

M 1.9 Concept Engineering Mixed-Technology Systems (CEMS)

M. Meiners

2024-11-23

Table of contents

Unsyllabus	1
I. Lecture	3
1. Introduction and Survey	5
1.1. Course Objectives	5
1.2. Scientific Computing	5
1.3. EDA Tools	5
1.4. OS-Tools	6
1.5. Code Editors	6
1.6. Data Science	6
1.7. Publish Computational Content	7
1.8. Are you writing or TeXing?	7
1.9. LaTeX Editors	7
1.10. Bibliography and LaTeX	7
1.11. Design Project	8
1.12. Course Prerequisites	8
1.13. Brave New World	9
1.14. From Sand to Silicon (Infineon, Dresden)	9
1.15. Sand to Silicon (GlobalFoundries, Dresden)	9
1.16. FinFET (Intel)	10
1.17. TSMC Fab (Next Gen 7/5 nm)	10
1.18. Once upon a time	10
1.19. First IC and today's chips	11
1.20. Packaging Densities	12
1.21. Moore's Law	12
1.22. System Hierarchy	13
1.23. System Assembly	14
1.24. Interfacing	15
1.25. Meeting a System (1)	15
1.26. System in a Package (SiP)	16
1.27. You will become an expert	16
1.28. Views on Hardware	17
1.29. Abstraction Layer	18
1.30. Design Flow	19
1.31. Verification	20

Table of contents

1.32. Frontend vs. Backend (analog)	21
1.33. Frontend vs. Backend (digital)	22
1.34. Analog Design Entry	23
1.35. Netlist	24
1.36. Layout	25
1.37. Digital Design Entry	26
1.38. Hardware Description Language	27
1.39. Technology-Gates and Netlisting	28
1.40. Standard Cell Layout	29
1.41. Sustainable Electronics	29
1.42. Why it is worth	29
1.43. Let's go to the beach	29
2. Systems Engineering for Gyros	31
2.1. Vehicles without ESP	31
2.2. Vehicle Dynamics Control Systems - ESP	32
2.3. Consumer Gyroscopes	32
2.4. MEMS for Automotive and Consumer Applications	35
2.5. MEMS Gyroscope in Action	36
2.6. Spring-Mass-Damping System	36
2.7. Functional Block Diagram	36
2.8. Multi-Domain Readout Block Diagram	37
2.9. Multi-Domain Modelling	37
2.10. ROM Modelling - Coordinate transformation	38
2.11. Gyro Behavioural Modelling	39
2.12. Mixed-domain simulation	39
2.13. Analog Closed-Loop Simulation	40
2.14. AMS Closed-Loop Simulation	41
2.15. Conclusion	41
II. Lab	43
3. MBSE and Design of an Inertial Sensor System	45
3.1. Design Project	45
3.2. Design Project Flow	45

List of Figures

1.1.	AMD Jaguar APU (CPU/GPU), 16 nm, 325 sqmm, 2016	9
1.2.	1906 Electron Tube	10
1.3.	1947 1st Transistor, Bell Labs	10
1.4.	1958 Jack Kilby's 1st IC	11
1.5.	Modern IC	11
1.6.	Wafer generations	12
1.7.	Blocks of an electronic system.	13
1.8.	Bottom-up Prozess, Integration.	14
1.9.	Interfacing.	15
1.10.	Wireless Communication System.	15
1.11.	Accelerometer.	16
1.12.	(c) M. Ortmanns, Univ. Ulm.	17
1.13.	(c) M. Ortmanns, Univ. Ulm.	18
1.14.	(c) M. Ortmanns, Univ. Ulm.	19
1.15.	(c) M. Ortmanns, Univ. Ulm.	20
1.16.	(c) M. Ortmanns, Univ. Ulm.	21
1.17.	(c) M. Ortmanns, Univ. Ulm.	22
2.1.	The moose test / elk test.	31
2.2.	Market launch of gyros.	32
2.3.	Model of gyroscope.	39
2.4.	Analog simulation with Cadence.	40
2.5.	AMS simulation with Cadence.	41

List of Tables

Unsyllabus

Name	Description
Course	M 1.9 Concept Engineering Mixed-Technology Systems (CEMS)
Term	Winter 2024/25
Instructor	Prof. Dr.-Ing. M. Meiners, Dipl.-Ing. (FH) T. Ziemann
Lectures	Thu., 9:45 h - 13:00 h
Room	E 507 ELIE Lab

Part I.

Lecture

1. Introduction and Survey

1.1. Course Objectives

- Interfacing Microsystems
 - Mixed-technology systems (System-on-Chip, SoC)
 - System analysis
 - System specification
- Design Methodology
 - Seamlessly modeling and design over all physical domains
- PCB (System) and IC Design
 - Architecture
 - Partitioning
 - Layout

1.2. Scientific Computing

- Python (Anaconda)
- Matlab (Campus Lizenz)
- Command-line tools

1.3. EDA Tools

- PCB / System Design
 - LTspice
 - KiCad EDA
 - Altium Designer
 - SiemensEDA PCB tools
 - cadence System Design & Analysis
- IC / Silicon Design

1. Introduction and Survey

- [IIC-OSIC-TOOLS](#) (open-source)
- [SiemensEDA](#) IC tools
- [cadence](#) IC Design & Verification
- [synopsys silicon design](#) (IC)

1.4. OS-Tools

- Microsoft-Terminal
- Microsoft-PowerShell
- MacOS-Terminal
- Linux/MacOS Shell zsh-tools,
- git (Versionskontrolle)

1.5. Code Editors

- Visual Studio Code
- Spyder IDE
- Thonny (Micro-)Python IDE
- Emacs
- Vim

1.6. Data Science

- File system: Files and directories
- Tabular data: Comma/Tab-Separated-Values (CSV/TSV), Spreadsheet (.xlsx, .ods)
- Special formats, e.g. MATLAB mat, HDF5
- Embedded [Databases](#)
 - [SQL](#), z.B. [SQLite](#)
 - [OLAP](#), z.B. [DuckDB](#)

1.7. Publish Computational Content

- Jupyter-Book
- quarto
 - Manuscripts

1.8. Are you writing or TeXing?

- MikTeX (Windows, MacOS, Linux)
- MacTeX (MacOS)
- TeXLive (Linux)

1.9. LaTeX Editors

- IDE's
 - TeXStudio
 - TeXMaker
- Collaborative Frameworks
 - Overleaf, Online LaTeX
 - CoCalc - Online LaTeX

1.10. Bibliography and LaTeX

- Citavi im Detail > Titel exportieren > Export nach BibTeX
- RefWorks - Library Guide Univ. Melbourne
- Benutzerdefinierte BibTex-Keys mit Zotero | nerdspause
- JabRef - Library Guide Univ. Melbourne
- EndNote - Library Guide Univ. Melbourne

1. Introduction and Survey

1.11. Design Project

Model-Based Systems Engineering of an Inertial Sensor System (MBSE).

- System level, behavioural model
 - Matlab/Simulink,
 - Python
 - HDL (Verilog-ams, VHDL-AMS)
- Circuit level, SPICE with behavioural blocks, e.g. OTA and comparator
- PCB level
 - [ESP8266 NodeMCU](#),
 - [TIs ADS1115](#),
 - [ADs ADXL335](#)
- IC level
- **Final Oral Exam/Project Presentation**

1.12. Course Prerequisites

- Fundamentals of linux operating systems
- Fundamentals of microelectronics
 - Device physics and models
 - Transistor level analog circuits, elementary gain stages
- Fundamentals of analog circuit design
 - Operational amplifier
 - Active filter design
 - Noise analysis
 - Switched-capacitor techniques
- Prior exposure to SPICE, Matlab, Python or equivalent.

1.13. Brave New World

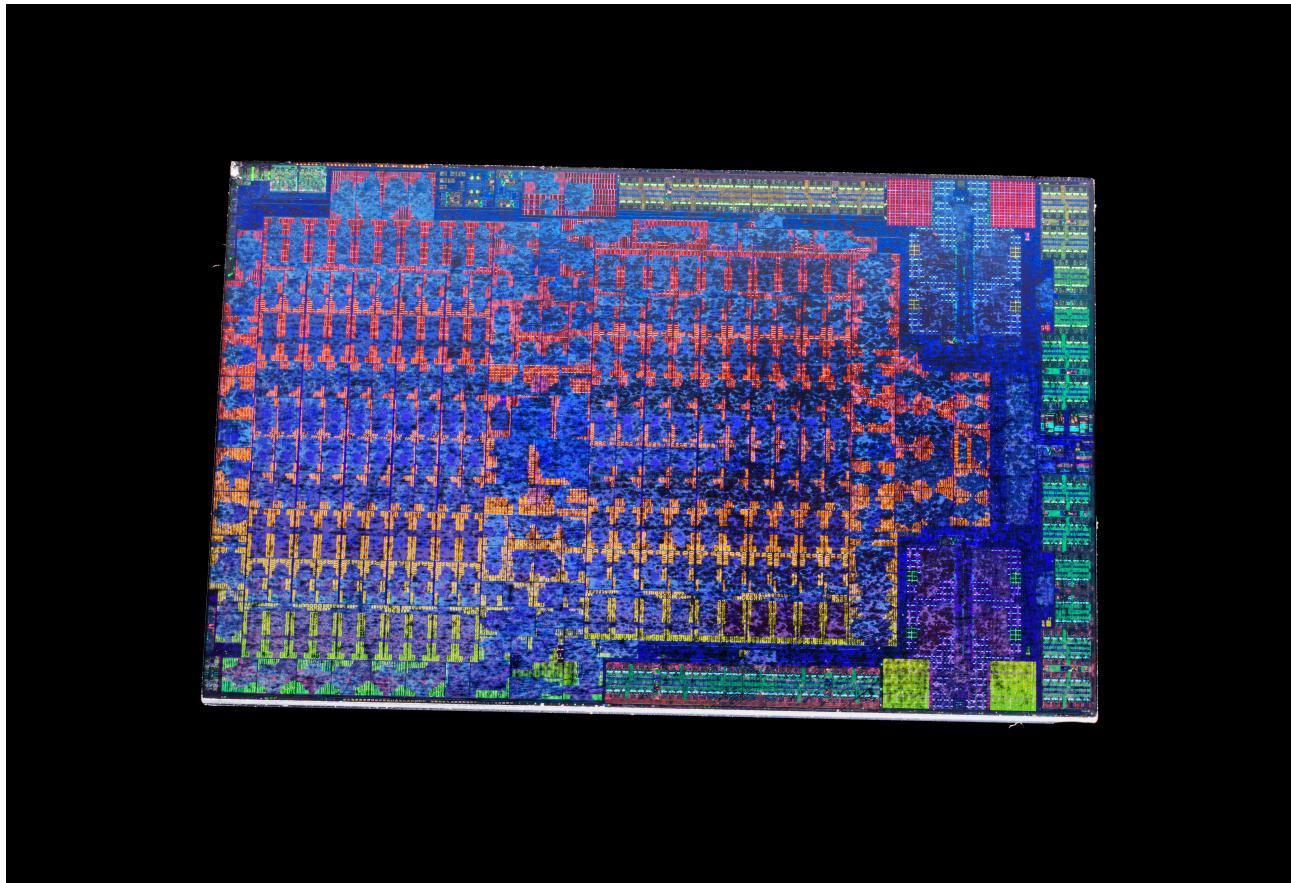


Figure 1.1.: AMD Jaguar APU (CPU/GPU), 16 nm, 325 sqmm, 2016

1.14. From Sand to Silicon (Infineon, Dresden)

https://youtu.be/bor0qLifjz4?list=PLO_wT97BGA6xC6hNy9VGtt1bKwVuQXI5B

1.15. Sand to Silicon (GlobalFoundries, Dresden)

https://www.youtube.com/embed/UvluuAIiA50?list=PLO_wT97BGA6xC6hNy9VGtt1bKwVuQX15B

