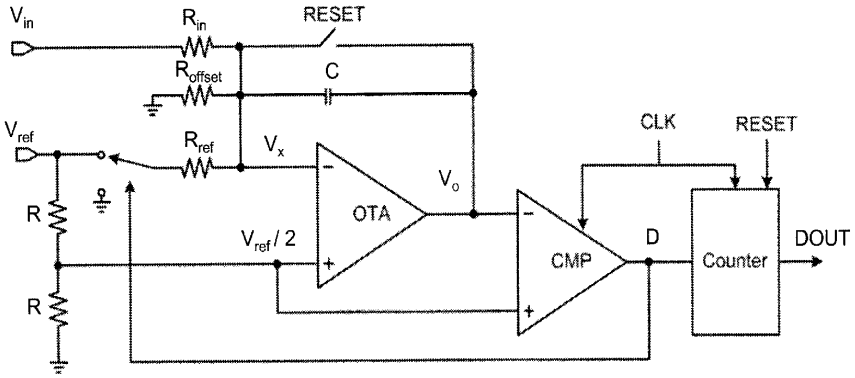
The input voltage is translated into a current, and the current is integrated on capacitor C. The R_{offset} is to change the mid-level voltage, while R_{ref} is the 1-bit feedback DAC. The comparator is the quantizer. When the clock strikes the comparator compares the V_o and $V_{ref}/2$ and outputs a 1-bit digital output D

The complete ADC is operated in a "incremental mode", which is a fancy way of saying

> Reset your sigma-delta modulator, run the sigma delta modulator for a fixed number of cycles (i.e 1024), and count the number of ones at D

The effect of an "incremental mode" is to combine the modulator and a output filter so the ADC appears to be a slow Nyquist ADC.

For more information, ask me, or see the patent at Analogue-to-digital converter



12.7 Want to learn more?

The design of sigma-delta modulation analog-to-digital converters

Delta-sigma modulation in fractional-N frequency synthesis

A CMOS Temperature Sensor With a Voltage-Calibrated Inaccuracy of ± 0.15 C (3sigma) From -55 Cto 125 C

A 20-mW 640-MHz CMOS Continuous-Time Sigma-Delta ADC With 20-MHz Signal Bandwidth, 80-dB Dynamic Range and 12-bit ENOB

A Micro-Power Two-Step Incremental Analog-to-Digital Converter

Voltage Regulation 13

Keywords: Battery, Vreg, LDOP, LDON, Flipped voltage follower, Buck, Boost, Load, Line, PSRR, MAX C, Quiescent, Settling, Efficiency, PWM, PFM

13.1 Voltage source

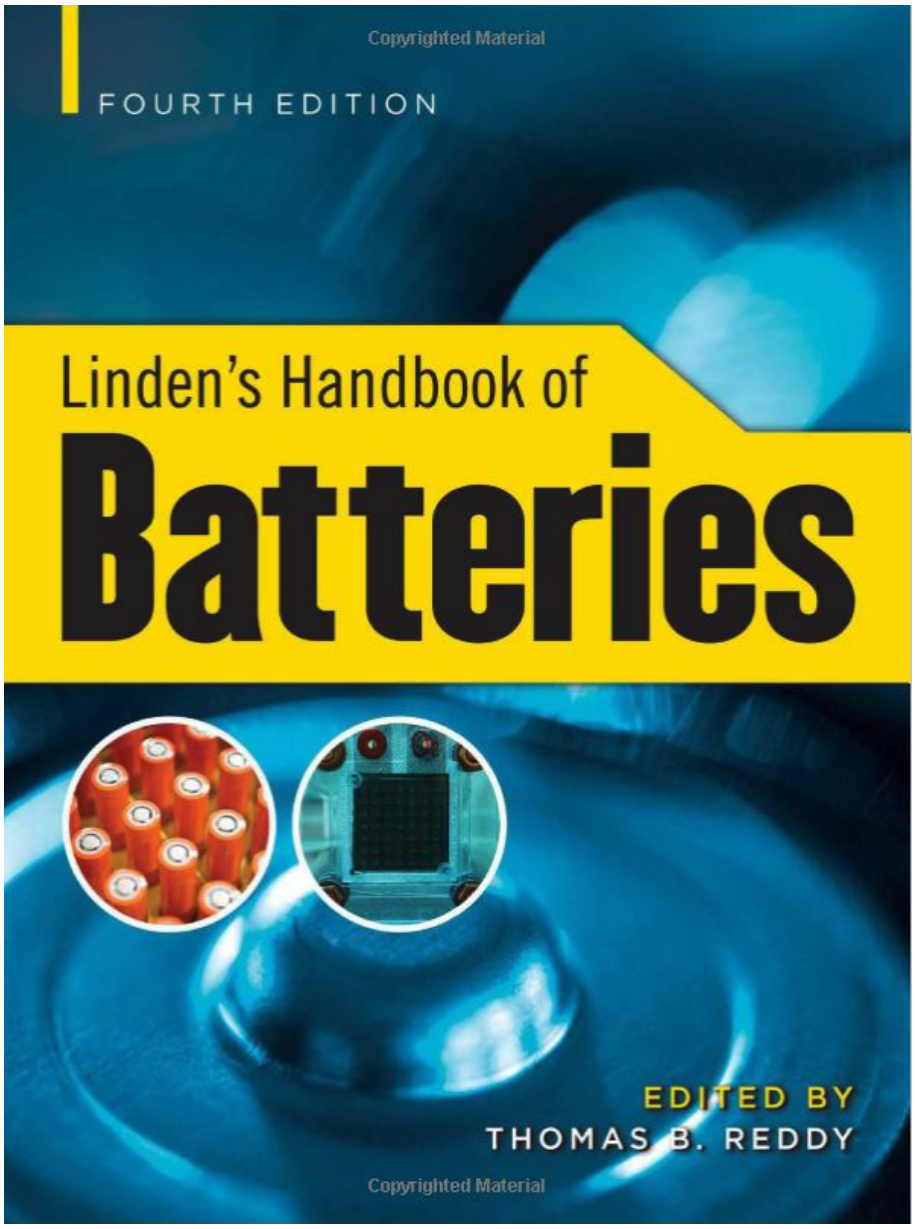
Most, if not all, integrated circuits need a supply and ground to work.

Assume a system is AC powered. Then there will be switched regulator to turn wall AC into DC. The DC might be 48 V, 24 V, 12 V, 5 V, 3 V 1.8 V, 1.0 V, 0.8 V, or who knows. The voltage depends on the type of IC and the application.

Many ICs are battery operated, whether it's your phone, watch, heart rate monitor, mouse, keyboard, game controller or car.

For batteries the voltage is determined by the difference in Fermi level on the two electrodes, and the Fermi level (chemical potential) is a function of the battery chemistry. As a result, we need to know the battery chemistry in order to know the voltage.

Linden's Handbook of Batteries is a good book if you want to dive deep into primary (non-chargeable) or secondary (chargeable) batteries and their voltage curves.



Some common voltage sources are listed below.

	Chemistry	Voltage [V]
Primary Cell	LiFeS2 , Zn/Alk/MnO2 , LiMnO2	0.8 - 3.6
Secondary Cell	Li-Ion	2.5 - 4.3
USB	-	4.0 - 6.5 (20)

The battery determines the voltage of the "electron source", however, can't we just run everything directly off the battery? Why do we need DC to DC converters or voltage regulators?

