IN3170/4170, Spring 2025

Philipp Häfliger
hafliger@ifi.uio.no

What to expect

Content

Why Transistor Level Design of Integrated Circuits?

Why Transistor Level Digital?

Why Analog?

Course Goal

Course Organization

Content

Why Transistor Level Design of Integrated Circuits?

Why Transistor Level Digital?

Why Analog?

Course Goal

Course Organization

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

- Costly (development, design iteration time, production)
- Inflexible and low level of reusability

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

- Costly (development, design iteration time, production)
- Inflexible and low level of reusability

Alternatives?

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

- Costly (development, design iteration time, production)
- Inflexible and low level of reusability

Alternatives?

Recombine higher level blocks:

- Embedded Systems
- Configure or program an existing GP IC, e.g. FPGA, CPU, microcontroller, SoC (originally pure digital, now some with mixed signal options)

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

- Costly (development, design iteration time, production)
- Inflexible and low level of reusability

Alternatives?

Recombine higher level blocks:

- Embedded Systems
- Configure or program an existing GP IC, e.g. FPGA, CPU, microcontroller, SoC (originally pure digital, now some with mixed signal options)

So why bother?

Why do a transistor level GP or AS IC design?

Abbreviations: general purpose (GP), application specific (AS), integrated circuit (IC)

Well, why not?

- Costly (development, design iteration time, production)
- Inflexible and low level of reusability

Alternatives?

Recombine higher level blocks:

- Embedded Systems
- Configure or program an existing GP IC, e.g. FPGA, CPU, microcontroller, SoC (originally pure digital, now some with mixed signal options)

So why bother?

- Ultimate performance (speed, power)
- Ultimate miniaturization
- Reliability (fewer points of failure)
- Very cheap for high volume production (e.g. CPUs)
- For specific applications (Mixed-Signal) Systems-on-Chip (SoC)

Content

Why Transistor Level Design of Integrated Circuits?

Why Transistor Level Digital?

Why Analog?

Course Goal

Course Organization

Why do a Digital transistor level/full custom design?

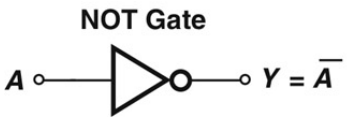
Why do a Digital transistor level/full custom design?

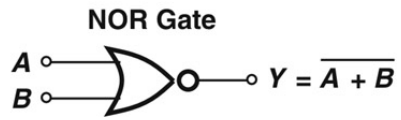
- See previous arguments for and against ASIC
- High development cost worthwhile for high volume production of GP ICs, e.g. CPU, FPGAs, GPUs, Microcontrollers etc. (mostly not 'full custom' design but automated 'synthesis')
- Ultimate performance in speed, miniaturization, power consumption. This requires real understanding on a single transistor level. Analogous in SW of where it's worth to program in Assembler for ultimate performance.

Even Computers are 'Analog' ;-)

Where the digital abstraction breaks down

- Gates

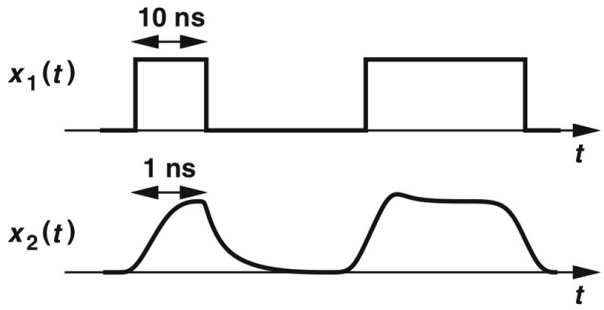




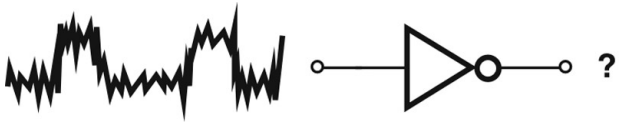
- Increasing speed
  - Why this degradation?
  - How do we improve performance?
  - Digital → analog
    - Going for speed…

100MHz

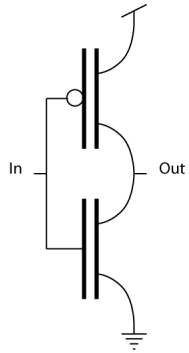
1GHz



- Noise/interference



  - Where does this noise originate?
  - How do we reduce this noise/interference?
  - Digital → analog
    - When scaling down size and scaling up complexity



Content

Why Transistor Level Design of Integrated Circuits?