1. Introduction and Survey

1.16. FinFET (Intel)

https://www.youtube.com/embed/_VMYPLXnd7E

1.17. TSMC Fab (Next Gen 7/5 nm)

<https://www.youtube.com/embed/Hb1WDxSoSec>

1.18. Once upon a time ...



Figure 1.2.: 1906 Electron Tube

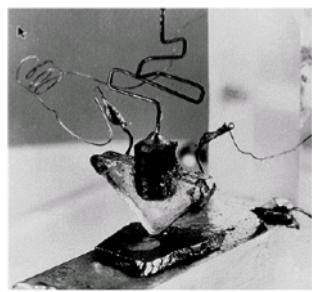


Figure 1.3.: 1947 1st Transistor, Bell Labs

1.19. First IC and today's chips

1.19. First IC and today's chips



Figure 1.4.: 1958 Jack Kilby's 1st IC

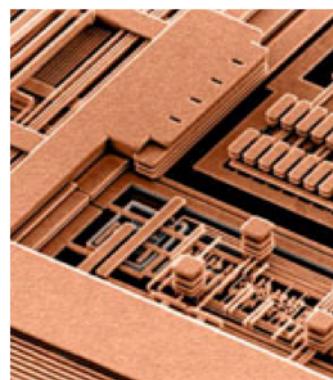


Figure 1.5.: Modern IC

1. Introduction and Survey

1.20. Packaging Densities

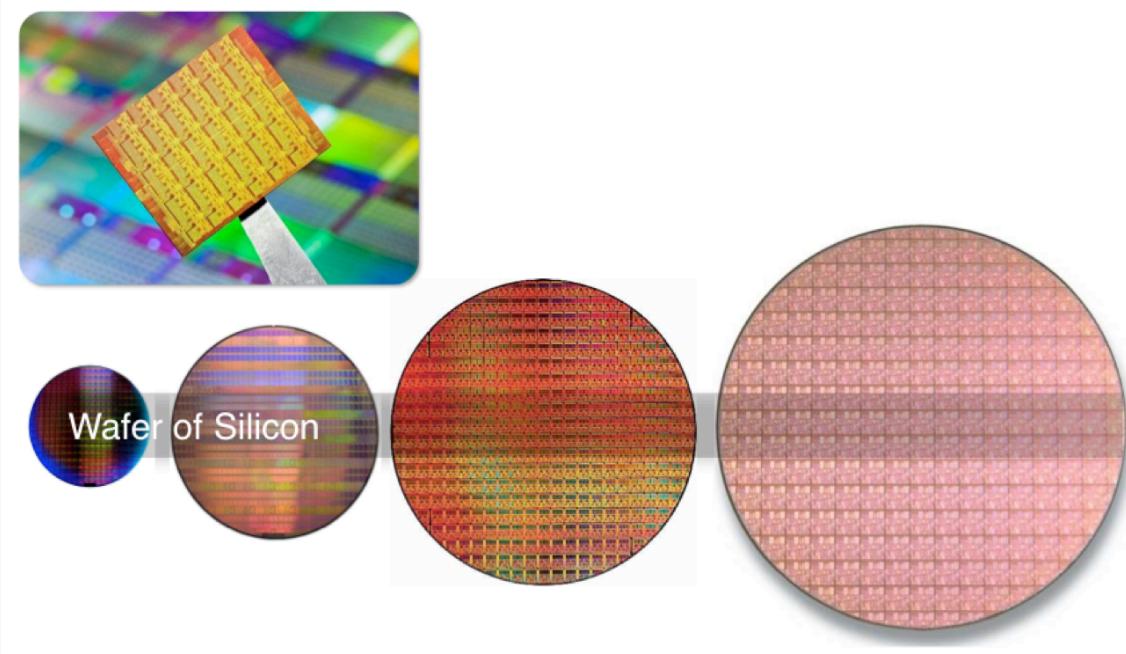


Figure 1.6.: Wafer generations

1.21. Moore's Law

https://www.youtube.com/embed/basGrfRDqts?list=PLO_wT97BGA6xC6hNy9VGtt1bKwVuQXI5B

1.22. System Hierarchy

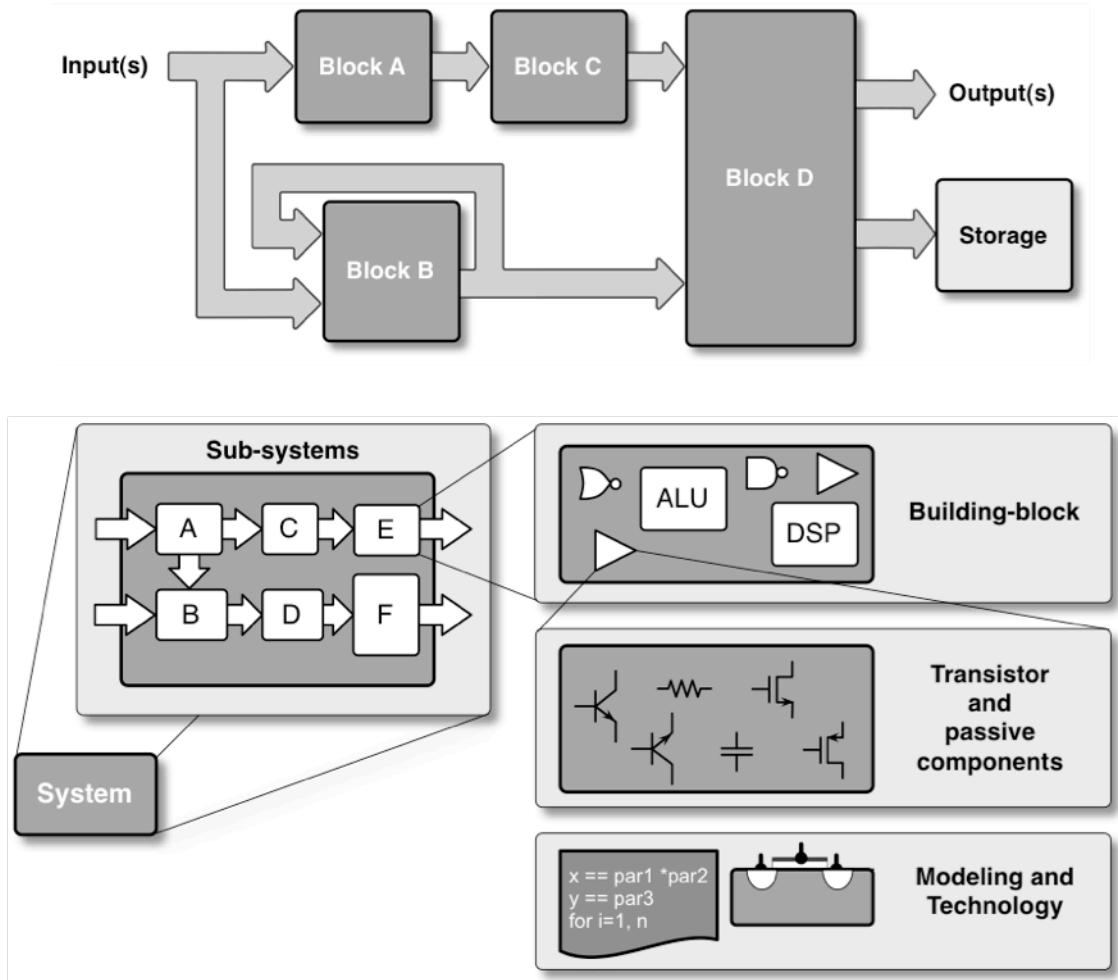
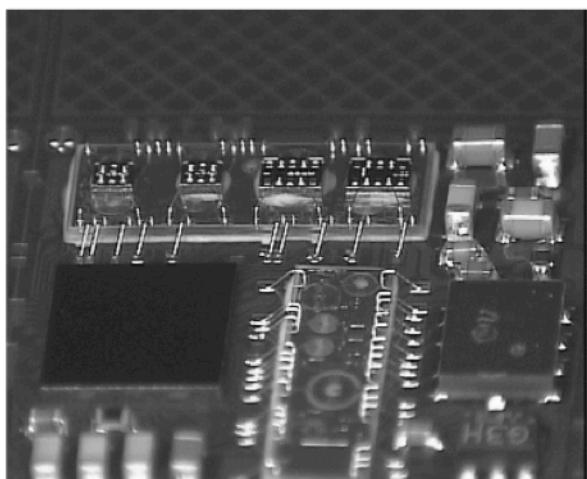


Figure 1.7.: Blocks of an electronic system.

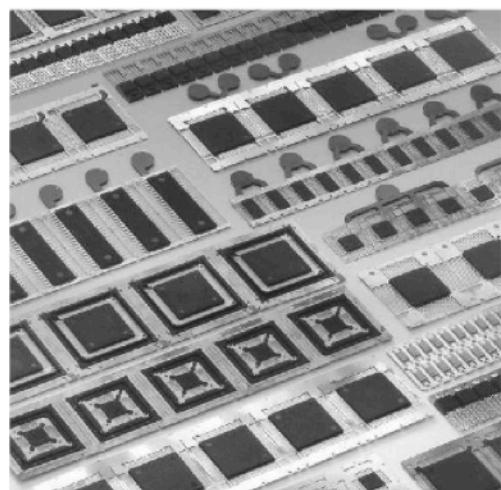
- Use hierarchy to describe complex systems
- Devide and conquer

1. Introduction and Survey

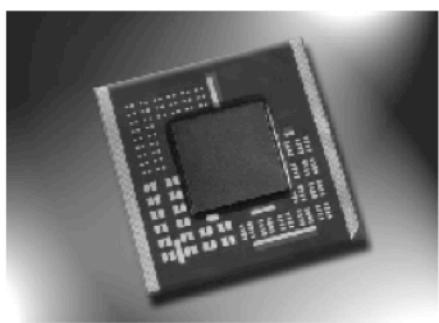
1.23. System Assembly



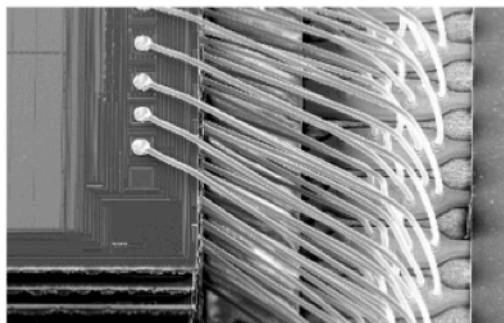
(a)



(b)



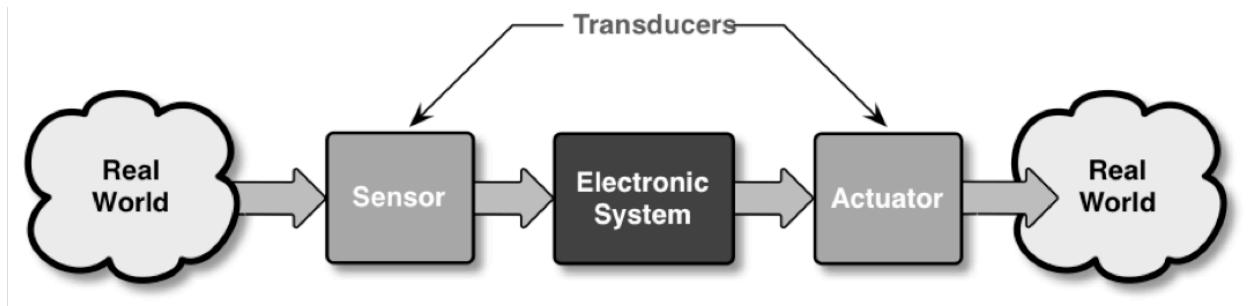
(c)



(d)

Figure 1.8.: Bottom-up Prozess, Integration.