Turns out, transistors can die.

Today's transistor, as shown below, are a complicated three dimensional structure. Dimensions are measured in nano-meter, which

makes the transistors fragile.

In Analog Circuit Design in Nanoscale CMOS Technologies Lanny explains how to design around some of the breakdown effects.

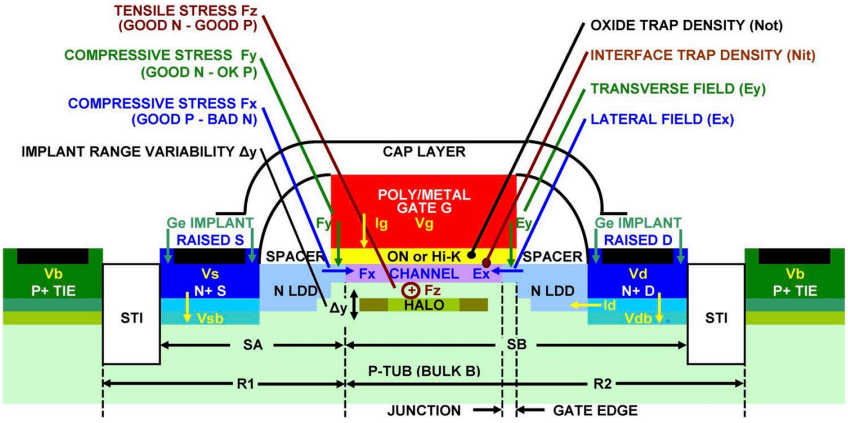


Fig. 2. *NMOS cross-section. In addition to stress from cap layers and Ge raised source-drain (S-D) implants, device dimensions such as distance from source-channel boundary to nearby STI (SA and SB), proximity and regularity of overlying metal patterns, and short distances to other device patterns within the local (< 2 μm) stress field induce transverse (F_y) and lateral (F_x and F_z) stress components, which affect threshold and mobility. Increasing the distance to P+ ties increases local tub (bulk) resistance components R1 and R2, which isolate the device MOS model substrate node from the device subcircuit symbol V_b node and degrade HF performance. Hot carrier reliability stress is dependent on the sum of transverse and lateral fields E_y and E_x. These fields are increased near the drain by increasing source to bulk (V_{sb}) and drain (V_d) to gate (V_g) or source (V_s) voltages in various combinations. As hot carrier stress increases, damage to channel from interface trap density (N_{it}) affects threshold and mobility, while gate oxynitirde (ON) or high-dielectric-constant (Hi-K) insulator trap density (N_{ot}) affects threshold and gate leakage.*

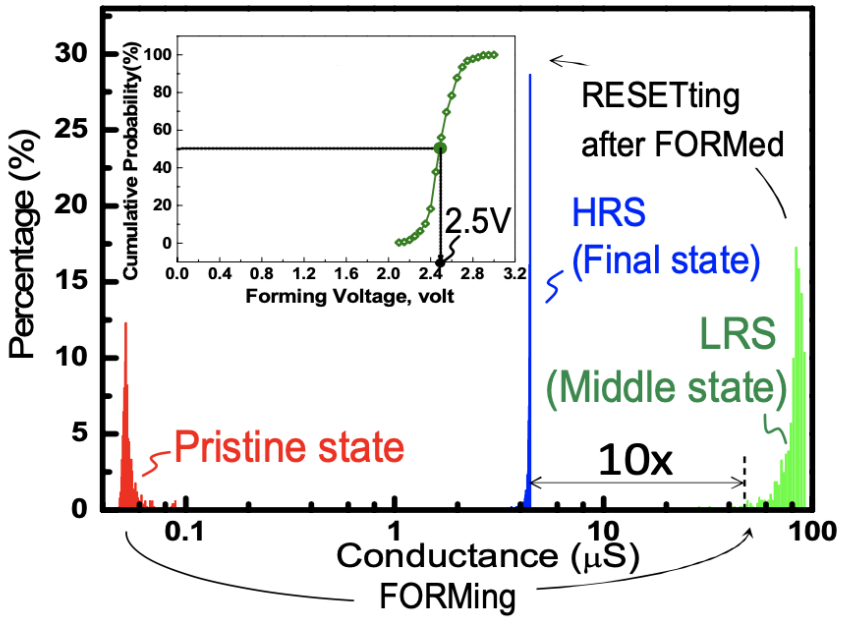
The transistors in a particular technology (from GlobalFoundries, TSMC, Samsung or others) have a maximum voltage that they can survive for a certain time. Exceed that time, or voltage, and the transistors die.

13.1.0.1 Why transistors die

A gate oxide will break due to Time Dependent Dielectric Breakdown (TDDB) if the voltage across the gate oxide is too large. Silicon oxide can break down at approximately 5 MV/cm. The breakdown forms a conductive channel from the gate to the channel and is permanent. After breakdown there will be a resistor of kOhms between gate and channel.

A similar breakdown phenomena is used in Metal-Oxide RRAM and the SkyWater ReRAM

Below is an example of ReRAM. In the Pristine state the conductance is low, resistance is in the hundreds of mega Ohm. In a transistor we want the oxide to stay high resistive. In ReRAM, however, we apply a high voltage across the oxide, which forms a conductive channel across the oxide. Turns out, that the conductive channel can be flipped back and forth between a high resistive state, and a low resistive state to store a 1 or a 0 in a non-volatile manner.



The threshold voltage of a transistor can shift excessively over time caused by Hot-Carrier Injection (HCI) or Negative Bias Temperature Instability.

Hot-Carrier injection is caused by electrons, or holes, accelerated to high velocity in the channel, or drain depletion region , causing impact ionization (breaking a co-valent bond releasing an electron/hole pair). At a high drain/source field, and
medium gate/(source or drain) field, the channel minority carriers can be accelerated to high energy and transition to traps in the oxide, shifting the threshold voltage.

Negative Bias Temperature Instability is a shift in threshold voltage due to a physical change in the oxide. A strong electric field across the oxide for a long time can break co-valent, or ionic bonds, in the oxide. The bond break will change the forces (stress) in the amorphous silicon oxide which might not recover. As such, there might be more traps (states) than before. See Simultaneous Extraction of Recoverable and Permanent Components Contributing to Bias-Temperature Instability for more details.

For a long time, I had trouble with “traps in the oxide” “. I had a hard time visualizing how electrons wandered down the channel and got caught in the oxide. I was trying to imagine the electric field, and that the electron needed to find a positive charge in the oxide to cancel. Diving a bit deeper into quantum mechanics, my mental image improved a bit, so I’ll try to give you a more accurate mental model for how to think about traps.

Quantum mechanics tells us that bound electrons can only occupy fixed states. The probability of finding an electron in a state is given

by the Fermi function, but if there is no energy state at a point in space, there cannot be an electron there.

For example, there might be a 50 % probability of finding an electron in the oxide, but if there is no state there, then there will not be any electron , and thus no change to the threshold voltage.

What happens when we make "traps", through TDDB, HCI, or NBTI is that we create new states that can potentially be occupied by electrons. For example one, or more, broken silicon co-valent bonds and a dislocation of the crystal lattice.

If the Fermi-Dirac statistics tells us the probability of an electron being in those new states is 50 %, then there will likely be electrons there.