Why Transistor Level Digital?

Why Analog?

Course Goal

Course Organization

The world is analog

Analog electronics for sensor/actuator interfaces



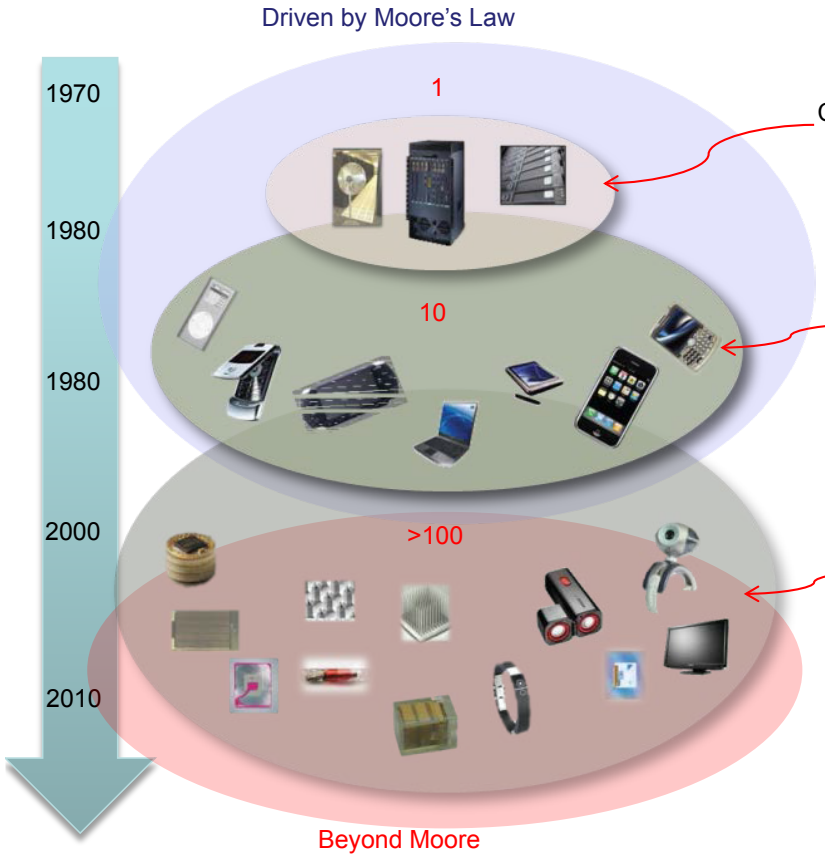
$\Leftrightarrow$

$\Leftrightarrow$ user $\Leftrightarrow$



Ubiquitous Sensors Interfaces

Trend to 'Cyberphysical Systems'



Computational Infrastructure
- Stationary/backend
- Wired
- High end computing

Mobile access devices
- Human interaction
- Portable
- Mostly wireless
- Battery

Sensory swarm
- Miniature
- Wireless
- Autonomous/self-contained
- Controlling and sensing natural processes

Transistor level tweaking essential in analog

While the same schematics in digital often behaves very/sufficiently similarly over a wide range of different CMOS technologies and transistor specifications, this is not the case for analog designs: The same schematics can result in quite different analog behaviour when implemented in different CMOS technologies or when transistor dimensions are changed.

Content

Why Transistor Level Design of Integrated Circuits?