1.24. Interfacing



Entire system involving signals of real world.

Figure 1.9.: Interfacing.

1.25. Meeting a System (1)

Block diagram of a wireless communication system

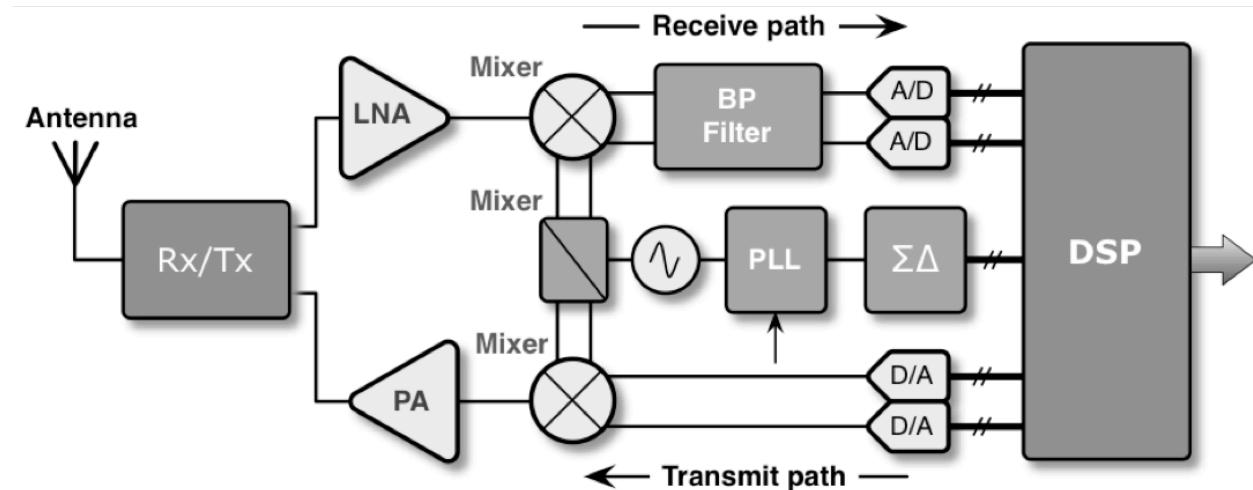
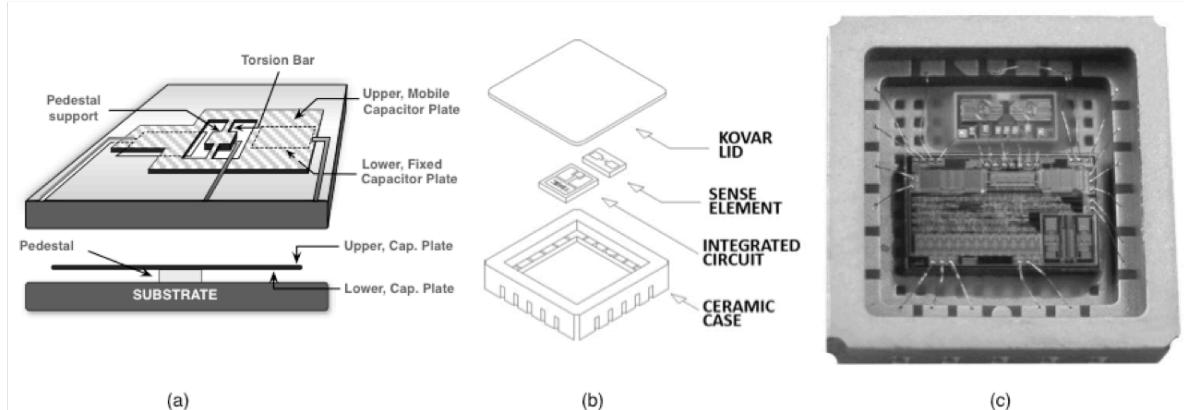


Figure 1.10.: Wireless Communication System.

1. Introduction and Survey

1.26. System in a Package (SiP)



- (a) Micro structure of an accelerometer.
- (b) Assembling diagram of the system-on- package.
- (c) Microphotograph. (*Courtesy of Silicon Designs, Inc.*).

Figure 1.11.: Accelerometer.

1.27. You will become an expert

Indicators.

- Background Knowledge
 - System Knowledge, Architecture, Processing, Implementation
- Subconscious Knowledge
 - Memorized experiences of success stories and dead ends
- Special Knowledge
 - Discipline related knowledge, e.g. physics, hardware, software
- Teamwork
 - Communication abilities, reporting and presentation
- Creativity
- Tool-Knowlege

1.28. Views on Hardware

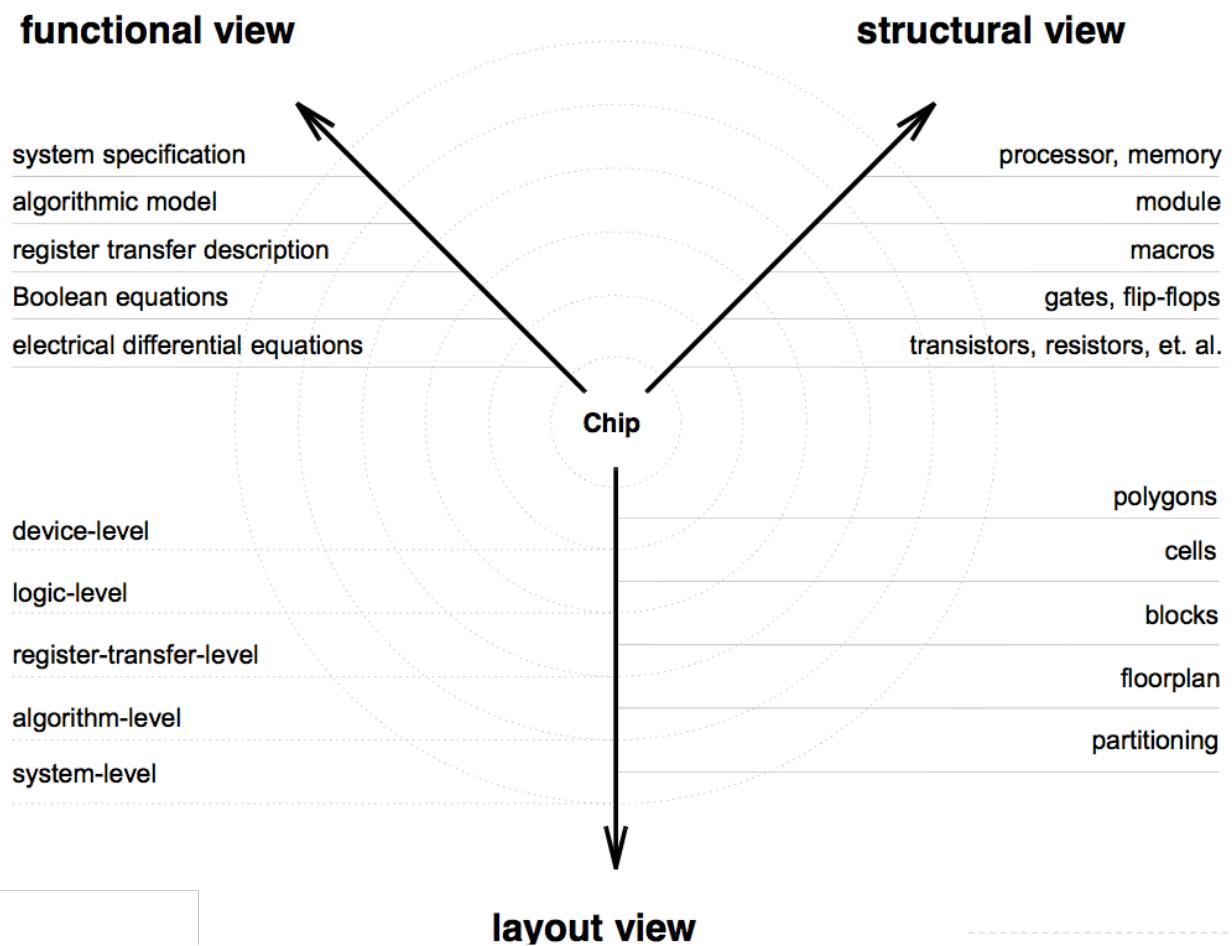


Figure 1.12.: (c) M. Ortmanns, Univ. Ulm.

1. Introduction and Survey

1.29. Abstraction Layer

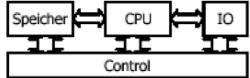
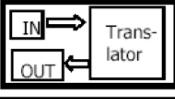
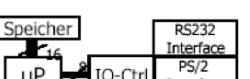
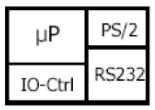
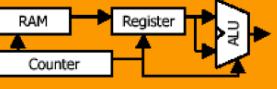
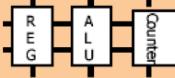
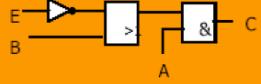
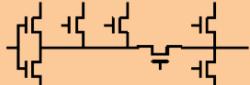
	function	structure	layout
system level	Inputs : Keyboard Output: Display Funktion:		
algorithmic level	while input Read „Schilling“ Calulate Euro Display „Euro“		
register transfer level	if A='1' then B:= B+1 else B:= B end if		
logic level	D = NOT E C = (D OR B) AND A		
device level	$\frac{dU}{dt} = R \frac{dI}{dt} + \frac{I}{C} + L \frac{d^2I}{dt^2}$		

Figure 1.13.: (c) M. Ortmanns, Univ. Ulm.