The threshold voltage is defined as the voltage at which we can invert the channel, or create the same density of electrons in the channel (for NMOS) as density of dopant atoms (density of holes) in the bulk.

If the oxide has a net negative charge (because of electrons in new states), then we have to pull harder (higher gate voltage) to establish the channel. As a result, the threshold voltage increases with electrons stuck in the oxide.

In quantum mechanics the time evolution, and the complex probability amplitude of an electron changing state, could, in theory, be computed with the Schrodinger equation. Unfortunately, for any real scenario, like the gate oxide of a transistor, using Schrodinger to compute exactly what will happen is beyond the capability of the largest supercomputers.

13.1.1 Core voltage

The voltage where the transistor can survive is estimated by the foundry, by approximation, and testing, and may be like the table below.

Node [nm]	Voltage [V]
180	1.8
130	1.5
55	1.2
22	0.8

13.1.2 IO voltage

Most ICs talk to other ICs, and they have a voltage for the general purpose input/output. The voltage reduction in I/O voltage does not need to scale as fast as the core voltage, because foundries have thicker oxide transistors that can survive the voltage.

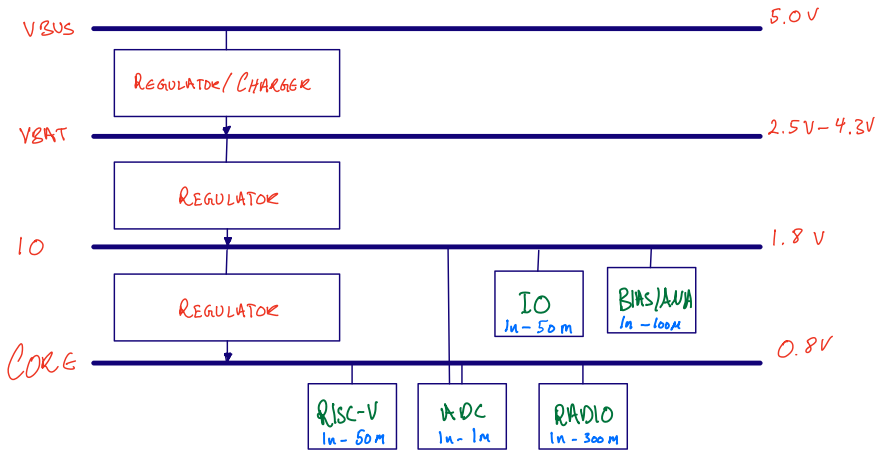
Voltage [V]
5.0
3.0
1.8
1.2

13.1.3 Supply planning

For any IC, we must know the application. We must know where the voltage comes from, the IO voltage, the core voltage, and any other requirements (like charging batteries).

One example could be an IC that is powered from a Li-Ion battery, with a USB to provide charging capability.

Between each voltage we need an analog block, a regulator, to reduce the voltage in an effective manner. What type of regulator depends again on the application, but the architecture of the analog design would be either a linear regulator, or a switched regulator.



The dynamic range of the power consumed by an IC can be large. From nA when it's not doing anything, to hundreds of mA when there is high computation load.

As a result, it's not necessarily possible, or effective, to have one regulator from 1.8 V to 0.8 V. We may need multiple regulators.

Some that can handle low load (nA - μA) effectively, and some that can handle high loads.

For example, if you design a regulator to deliver 500 mA to the load, and the regulator uses 5 mA, that's only 1 % of the current, which may be OK. The same regulator might consume 5 mA even though the load is 1 uA, which would be bad. All the current flows in the regulator at low loads.

Name	Voltage	Min [nA]	Max [mA]	PWR DR [dB]
VDD_VBUS	5	10	500	77
VDD_VBAT	4	10	400	76
VDD_IO	1.8	10	50	67
VDD_CORE	0.8	10	350	75

Most product specifications will give you a view into what type of regulators there are on an IC. The picture below is from nRF5340 (page 23)

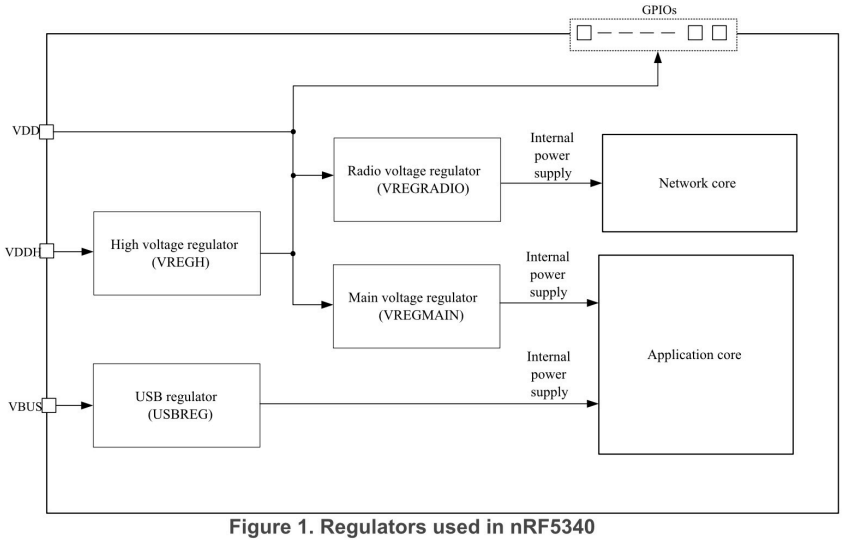
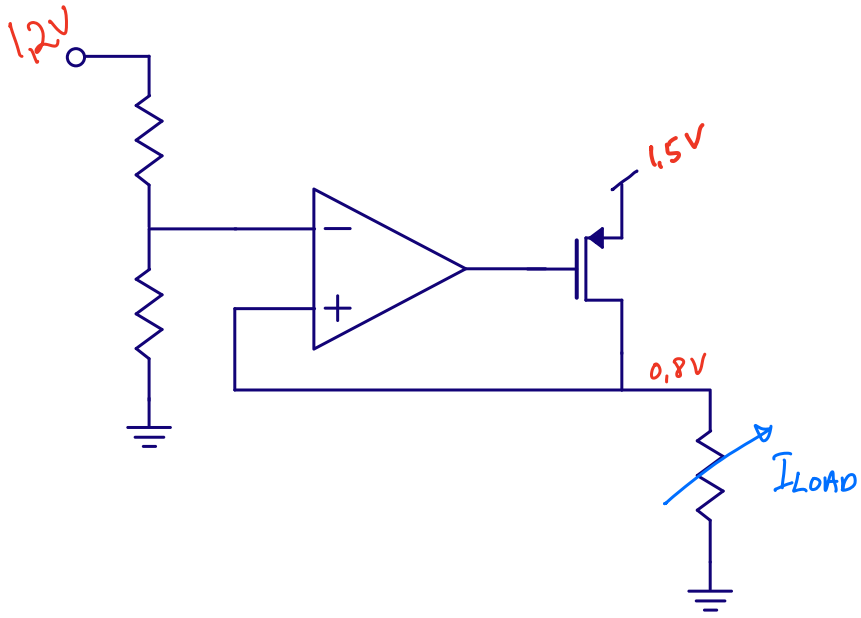


Figure 1. Regulators used in nRF5340

13.2 Linear Regulators

13.2.1 PMOS pass-fet

One way to make a regulator is to control the current in a PMOS with a feedback loop, as shown below. The OTA continuously adjusts the gate-source voltage of the PMOS to force the input voltages of the OTA to be equal.