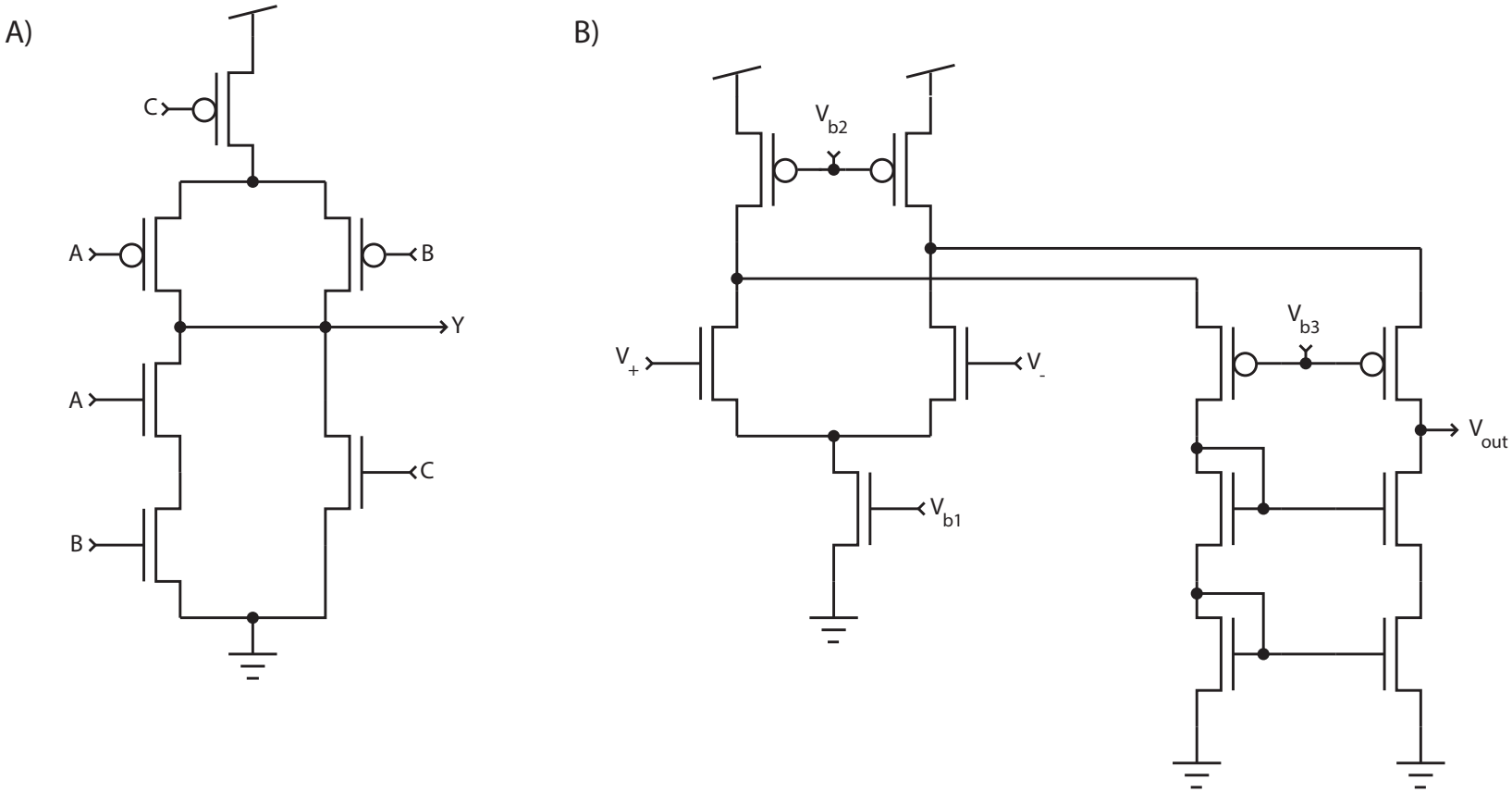
Why Transistor Level Digital?

Why Analog?

