TECHNISCHE
UNIVERSITÄT
DARMSTADT

Racyics

Industriekolloquium 2023 Datentechnik

Herausforderungen der Mikroelektronik im 21. Jahrhundert

From specification to the System on Chip

Design flow of a SoC IC

Thilo Wagner, Racyics GmbH

2023/July/10

About me

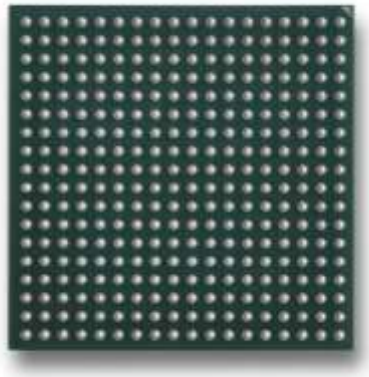
- Name: Thilo Wagner
- 2004 Diploma Electrical Engineering and Information Technology at TU Darmstadt
- 18 years SoC experience
- Head of Design for Testing (DFT) at Racyics
- http://linkedin.com/in/thilo-wagner
- http://www.racyics.de

General Context: ICs…

- ICs are found almost everywhere in technological products
- As an example, a phone has around 160 microchips
- IC handle tasks such as
  - Control, Data acquisition, Communication, Processing
- ICs can essentially act as “brains” of these products
- The most sophisticated ones comes in the forms of
  - **FPGAs**
  - **ASICs**
- Let’s have a closer look on these two..

General Context: FPGA

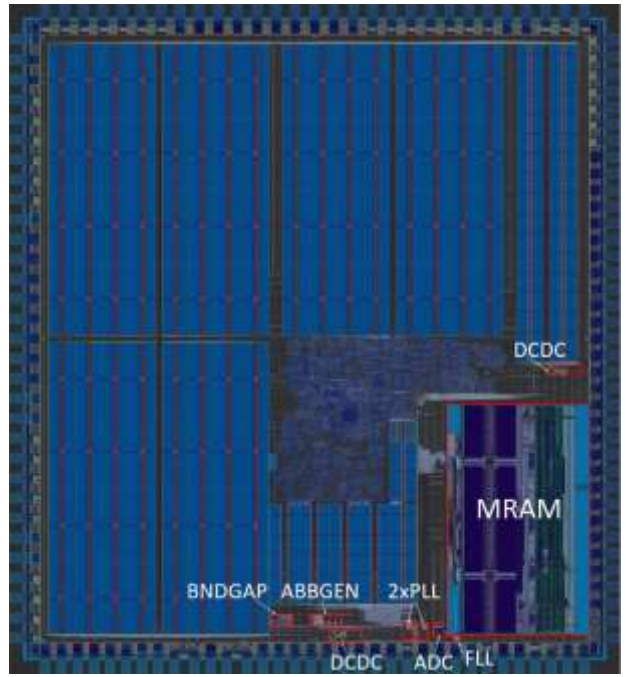
- FPGA stands for **F**ield-**p**rogrammable **g**ate **a**rray
  - Which is an IC which hardware functionality can be changed after manufacturing
  - It has configurable logic blocks that can be programmed, reprogrammed and rewired at will
    - FPGA => is an **adaptive hardware**
    - Hardware function of the microchip defined/redefined by the user
  - Programmed through HDLs (Hardware description languages)
    - Verilog, VHDL, Lucid
  - Can implement several **digital** tasks (control, data processing, memory, ..)
  - Integrates predefined Analog/Digital blocks
    - ADC, DAC, Communication interfaces
    - Even CPU, GPU (SoC-FPGA)



https://colognechip.com/programmable-logic/gatemate/

General Context: ASICs

- ASIC stands for Application-Specific Integrated Circuit
  - Which is an IC targeting the implementation of a defined function
  - Contrary to the FPGA, the hardware of an ASIC is fixed after manufacturing
  - Fitted for product permanent applications
  - Its digital functions are programmed through HDLs (Hardware description languages)
    - Verilog, VHDL, Lucid
  - Yet, an ASIC can embed both digital and analog functions
    - => Analog Mixed-Signal design (AMS)
  - ASIC designs can be fully customized
- ASSP: Application-Specific Standard Product
- **In this lecture I will mainly talk about ASICs**