1.30. Design Flow

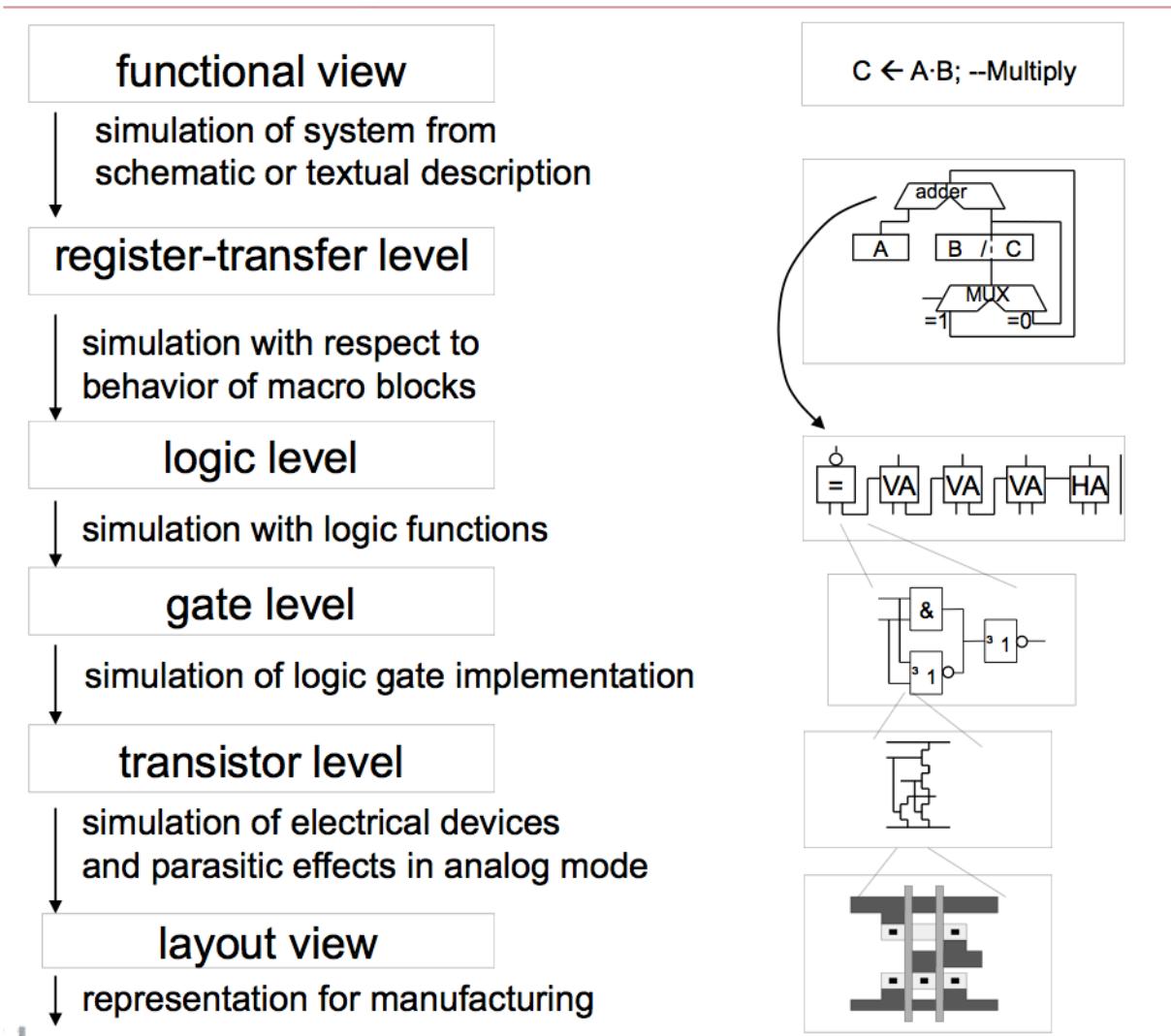


Figure 1.14.: (c) M. Ortmanns, Univ. Ulm.

1.31. Verification

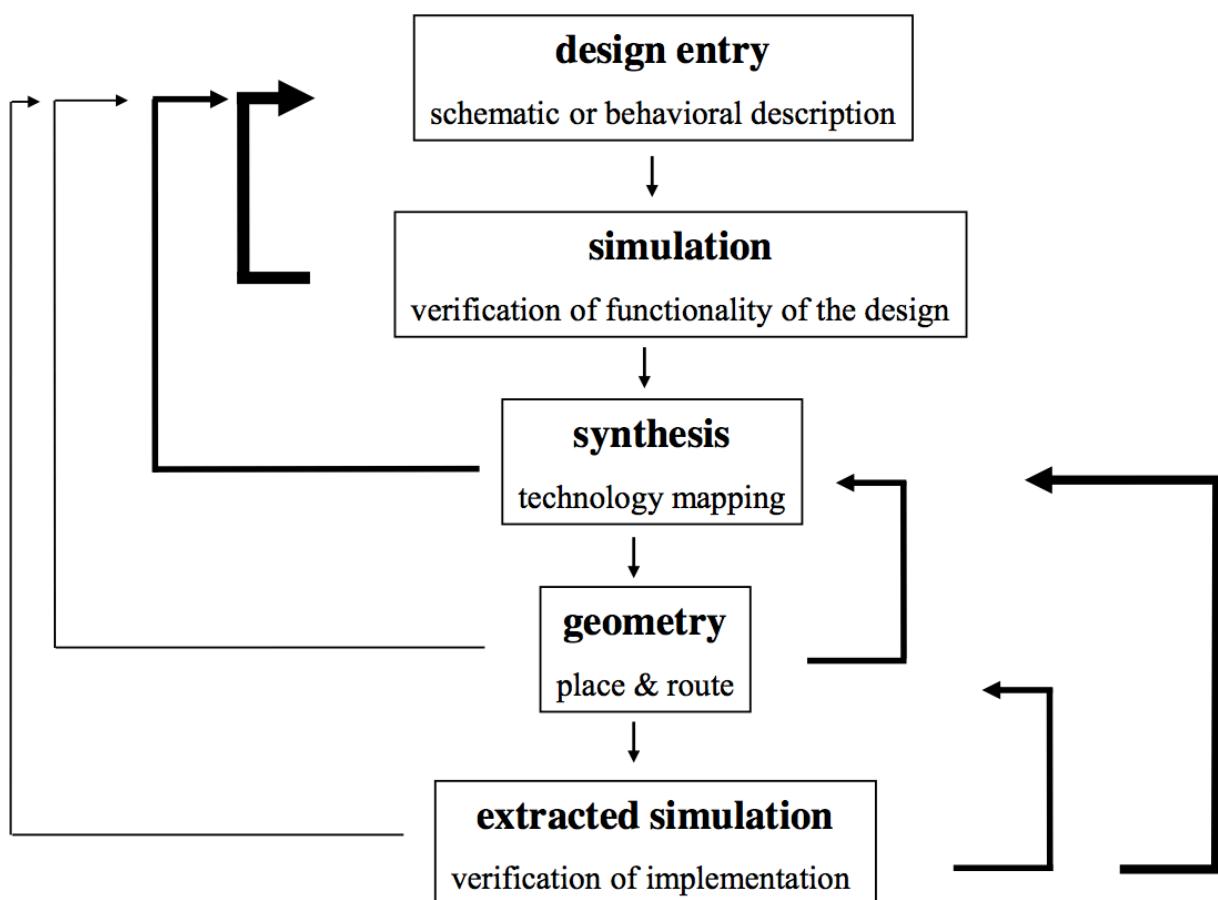


Figure 1.15.: (c) M. Ortmanns, Univ. Ulm.

1.32. Frontend vs. Backend (analog)

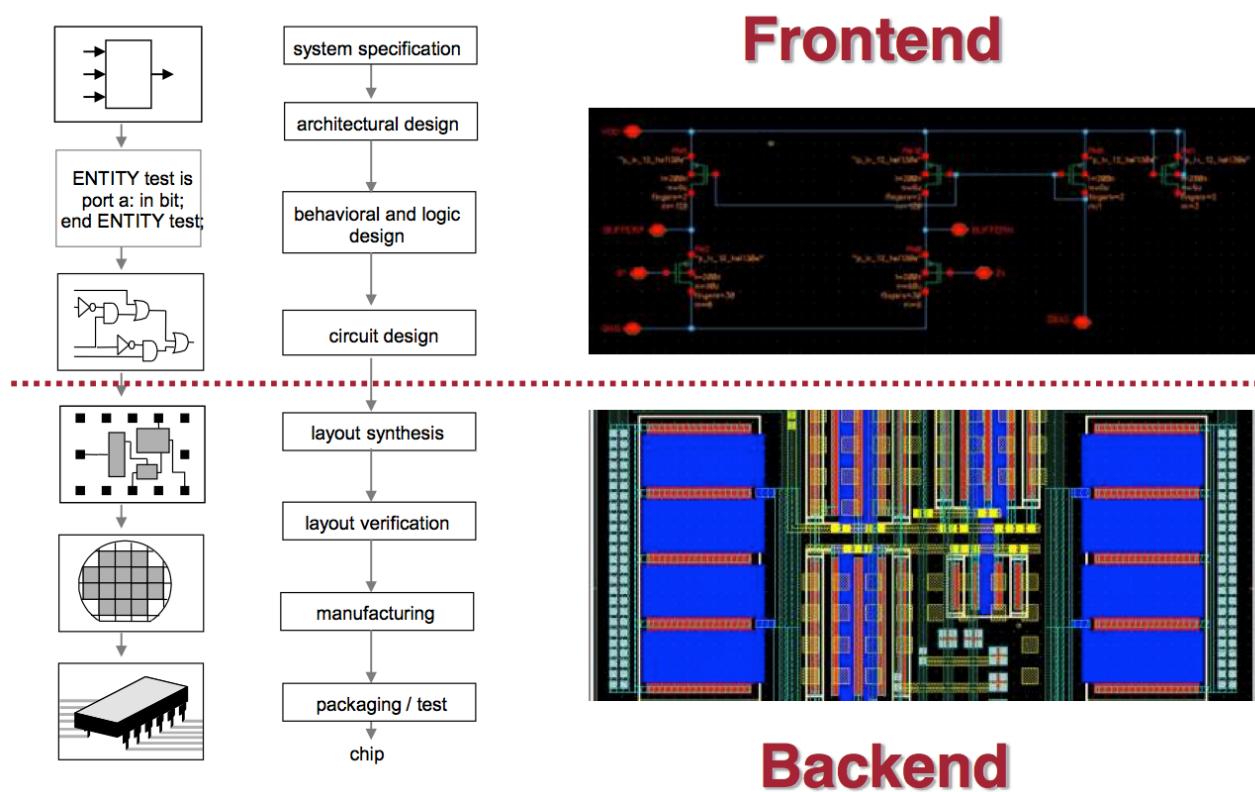


Figure 1.16.: (c) M. Ortmanns, Univ. Ulm.