For digital loads, where I_{load} is a digital current, with high current every rising edge of the clock, it's an option to place a large external decoupling capacitor (a reservoir of charge) in parallel with the load. Accordingly, the OTA would supply the average current.

The device between supply (1.5 V) and output voltage (0.8 V) is often called a pass-fet. A PMOS pass-fet regulator is often called a LDO, or low dropout regulator, since we only need a V_{DSSAT} across the PMOS, which can be a few hundred mV.

Key parameters of regulators are

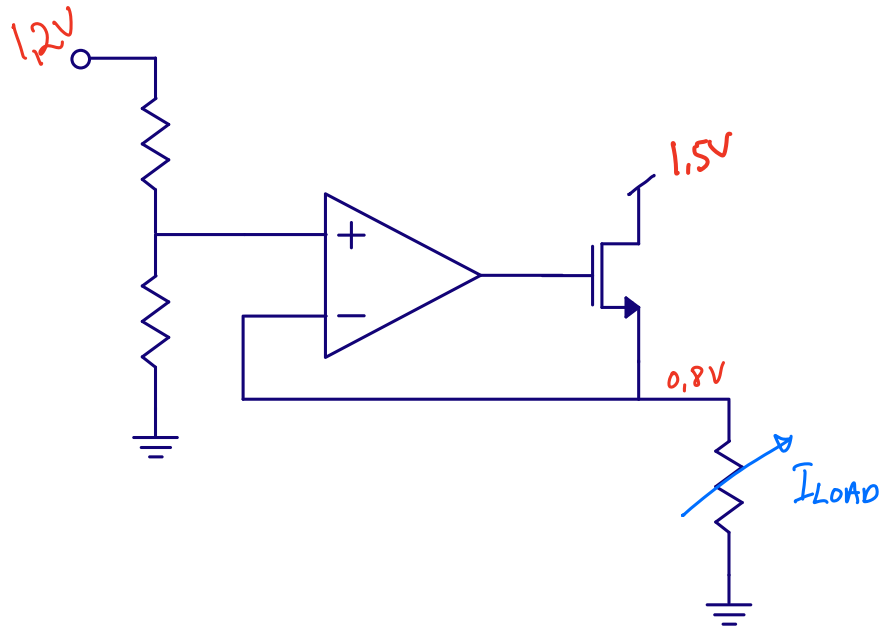
Parameter	Description	Unit
Load regulation	How much does the output voltage change with load current	V/A
Line regulation	How much does the output voltage change with input voltage	V/V
Power supply rejection ratio	What is the transfer function from input voltage to output voltage? The PSRR at DC is the line regulation	dB
Max current	How much current can be delivered through the pass-fet?	A
Quiescent current	What is the current used by the regulator	A
Settling time	How fast does the output voltage settle at a current step	s

A disadvantage of a PMOS is the hole mobility, which is lower than for NMOS. If the maximum current of an LDO is large, then the PMOS can be big. Maybe even 50 % of the IC area.

13.2.2 NMOS pass-fet

An NMOS pass-fet will be smaller than a PMOS for large loads. The disadvantage with an NMOS is the gate-source voltage needed. For some scenarios the needed gate voltage might exceed the input voltage (1.5 V). A gate voltage above input voltage is possible, but increases complexity, as a charge pump (switched capacitor regulator) is needed to make the gate voltage.

Another interesting phenomena with NMOS pass-fet is that the PSRR is usually better, but we do have a common gate amplifier, as such, high frequency voltage ripple on output voltage will be amplified to the input voltage, and may cause issues for others using the input voltage.



13.2.3 Control of pass-fet

The large dynamic range in power management systems can make it challenging to have a single pass-fet.

The size of the pass-fet is set by the maximum Vgs, and the current that needs to be delivered.

Assume we need 500 mA from the LDO. If we assume that the maximum Vgs is 1.5 V, then we can simulate to try and find a size.

I've made a testbench at

Testbench for LDO pass-fet

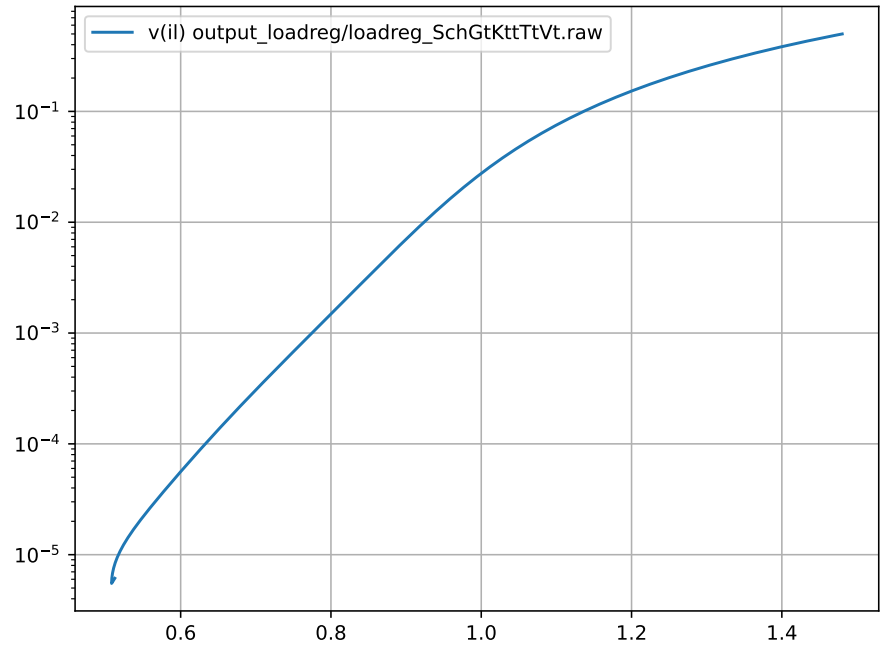
Below is an excerpt from the testbench. The pass-fet size has been determined by iteration.

The OTA in the LDO is modeled by the B source. Notice the use of the tanh function in order to keep the G voltage within the rails.

```
* Pass-fet
XM1 OUT G VDD VDD sky130_fd_pr__pfet_01v8 L=0.252 W=11.52 nf=2 ...

* Reference
VREF VREF 0 dc 0.8

* OTA
BOTA G 0 V=(1 + tanh(-1000*(v(vref) -v(out) )))/2*{AVDD}

* Load cap
CL OUT 0 1u

* Current load
ILOAD OUT 0 pwl 0 0 1u 0 50u 0.5
```

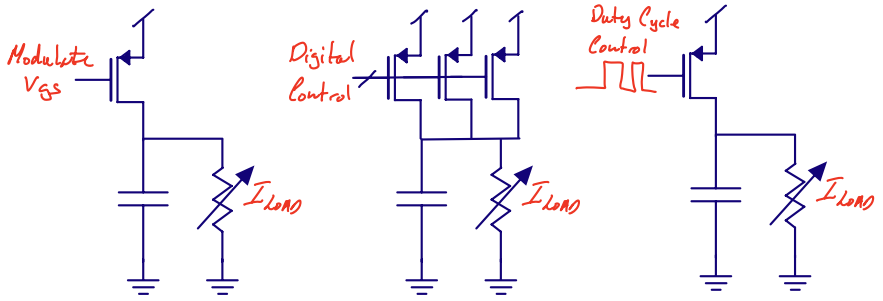
Below is a plot of the current on the y-axis as a function of the Vgs on the x-axis. Although it's possible to have almost 6 orders of magnitude change in current in the transistor it does become hard to make the loop stable over such a large range.