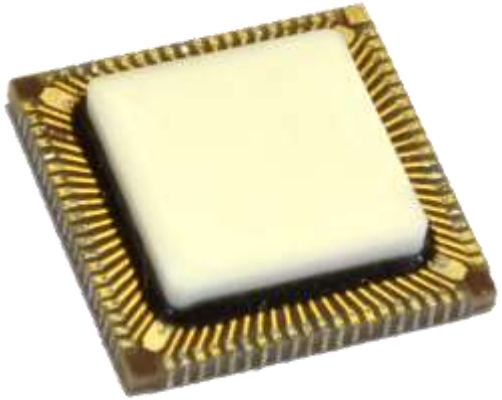


General Context: FPGAs vs ASICs

	FPGA	ASIC
Reconfigurability	✓	
Performance		✓
Frequency		✓
Size		✓
Power consumption		✓
Cost (low-volume)	✓	
Cost (high-volume)		✓
Time-to-Market	✓	
Design prototyping	✓	
Analog		✓

General Context: SoC

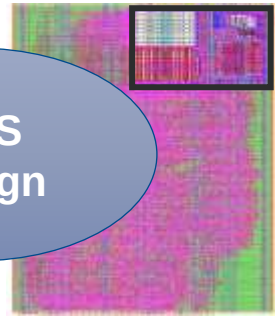
- SoC stands for **System-on-chip**
  - Which is a chip that incorporates multiple electronic components, such as microprocessors, memory units, input/output interfaces, and specialized hardware modules, onto a single chip.
  - A SoC can be compared to a PC motherboard embedding components such as CPU, GPU, RAM/ROM, Sensors etc… on the same board
  - Even though the borders are floating, SoCs are usually done as ASICs



From Spec to Chip – high level overview

Specification

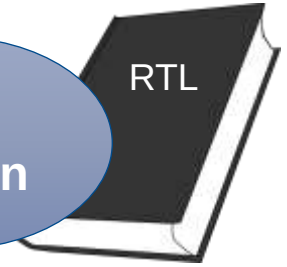
AMS Design



Custom Mixed Signal Block (Abstracts)

+

SOC Design

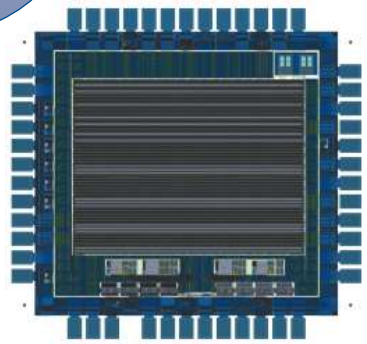


Logic description (hardware related code)

RTL2GDS

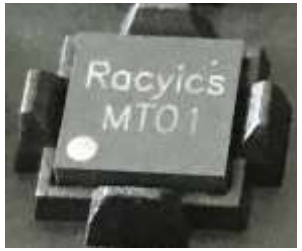
Physical Implementation

DFT



Geometric shapes

PRODUCTION



Real silicon in package

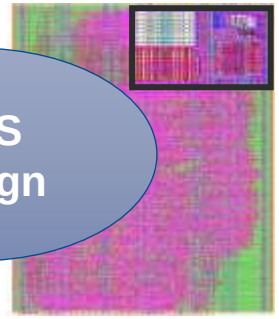
Testing & Validation

Package Design

Application and Product Engineering

From Spec to Chip – SoC Design

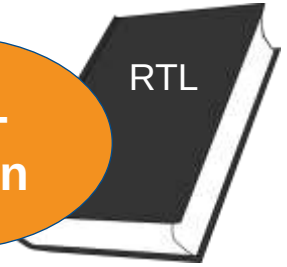
Specification



AMS Design

Custom Mixed Signal Block (Abstracts)

+



RTL

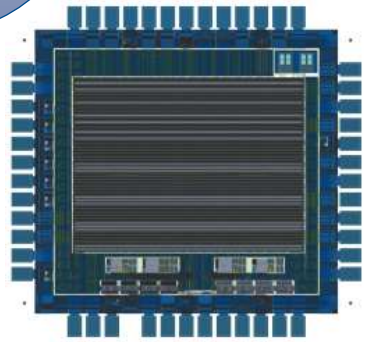
SOC-Design

Logic description (hardware related code)