1. Introduction and Survey

1.33. Frontend vs. Backend (digital)

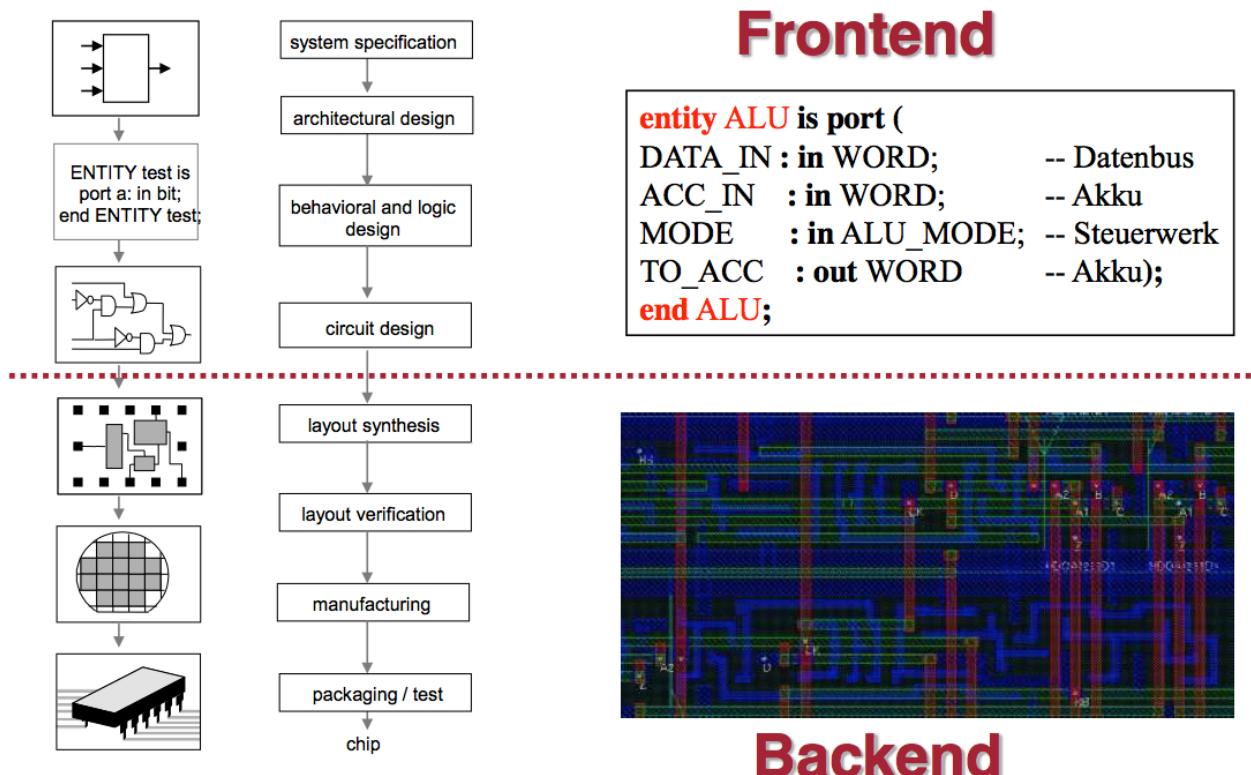
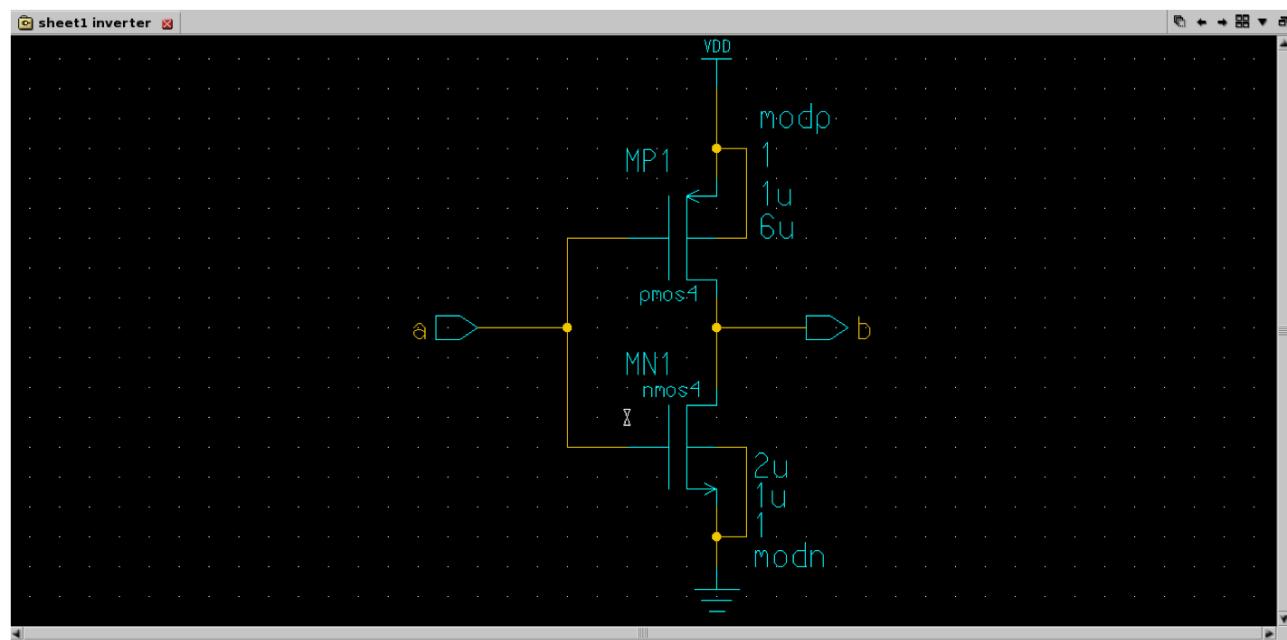


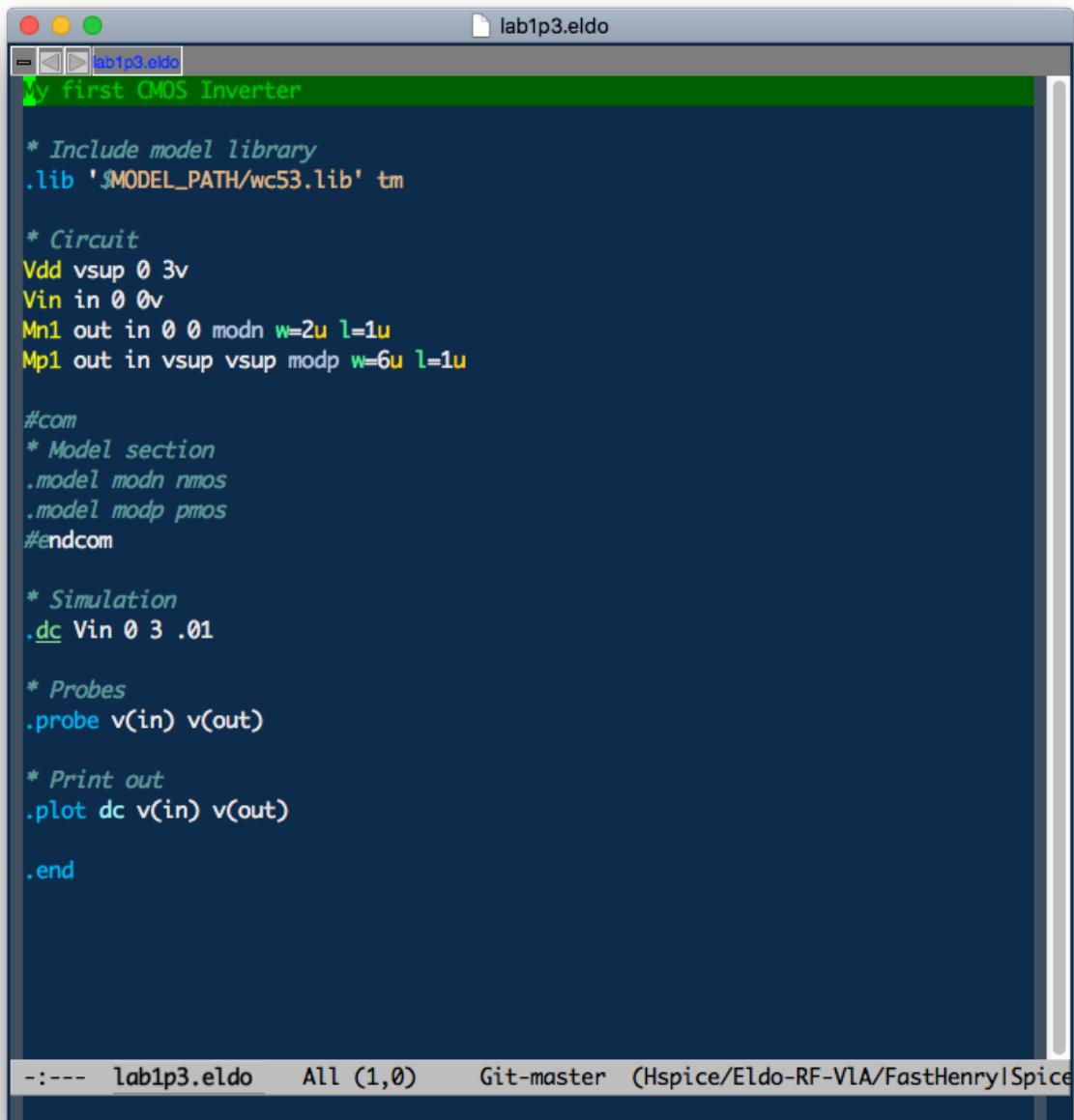
Figure 1.17.: (c) M. Ortmanns, Univ. Ulm.