Course Goal

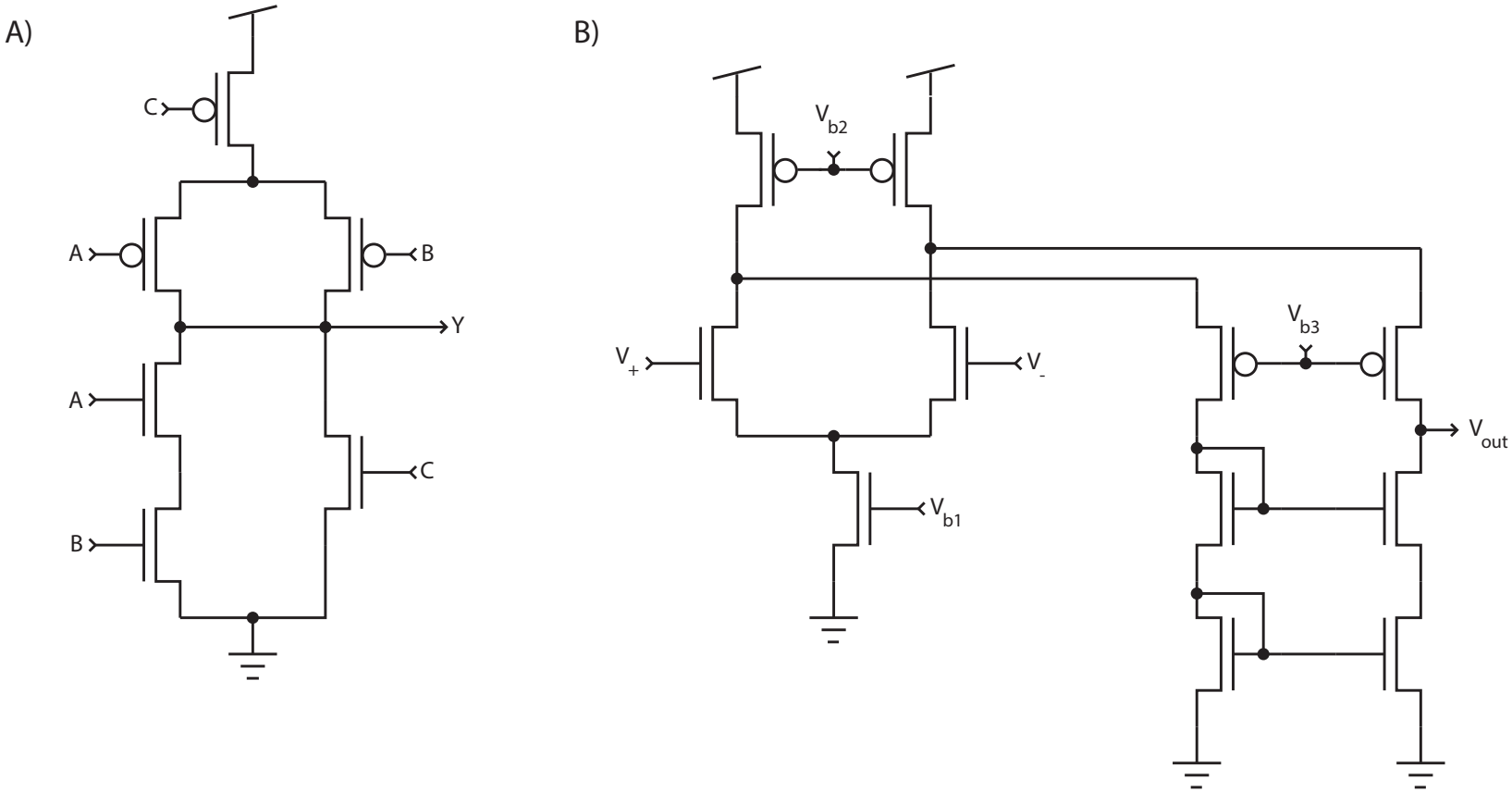
Course Organization

Course Goal (1/3)



Understand these two circuits thoroughly!

Course Goal (1/3)



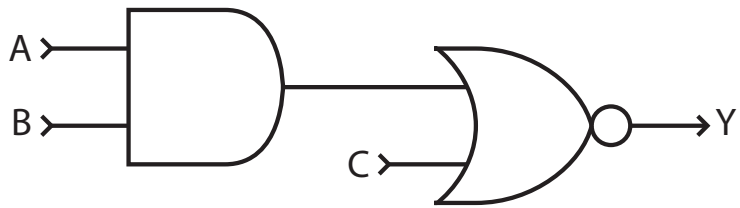
Understand these two circuits thoroughly!