Physical Implementation

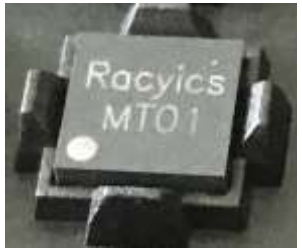
RTL2GDS

DFT



Geometric shapes

PRODUCTION



Real silicon in package

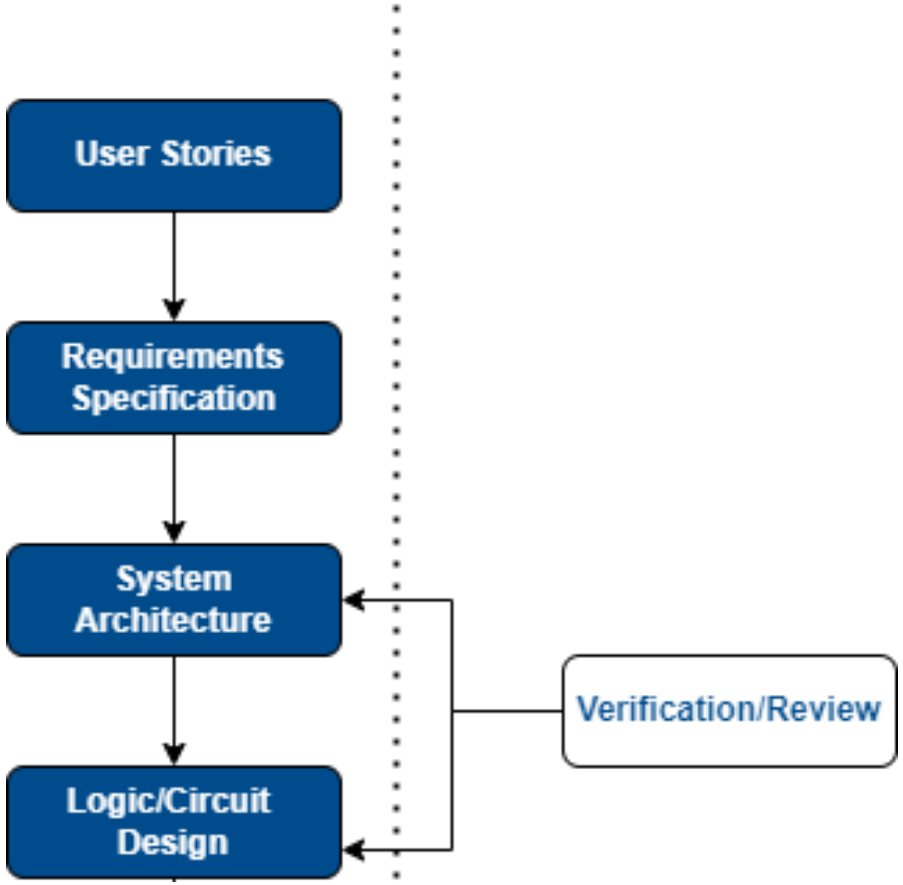
Testing & Validation

Package Design

Application and Product Engineering

SoC Design – from Spec to RTL

- User stories & elicitations
  - Input for Requirements Engineering & Specification
- Requirements Specification
  - Based on the user stories and design targets
- Architecture definition
  - Consistent with the targeted speed, power consumption, chip cost
- Logic/circuit design
  - Functional RTL design & verification



RTL Coding

- RTL = Register Transfer Level
- RTL coding is a high-level abstraction technique used in digital circuit design.
- It involves describing the behavior and functionality of a digital circuit in a Hardware Description Language (HDL).
- RTL serves as an intermediate representation between the design specifications and the physical implementation of the circuit.
- Typical HDLs are Verilog or VHDL