1.34. Analog Design Entry



1. Introduction and Survey

1.35. Netlist



The screenshot shows a terminal window titled "lab1p3.eldo" containing a netlist for a CMOS inverter. The netlist includes comments for model library inclusion, circuit components (Vdd, Vin, Mn1, Mp1), model sections for NMOS and PMOS, simulation setup (.dc), probe definitions, plot commands, and an end statement. The code uses color coding for different syntax elements.

```
* Include model library
.lib '$MODEL_PATH/wc53.lib' tm

* Circuit
Vdd vsup 0 3v
Vin in 0 0v
Mn1 out in 0 0 modn w=2u l=1u
Mp1 out in vsup vsup modp w=6u l=1u

#com
* Model section
.model modn nmos
.model modp pmos
#endcom

* Simulation
.dc Vin 0 3 .01

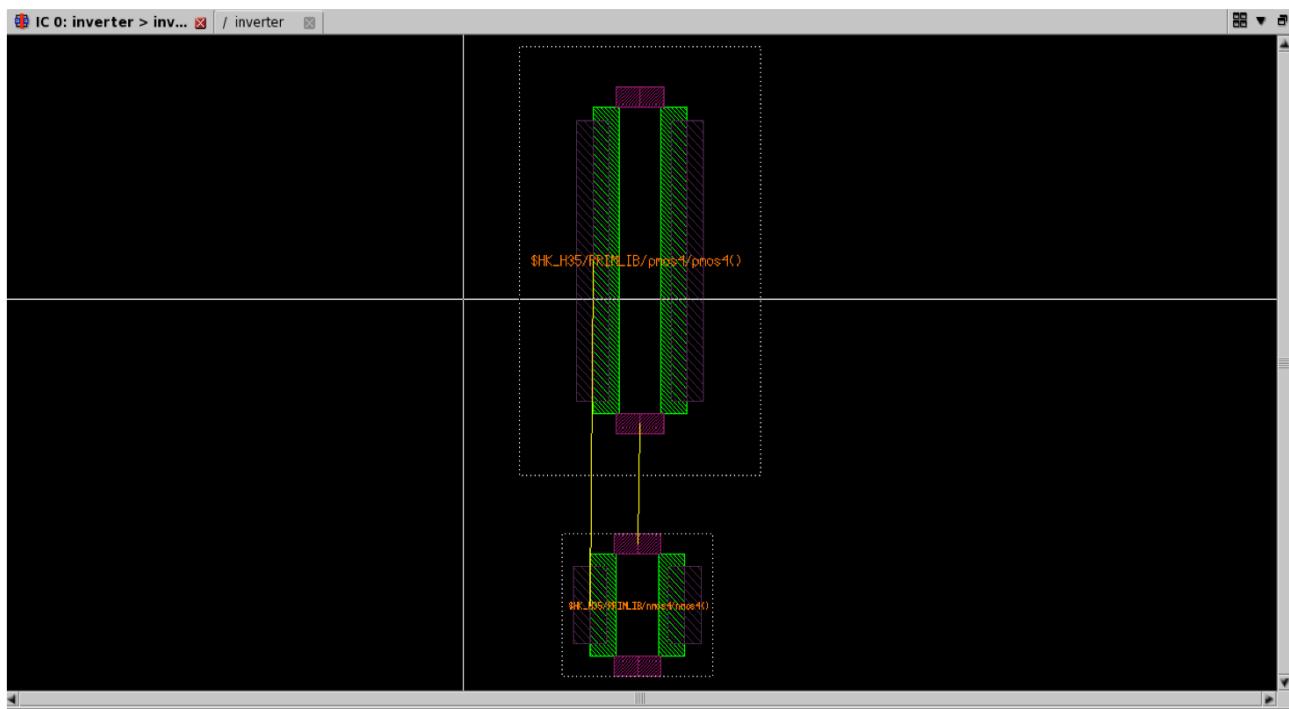
* Probes
.probe v(in) v(out)

* Print out
.plot dc v(in) v(out)

.end

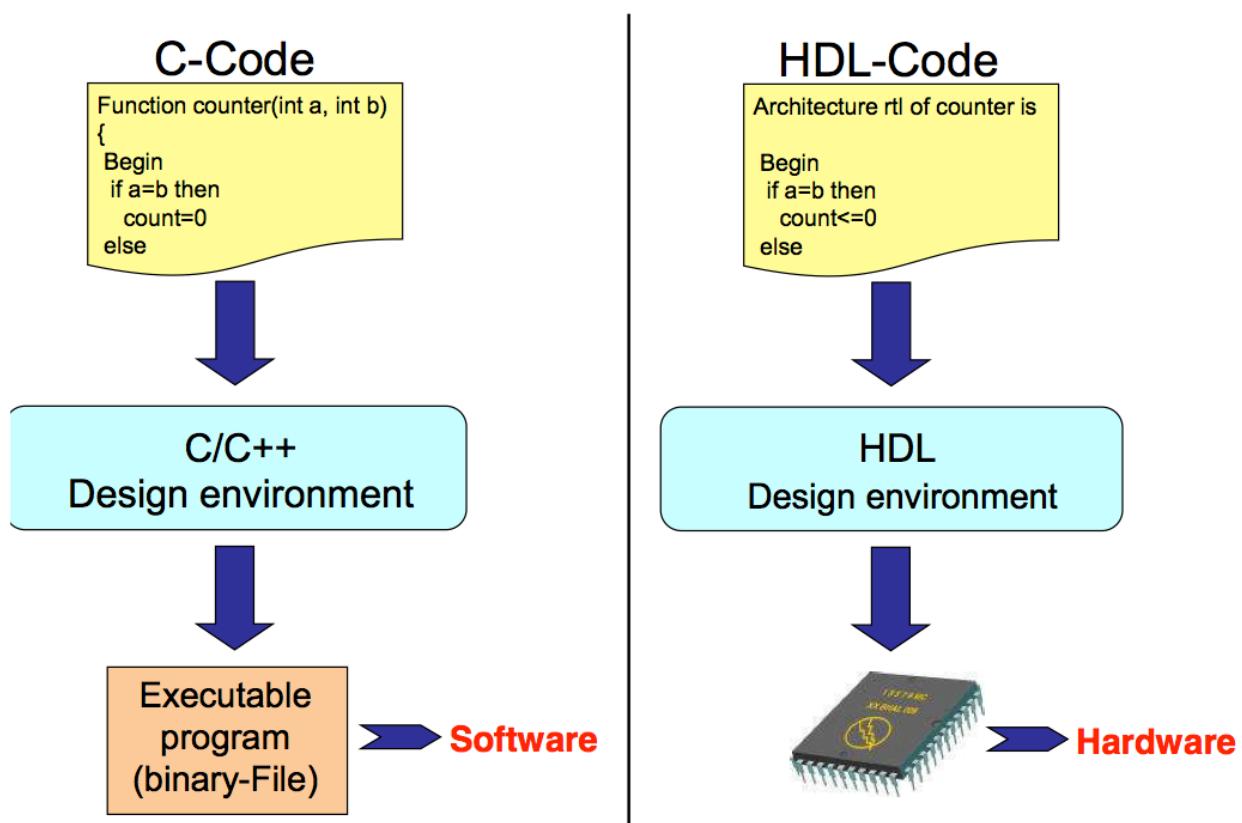
-:--- lab1p3.eldo All (1,0) Git-master (Hspice/Eldo-RF-VIA/FastHenry|Spice
```

1.36. Layout

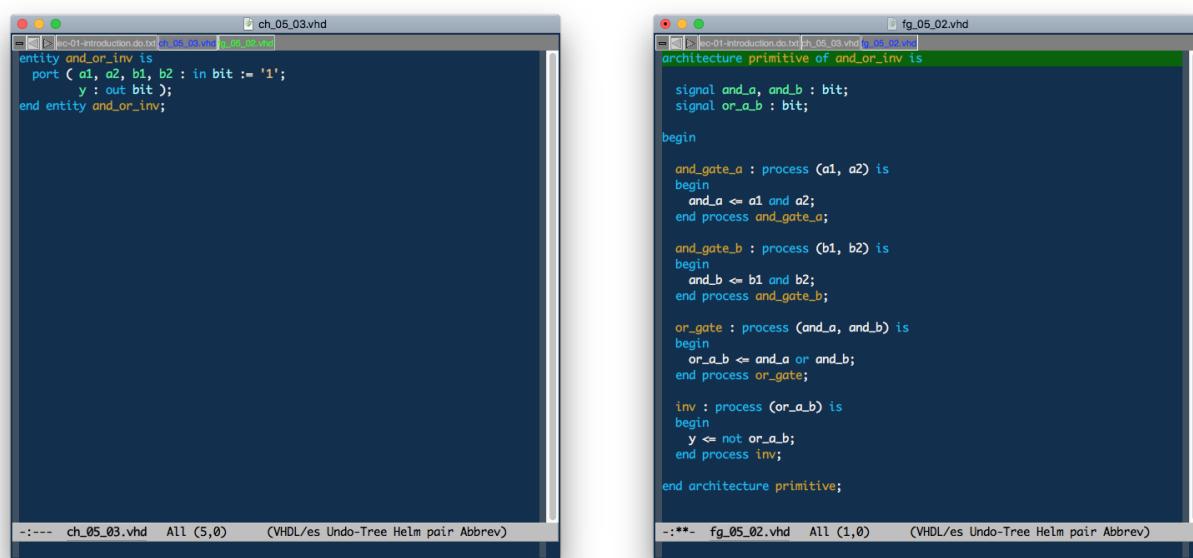


1. Introduction and Survey

1.37. Digital Design Entry



1.38. Hardware Description Language



The image shows two side-by-side windows of a VHDL editor. Both windows have a dark blue background and white text.

Left Window (ch_05_03.vhd):

```

entity and_or_inv is
  port ( a1, a2, b1, b2 : in bit := '1';
        y : out bit );
end entity and_or_inv;

```

Right Window (fg_05_02.vhd):

```

architecture primitive of and_or_inv is
begin
  and_gate_a : process (a1, a2) is
  begin
    and_a <= a1 and a2;
  end process and_gate_a;

  and_gate_b : process (b1, b2) is
  begin
    and_b <= b1 and b2;
  end process and_gate_b;

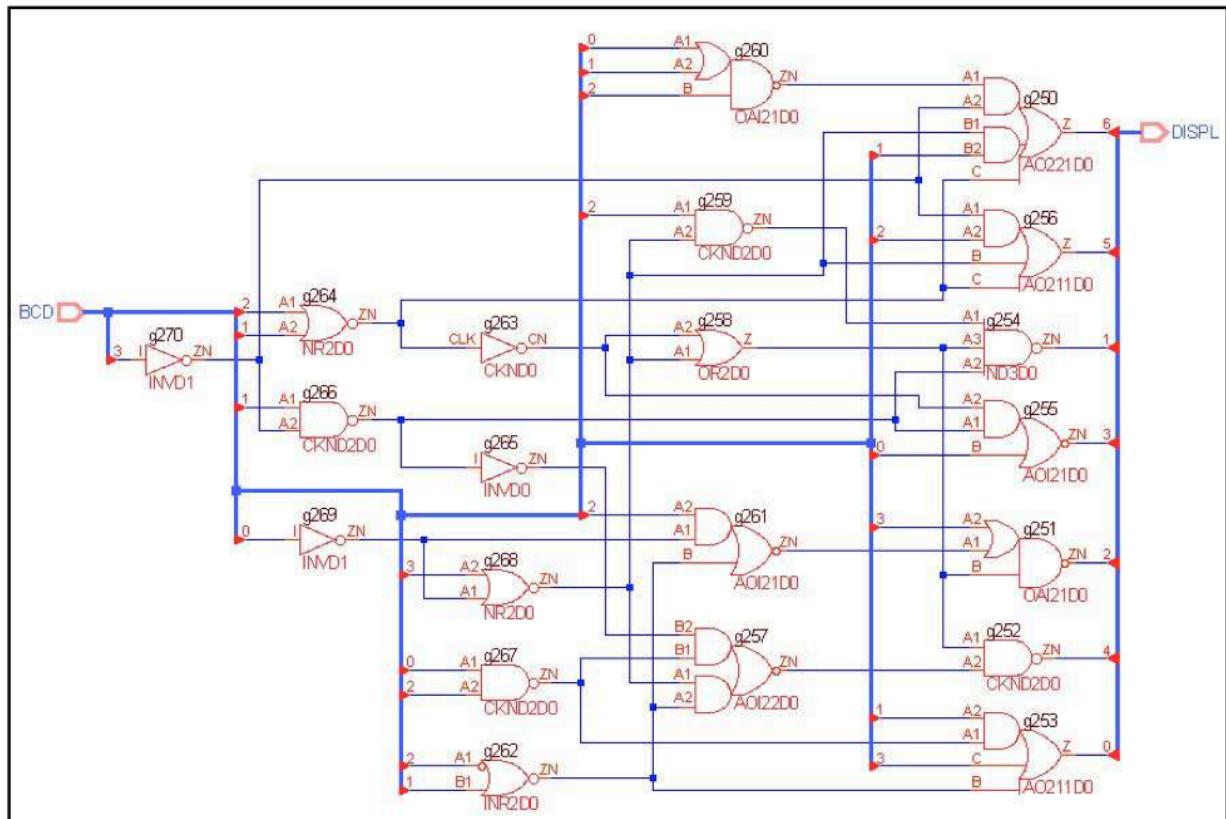
  or_gate : process (and_a, and_b) is
  begin
    or_ab <= and_a or and_b;
  end process or_gate;

  inv : process (or_ab) is
  begin
    y <= not or_ab;
  end process inv;
end architecture primitive;

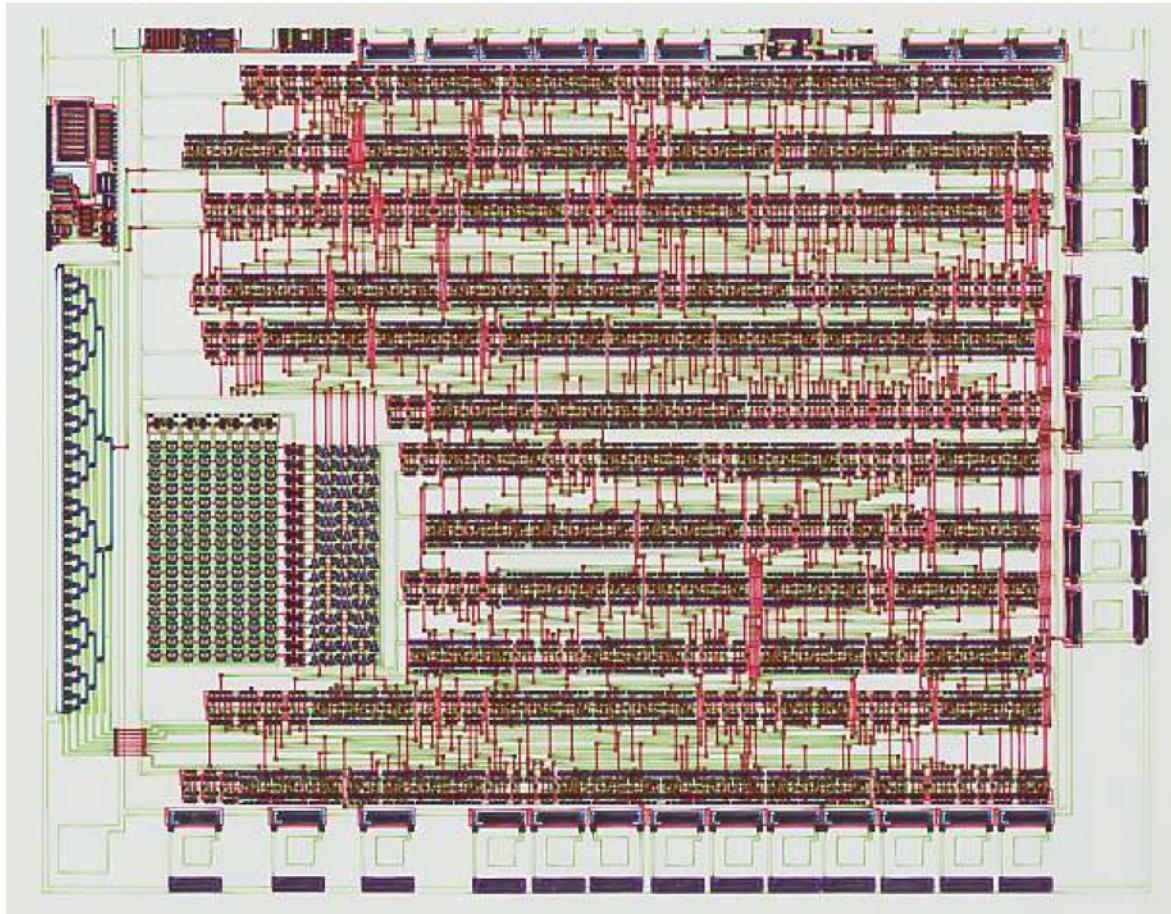
```