Sometimes it's easier to split the range into multiple ranges.



As such, there are multiple control options for the pass-fet. Below is a summary of a few methods.

We can control the Vgs, or we can switch the number of instances, or we can turn the pass-fet on and off dynamically. What we choose will depend on the application.



13.3 Switched Regulators

Linear regulator have poor power efficiency. Linear regulators have the same current in the load, as from the input.

For some applications a poor efficiency might be OK, but for most battery operated systems we're interested in using the electrons from the battery in the most effective manner.

Another challenge is temperature. A linear regulator with a 5 V input voltage, and 1 V output voltage will have a maximum power

efficiency of 20 % (1/5). 80 % of the power is wasted in the pass-fet as heat.

Imagine a LDO driving an 80 W CPU at 1 V from a 5 V power supply. The power drawn from the 5 V supply is 400 W, as such, 320 W would be wasted in the LDO. A quad flat no-leads (QFN) package usually have a thermal resistance of 20 °C/W, so if it would be possible, the temperature of the LDO would be 6400 °C. Obviously, that cannot work.

For increased power efficiency, we must use switched regulators.

Imagine a switched regulator with 93 % power efficiency. The power from the 5 V supply would be $80\,\mathrm{W}/0.93 = 86\,\mathrm{W}$, as such, only 6 W is wasted as heat. A temperature increase of $6\,\mathrm{W} \times 20\,^{\circ}\mathrm{C/W} = 120^{\circ}\mathrm{C}$ is still high, but not impossible with a small heat-sink.

All switched regulators are based on devices that store electric field (capacitors), or magnetic field (inductors).

13.3.1 Principles of switched regulators

> There is a big difference between the idea for a circuit, and the actual implementation. A real DC/DC implementation may seem overwhelming.

Just look at figure 7 in A 10-MHz 2–800-mA 0.5–1.5-V 90% Peak Efficiency Time-Based Buck Converter With Seamless Transition Between PWM/PFM Modes

So before we go into details, let's have a look at the principles.

13.3.1.1 Inductive BUCK DC/DC

Below is a common illustration of a inductive DC/DC to step down the voltage.

Imagine Vout is at our desired output voltage, for example 0.8 V. Assume Vin is 1.8 V.

When we close the switch, the inductor will begin to integrate the voltage across the inductor, and the current from Vin to Vout increases.

When we turn off the switch, the inductor current will not stop immediately, it cannot, that's what

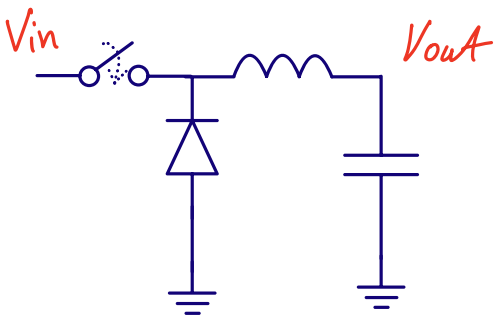
$$V = L\frac{dI}{dt}$$

tells us. As a result, the current continues, but now the current is pulled from ground through the diode.

Since we're pulling current from ground, it should be intuitive that the current from Vin is less than the load current at Vout, assuming Vin > Vout.

The output voltage can be controlled by how long we turn on the switch. Each time we turn on the switch the inductor will inject a charge packet into the load capacitance.

If we have a control loop on the output voltage, then we can get an output voltage that is independent of the input voltage.



13.3.1.2 Capacitive BUCK DC/DC

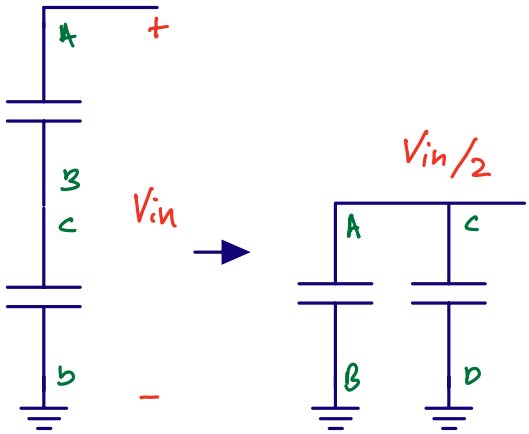
In a capacitive buck below what we're doing is charging two capacitors in series to a high voltage, Vin, and then re-configuring the capacitors to be in parallel.

If the capacitors are the same size, then the output voltage would be half the input voltage.

To re-configure the circuit we'd use switches.

A disadvantage with capacitive bucks is that the output voltage is always a factor of the input voltage. When the input voltage changes, the output voltages changes proportionally.

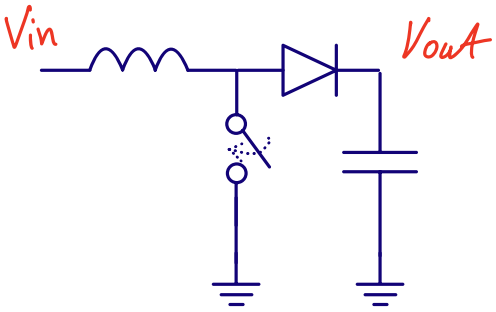
Often we have to insert an LDO after a capacitive buck to make the output voltage independent of input voltage.



13.3.1.3 Inductive BOOST DC/DC

Consider the circuit below. Here we setup a current from Vin to ground when the switch is on. When the switch is off push the current through the diode, and thus, the Vout can be higher than Vin.

In a similar manner to the Buck, the output voltage will be impacted by how long we turn on the switch for.

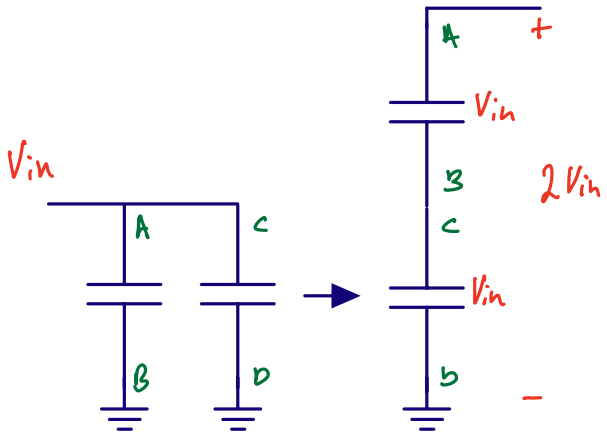


13.3.1.4 Capacitive BOOST DC/DC

In a capacitive boost we start with a parallel connection, charge the capacitors to Vin, then reconfigure the circuit to a series combination.

As such, the output voltage would be two times the input voltage, assuming the capacitors are equal.

The configuration below is quite often called a “Charge pump”, and can be configured to generate both positive, or negative voltages.