Understand: analysis, properties, applications, limitations, tweaking, high level descriptions ...

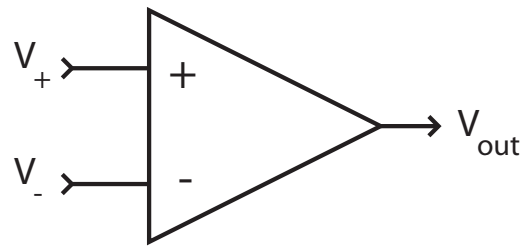
Course Goal (2/3)

And thereby understand the basic building blocks of analog and digital circuits:

A)



B)



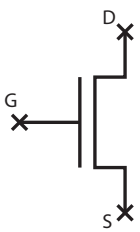
$$Y = \neg((A \wedge B) \vee C)$$

$$V_{out} = A(V_+ - V_-) \tag{1}$$

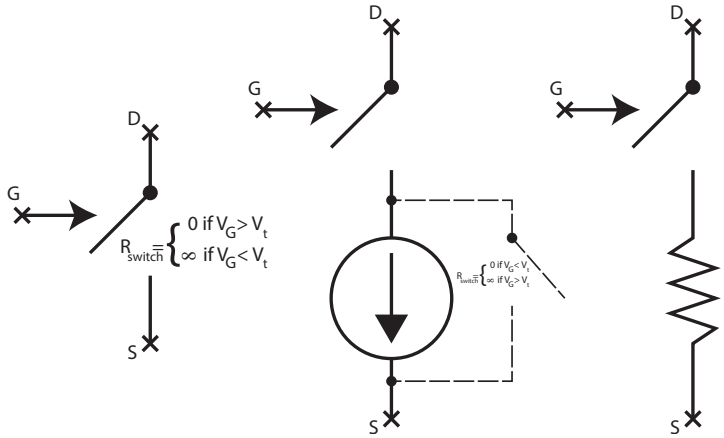
Course Goal (3/3)

... starting from modelling the basic active element of CMOS electronics, the field effect transistor (FET)

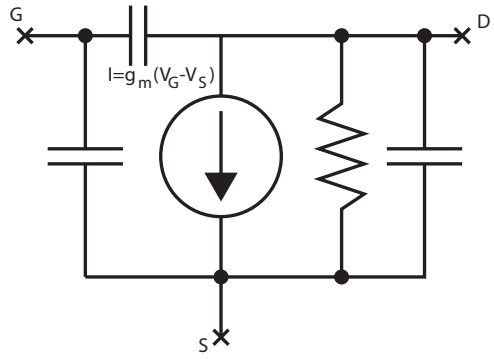
nFET symbol



Digital Abstractions



Analog Linear Abstraction



Content

Why Transistor Level Design of Integrated Circuits?

Why Transistor Level Digital?

Why Analog?

Course Goal

Course Organization

Course Organization (1/2)

Teaching: 17+ lectures (Mondays 10-12 seminar room Perl (OJD 2453), sometimes Tuesdays 14-16 in Java (OJD 2423)) , lecture foils, podcast (no guarantee! Backup: last year's recordings), course compendium: 'Microelectronics Essentials', selected papers, additional recommended books by Sedra & Smith, or Razavi, or Johns & Martin.

Paper exercises: exercises in preparation for exam (!) mostly from the compendium, Tuesdays 14-16 in seminar room Java (OJD 2423), teaching assistant: Halvor

Course Organization (2/2)

Labs: course-lab OJD 5419, 3 tasks (counting 40% towards final mark. Task 1 is only pass/not pass but a pass is required!), lab assistant: Andreas, workgroups with up to 3 students (register in Devilry), TA group sessions Wednesdays 10-12 and Fridays 10-12

Exam: written exam, counting 60% towards final mark, 12-June, 9:00-13:00, Silurveien, all paper material allowed, MATLAB installed

Tools Cadence, matlab, solder iron/bread board, lab equipment

Skills electronics, maths, physics, programming