1. Introduction and Survey

1.39. Technology-Gates and Netlisting



1.40. Standard Cell Layout



1.41. Sustainable Electronics ...

<https://www.youtube.com/embed/7S5IuaKiZIY>

1.42. Why it is worth ...

<https://www.youtube.com/embed/SwPGxwBZw6I>

1.43. Let's go to the beach ...

<https://www.youtube.com/embed/ekkJlQf-K4I>

2. Systems Engineering for Gyros

2.1. Vehicles without ESP



Figure 2.1.: The moose test / elk test.

2. Systems Engineering for Gyros

2.2. Vehicle Dynamics Controls Systems - ESP

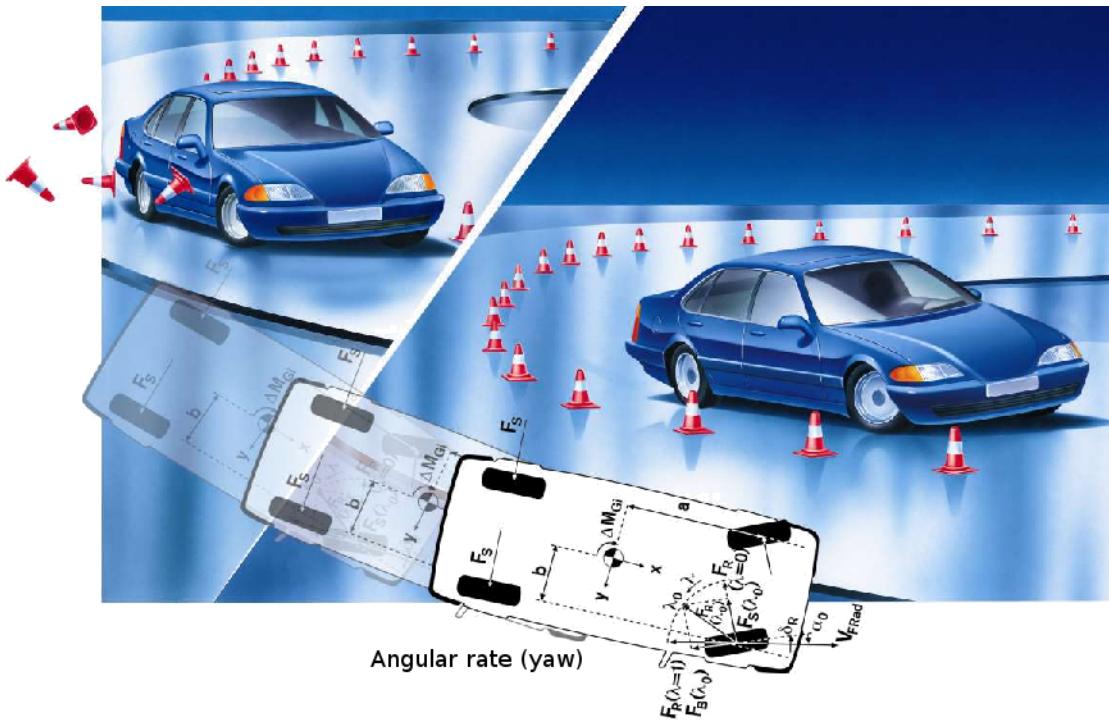


Figure 2.2.: Market launch of gyros.

2.3. Consumer Gyroscopes

Market segments. * Mobile phones * Digital cameras * Pointing devices * Gaming consoles * GPS portables

2.3. Consumer Gyroscopes



Newsstand

Notification Center

Messages



2. Systems Engineering for Gyros



2.4. MEMS for Automotive and Consumer Applications



2.4. MEMS for Automotive and Consumer Applications

- Jiri Marek, Senior Vice President, Robert Bosch, Reutlingen, Germany

<https://www.youtube.com/embed/5MKnlsLtK34>

2. Systems Engineering for Gyros

2.5. MEMS Gyroscope in Action

[..../mov/lec-02-mm3drive.mp4](#)

[..../mov/lec-02-mm3sense.mp4](#)

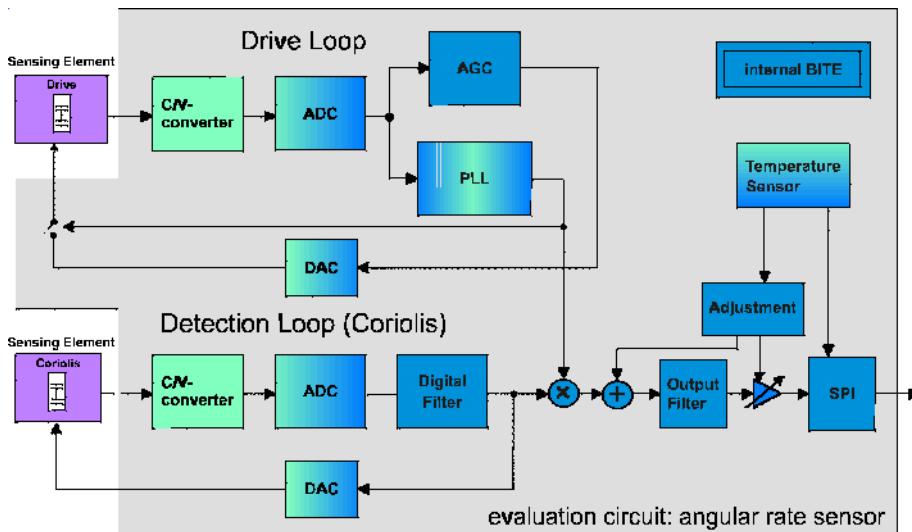
2.6. Spring-Mass-Damping System

- 1-D equation of motion (EoM) $F = m\ddot{x} + d\dot{x} + kx$
- Laplace transformation

$$H(s) = \frac{1}{ms^2 + ds + k} \quad (2.1)$$

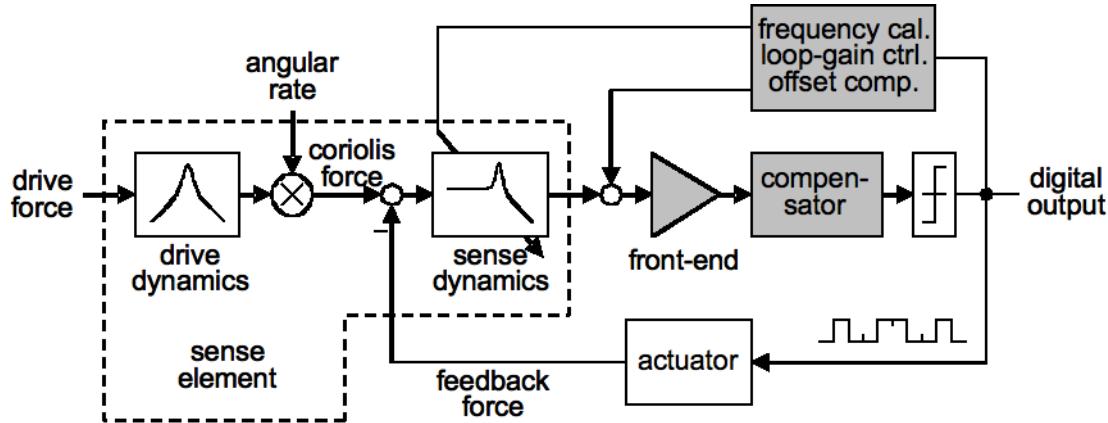
$$= \frac{\frac{1}{m}}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad (2.2)$$

2.7. Functional Block Diagram



- Coriolis force principle, $F_C = 2m(\mathbf{v} \times \boldsymbol{\Omega})$
- Drive loop to have an accelerated mass
- Sense loop to detect angular rate
- Distinction of closed-loop and open-loop system

2.8. Multi-Domain Readout Block Diagram



$$H_s(s) = \frac{\frac{1}{m}}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad \text{sensor} \quad (2.3)$$