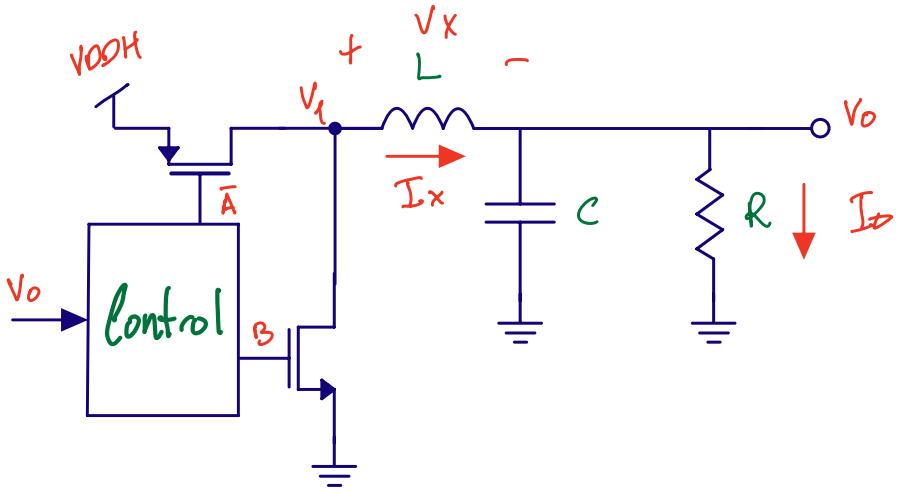
13.3.2 Inductive DC/DC converter details

I've found that people struggle with inductive DC/DCs. They see a circuit inductors, capacitors, and transistors and think filters, Laplace and steady state. The path of Laplace and steady state will lead you astray and you won't understand how it works.

Hopefully I can put you on the right path to understanding.

In the figure below we can see a typical inductive switch mode DC/DC converter. The input voltage is V_{DDH}, and the output is V_O.

Most DC/DCs are feedback systems, so the control will be adjusted to force the output to be what is wanted, however, let's ignore closed loop for now.



To see what happens I find the best path to understanding is to look at the integral equations.

The current in the inductor is given by

$$I_x(t) = \frac{1}{L}\int V_x(t)dt$$

and the voltage on the capacitor is given by

$$V_o(t) = \frac{1}{C}\int (I_x(t) - I_o(t))dt$$

Before you dive into Matlab, Mathcad, Maple, SymPy or another of your favorite math software, it helps to think a bit.

My mathematics is not great, but I don't think there is any closed form solution to the output voltage of the DC/DC, especially since the state of the NMOS and PMOS is time-dependent.

The output voltage also affect the voltage across the inductor, which affects the current, which affects the output voltage, etc, etc.

The equations can be solved numerically, but a numerical solution to the above integrals needs initial conditions.

There are many versions of the control block, let's look at two.

13.3.3 Pulse width modulation (PWM)

Assume $I_x = 0$ and $I_o = 0$ at $t = 0$. Assume the output voltage is $V_O = 0$. Imagine we set $A = 1$ for a fixed time duration. The voltage at $V_1 = V_{DDH}$, and $V_x = V_{VDDH} - V_O$. As V_x is positive, and roughly constant, the current I_x would increase linearly, as given by the equation of the current above.

Since the I_x is linear, then the increase in V_o would be a second order, as given by the equation of the output voltage above.

Let's set $A = 0$ and $B = 1$ for fixed time duration (it does not need to be the same as duration as we set $A = 1$). The voltage across the inductor would be $V_x = 0 - V_o$. The output voltage would not have increased much, so the absolute value of V_x during $A = 1$ would be higher than the absolute value of V_x during the first $B = 1$.

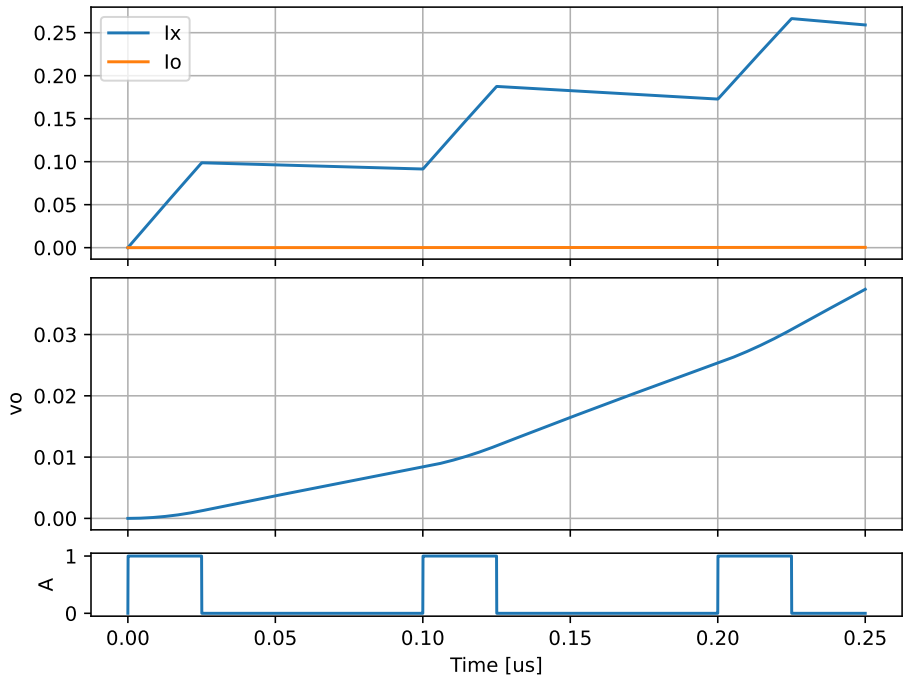
The V_x is now negative, so the current will decrease, however, since V_x is small, it does not decrease much.

I've made a

Jupyter PWM BUCK model

that numerically solves the equations.

In the figure below we can see how the current during A increases fast, while during B it decreases little. The output voltage increases similarly to a second order function.

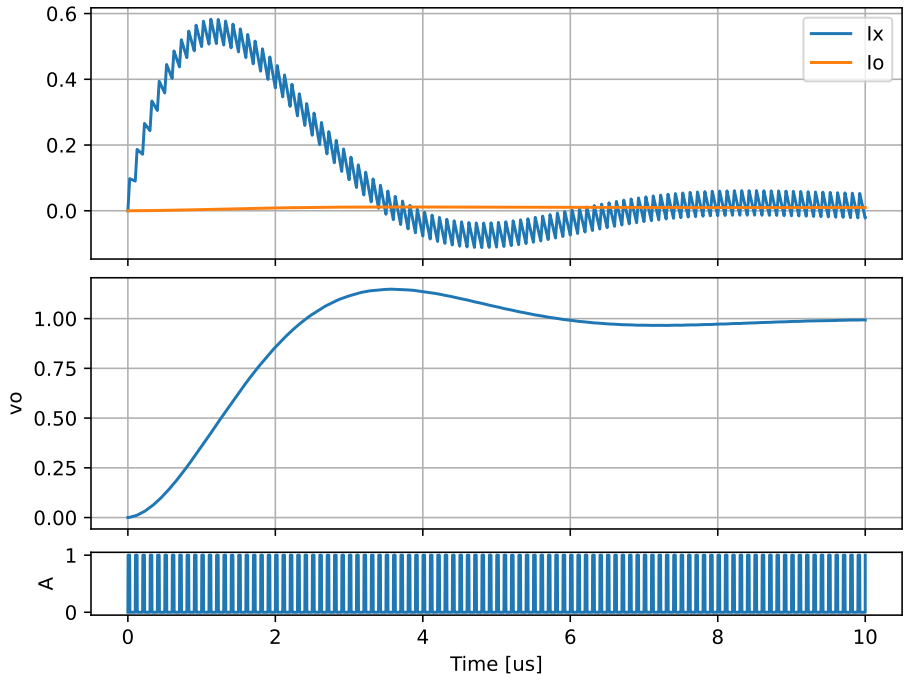


If we run the simulation longer, see plot below, the DC/DC will start to settle into a steady state condition.