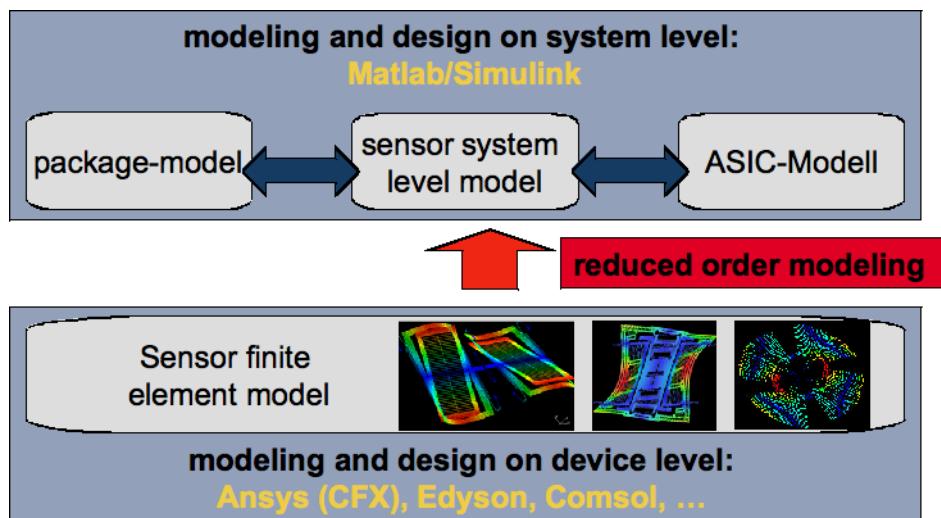
$$H_{CV}(s) = \frac{g_m}{C_L} \frac{1 - e^{-sT_{int}}}{s} \quad \text{CV converter} \quad (2.4)$$

$$H_{lf}(z) = -\frac{z}{z+a} \frac{z^2 + b_1 z + b_0}{z^2 + c_1 z + c_0} \quad \text{loop filter} \quad (2.5)$$

2.9. Multi-Domain Modelling

- Describing kinematic and electrical behaviour with the help of HDL
 - VHDL, VHDL-AMS
 - Verilog, Verilog-a, Verilog-ams
- Using ROM for a MATLAB/SIMULINK model and real-time workshop to port model for use with Cadence → Verilog-AMS is used for wrapping.
- Parasitic SPICE circuit equivalent from FEM sensor model and layout extraction

2.10. ROM Modelling - Coordinate transformation



$$M\ddot{x} + Kx = F \quad \text{FEM, 100.000 DOF} \quad (2.6)$$

$$M\phi\ddot{q} + K\phi q = F \quad (2.7)$$

$$\underbrace{\phi' M \phi}_{\tilde{M}} \underbrace{\ddot{q}}_{\tilde{q}} + \underbrace{\phi' K \phi}_{\tilde{K}} q = \phi' F \quad (2.8)$$

$$\tilde{M}\tilde{q} + \tilde{K}q = \tilde{F} \quad \text{ROM, approx. 10 DOF} \quad (2.9)$$

2.11. Gyro Behavioural Modelling

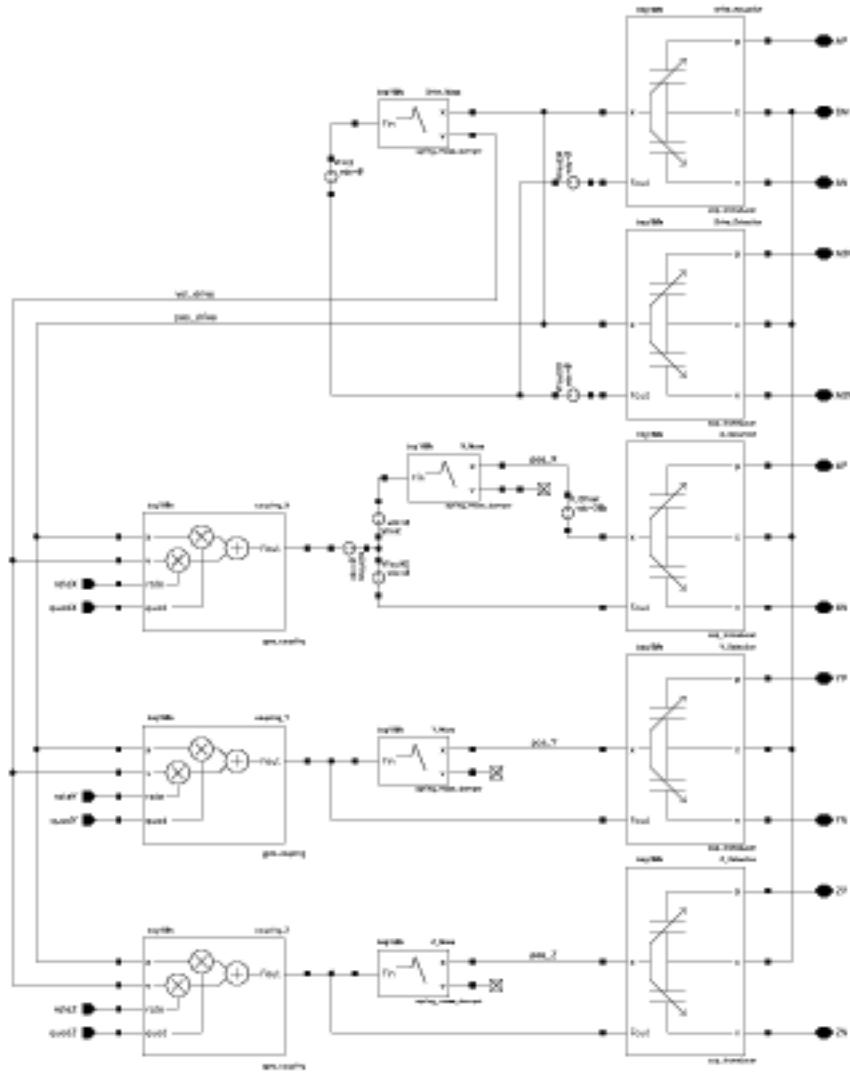


Figure 2.3.: Model of gyroscope.

2.12. Mixed-domain simulation

- All models from previous slide can be used in Cadence design frame work
- Pure analog closed-loop transient simulation with Spectre (turbo, aps), circuits and verilog-a model
- Pure analog closed-loop simulation with SPICE circuit equivalent

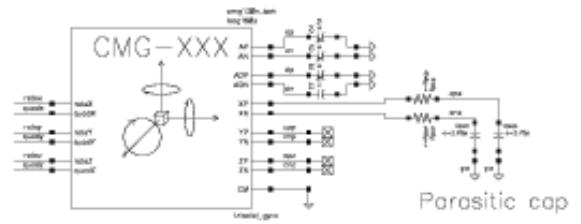
2. Systems Engineering for Gyros

- Mixed-domain, mixed-mode simulation with AMSDesigner (ncsim, spectre-turbo/aps)

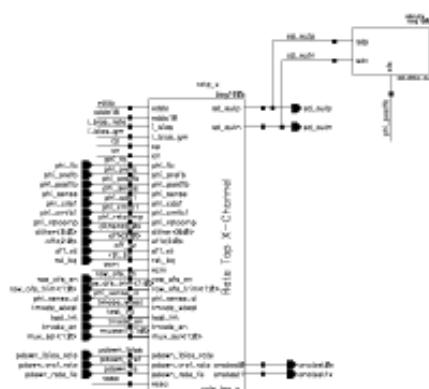
2.13. Analog Closed-Loop Simulation



Sinks and sources



Sensor model



X axis readout

Figure 2.4.: Analog simulation with Cadence.

2.14. AMS Closed-Loop Simulation

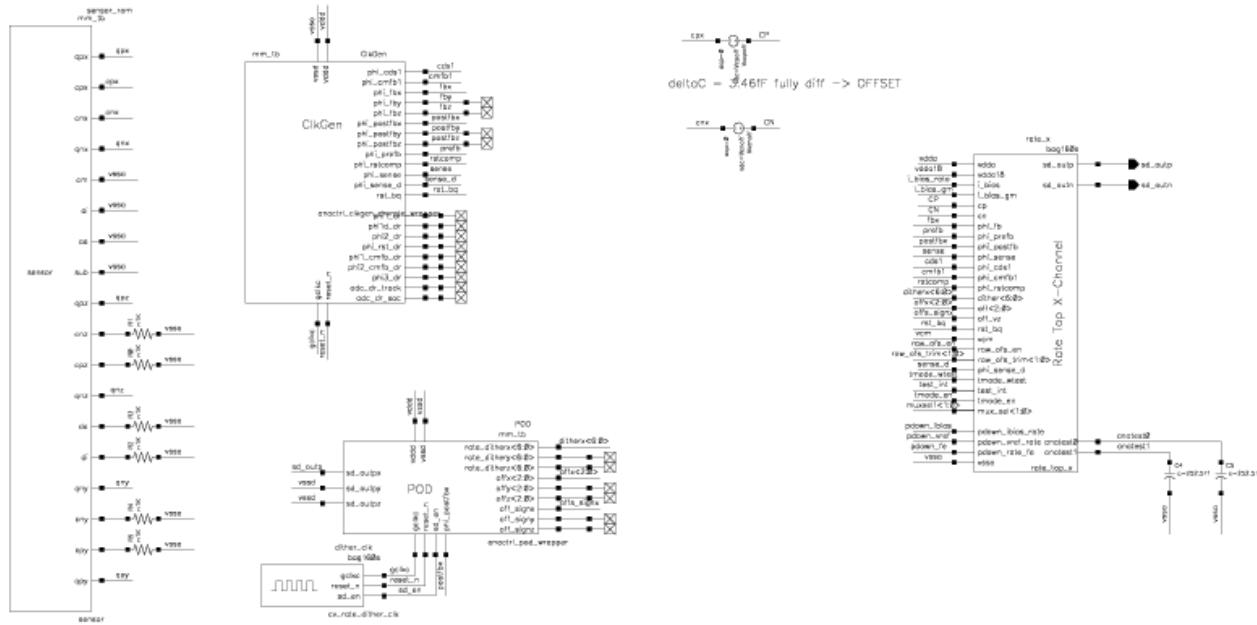


Figure 2.5.: AMS simulation with Cadence.

2.15. Conclusion

Vibratory Gyroscopes.

- Automotive and consumer applications
- System architectures
- Mixed-domain, mixed-mode analysis
- Interdisciplinarity → IC Systems Engineering Control theory, signal theory, process technology and micromechanics

More DOF's.

- Acceleration and angular rate (6 DOF's)
- Angulare rate and magneto sensors (6 DOF's)
- Acceleration, angular and magneto (9 DOF's)

Part II.

Lab

3. MBSE and Design of an Inertial Sensor System

3.1. Design Project

- System level, behavioural model
 - Matlab/Simulink,
 - Python
 - HDL (Verilog-ams, VHDL-AMS)
- Circuit level, SPICE with behavioural blocks, e.g. OTA and comparator
- PCB level
 - [ESP8266 NodeMCU](#),
 - [TIs ADS1115](#),
 - [ADs ADXL335](#)
- IC level

3.2. Design Project Flow

- Literature research in journals, professional (serious) internet forums (e.g. application notes of semiconductor companies) and library
- Set-up bibliography, e.g. [JabRef](#), [Citavi](#)
- Concept of your system
 - Partitioning
 - Functions
 - Work packages
- Design, implementation and validation
 - Mathematical description, e.g. Matlab/Simulink model
 - SPICE modeling and simulation, LTspice circuit
 - Data analysis and validation, Serial monitor