On the top we can see the current I_x and I_o, the second plot you can see the output voltage. Turns out that the output voltage will be

$$V_o = V_{in} \times \text{Duty-Cycle}$$

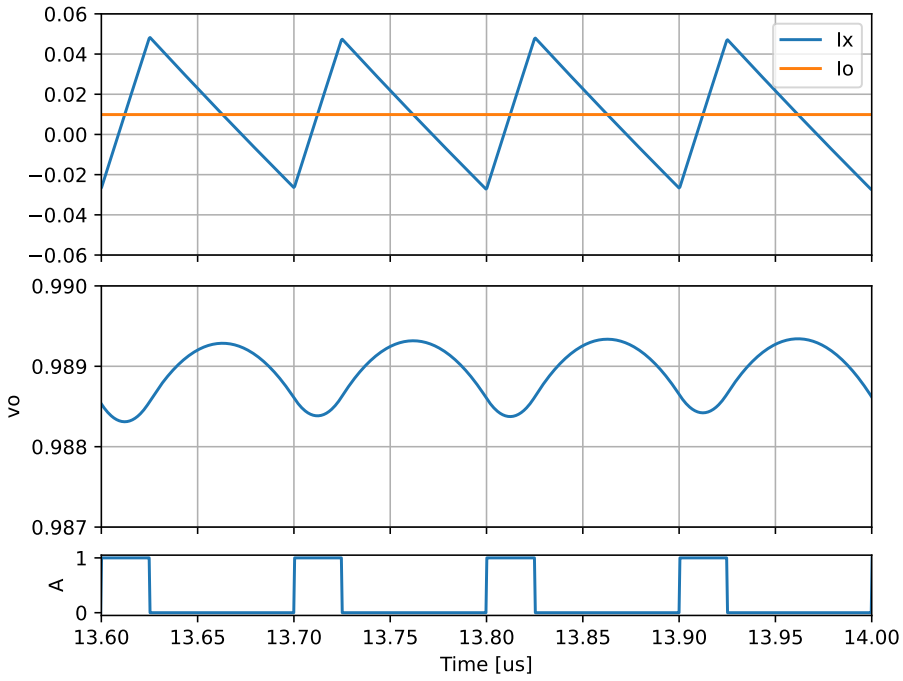
, where the duty-cycle is the ratio between the duration of $A = 1$ and $B = 1$.



Once the system has fully settled, see figure below, we can see the reason for why DC/DC converters are useful.

During $A = 1$ the current I_x increases fast, and it's only during $A = 1$ we pull current from V_{DDH}. At the start of $A = 0$ the current is still positive, which means we pull current from ground. The average current in the inductor is the same as the average current in the load, however, the current from V_{DDH} is lower than the average inductor current, since some of the current comes from ground.

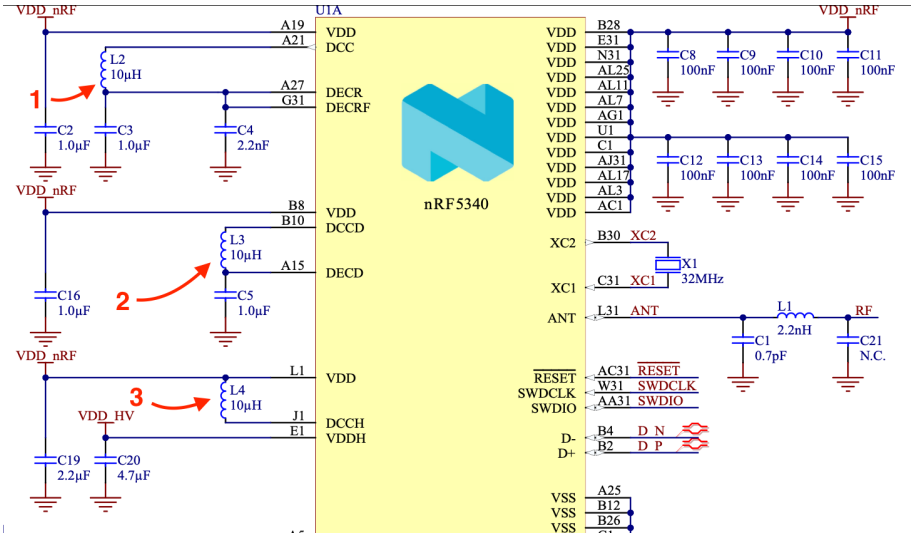
If the DC/DC was 100% efficient, then the current from the 4 V input supply would be 1/4'th of the 1 V output supply. 100% efficient DC/DC converters violate the laws of nature, as such, we can expect to get up to 9X% under optimal conditions.



13.3.4 Real world use

DC/DC converters are used when power efficiency is important. Below is a screenshot of the hardware description in the nRF5340 Product Specification.

We can see 3 inductor/capacitor pairs. One for the "VDDH", and two for "DECRF" and "DECD", as such, we can make a good guess there are three DC/DC converters inside the nRF5340.

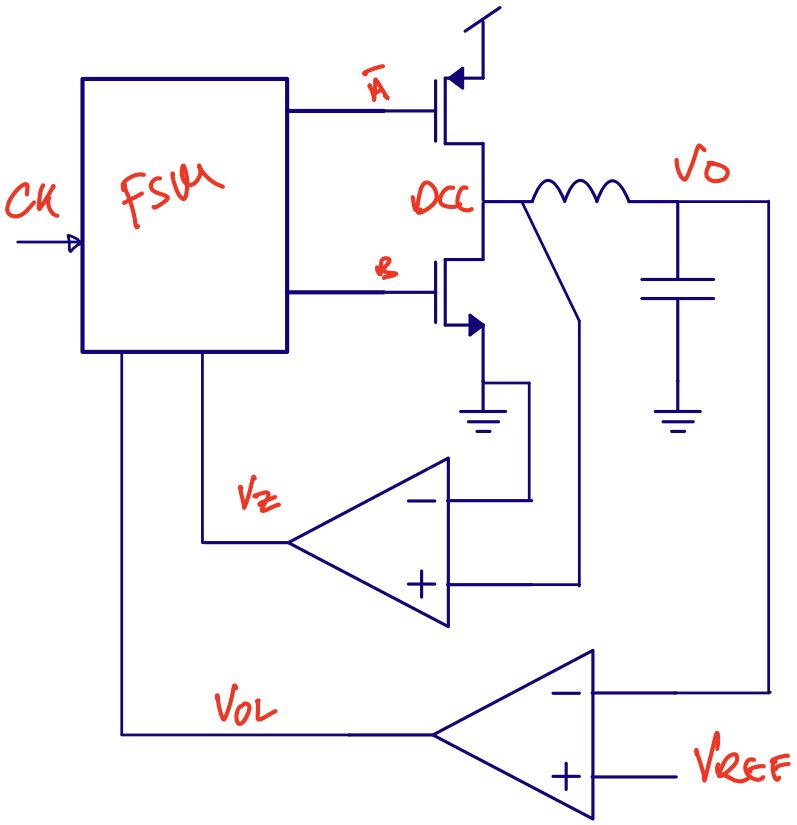


13.3.5 Pulsed Frequency Mode (PFM)

Power efficiency is key in DC/DC converters. For high loads, PWM, as explained above, is usually the most efficient and practical. For lighter loads, other configurations can be more efficient.

In PWM we continuously switch the NMOS and PMOS, as such, the parasitic capacitance on the V_1 node is charged and discharged, consuming power. If the load is close to 0 A, then the parasitic load's can be significant.

In pulsed-frequency mode we switch the NMOS and PMOS when it's needed. If there is no load, there is no switching, and V_1 or DCC in figure below is high impedant.



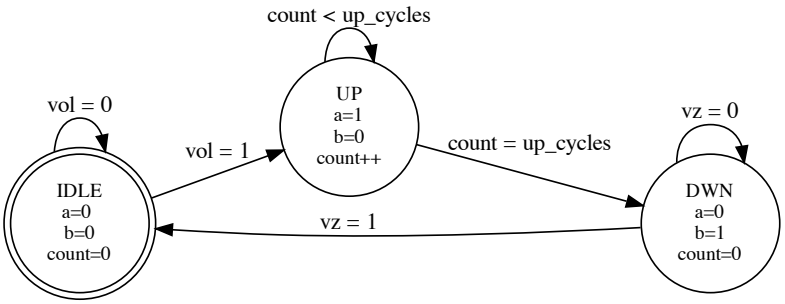
Imagine V_o is at 1 V, and we apply a constant output load. According to the integral equations the V_o would decrease linearly.

In the figure above we observe V_o with a comparator that sets V_{OL} high if the $V_o < V_{REF}$. The output from the comparator could be the inputs to a finite state machine (FSM).

Consider the FSM below. On $vol = 1$ we transition to "UP" state where turn on the PMOS for a fixed number of clock cycles. The inductor current would increase linearly. From the "UP" state we go to the "DOWN" state, where we turn on the NMOS. The inductor current would decrease roughly linearly.

The "zero-cross" comparator observes the voltage across the NMOS drain/source. As soon as we turn the NMOS on the current direction in the inductor is still from DCC to V_o. Since the current is pulled from ground, the DCC must be below ground. As the current in the inductor decreases, the voltage across the NMOS will at some point be equal to zero, at which point the inductor current is zero.

When $vz = 1$ happens in the state diagram, or the zero cross comparator triggers, we transition from the "DWN" state back to "IDLE". Now the FSM wait for the next time $V_o < V_{REF}$.

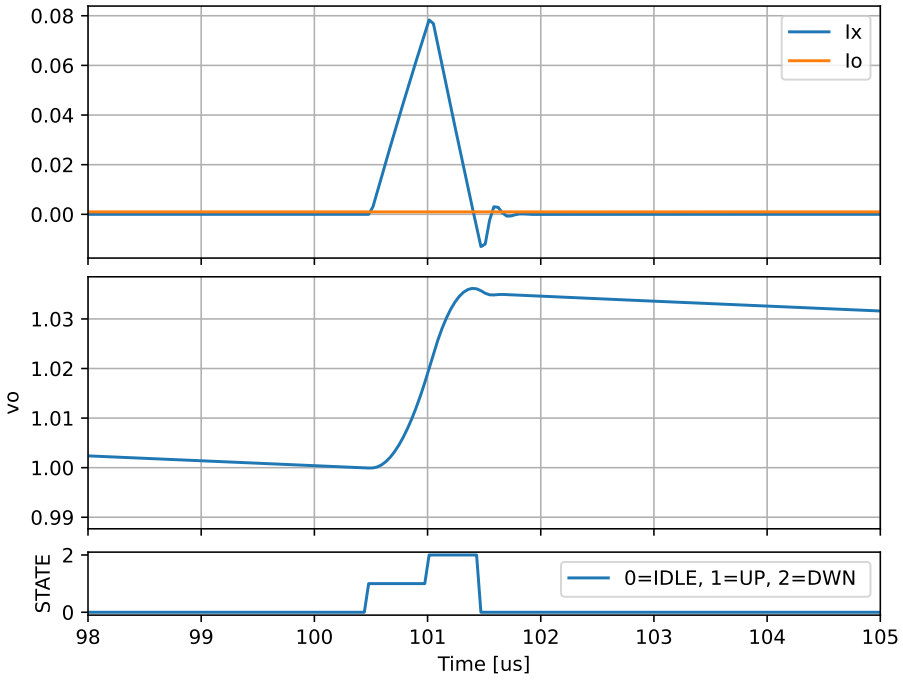


I think the name "pulsed-frequency mode" refers to the fact that the frequency changes according to load current, however, I'm not sure of the origin of the name. The name is not important. What's important is that you understand that mode 1 (PWM) and mode 2 (PFM) are two different "operation modes" of a DC/DC converter.

I made a jupyter model for the PFM mode. I would encourage you to play with them.

Below you can see a period of the PFM buck. The state can be seen in the bottom plot, the voltage in the middle and the current in the inductor and load in the top plot.

Jupyter PFM BUCK model



13.4 Want to learn more?

Search terms: regulator, buck converter, dc/dc converter, boost converter

13.4.1 Linear regulators

A Scalable High-Current High-Accuracy Dual-Loop Four-Phase Switching LDO for Microprocessors Overview of fancy LDO schemes, digital as well as analog

Development of Single-Transistor-Control LDO Based on Flipped Voltage Follower for SoC In capacitor less LDOs a flipped voltage follower is a common circuit, worth a read.

A 200-mA Digital Low Drop-Out Regulator With Coarse-Fine Dual Loop in Mobile Application Processor Some insights into large power systems.

13.4.2 DC-DC converters

Design Techniques for Fully Integrated Switched-Capacitor DC-DC Converters Goes through design of SC DC-DC converters. Good place to start to learn the trade-offs, and the circuits.

High Frequency Buck Converter Design Using Time-Based Control Techniques I love papers that challenge "this is the way". Why should we design analog feedback loops for our bucks, why not design digital feedback loops?

Single-Inductor Multi-Output (SIMO) DC-DC Converters With High Light-Load Efficiency and Minimized Cross-Regulation for Portable Devices Maybe you have many supplies you want to drive, but you don't want to have many inductors. SIMO is then an option

A 10-MHz 2–800-mA 0.5–1.5-V 90% Peak Efficiency Time-Based Buck Converter With Seamless Transition Between PWM/PFM Modes Has some lovely illustrations of PFM and PWM and the trade-offs between those two modes.

A monolithic current-mode CMOS DC-DC converter with on-chip current-sensing technique In bucks converters there are two "religious" camps. One hail to "voltage mode" control loop, another hail to "current mode" control loops. It's good to read about both and make up your own mind.

Clocks and PLLs 14

15 Oscillators

Low Power Radio 16

17 Analog SystemVerilog

18 Energy Sources

Bibliography