RTL Coding

- Small Verilog Example

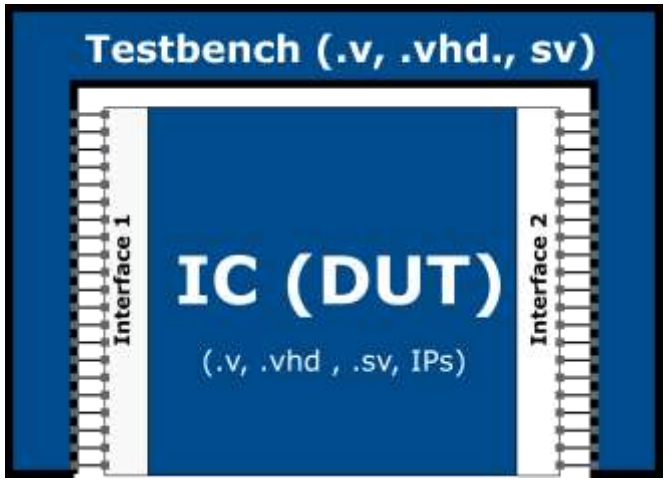
```
module spi_interface (
  input wire spi_clk,       // SPI clock
  input wire spi_cs,        // Chip select
  input wire spi_mosi,      // Master out, slave in
  output wire spi_miso      // Master in, slave out
);

  reg [7:0] spi_data_out;   // Data to be transmitted
  reg [7:0] spi_data_in;    // Received data

  reg spi_shift_reg;        // Shift register for data transfer
  reg spi_tx_complete;      // Transmission complete flag

  always @(posedge spi_clk) begin
    if (spi_cs == 0) begin   // Chip select is active low
      spi_shift_reg <= 1'b0; // Reset shift register
      spi_tx_complete <= 1'b0;
    end
    else begin
      spi_shift_reg <= spi_shift_reg << 1;  // Shift left by 1 bit
      spi_shift_reg[0] <= spi_mosi;          // Assign input to LSB
      spi_data_in <= spi_shift_reg;          // Assign input to received
```

Verifying SoCs : Functional Verification

- Functional Verification
  - Verify the functional correctness of the SoC
  - All functional requirements are met?
- Static Functional Verification
  - The design is not run/executed, static methods used
  - Among the types of static functional verification techniques
    - Clock-domain-crossing verification
    - Reset-domain-crossing verification
    - Linting
    - Formal verification

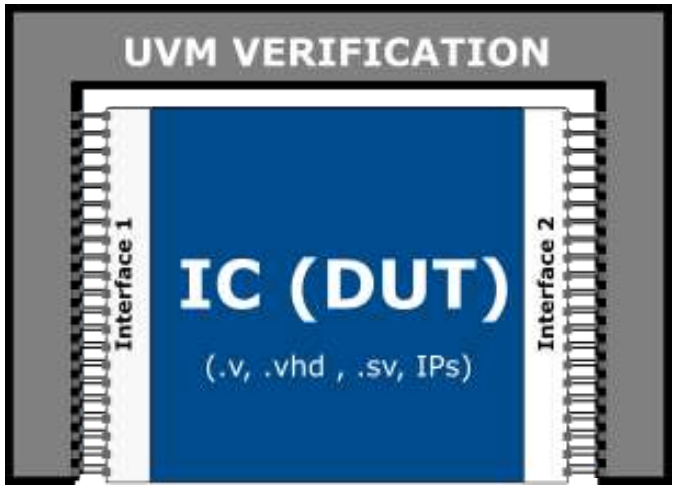
Verifying SoCs : Functional Verification

- Simulation based functional Verification
  - Verify the function through simulation of the SoC
  - Integrate the SoC (DUT, Device Under Test) in a specific testbench
  - The testbench is generally written in Verilog, VHDL, or Systemverilog
  - Methods like UVM or OVM are used



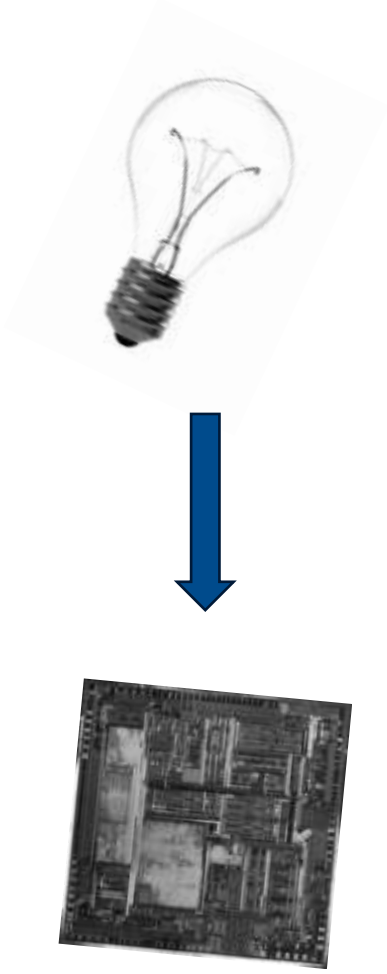
Verifying SoCs : UVM Verification

- Simulation based functional Verification
  - Each engineer can write his custom testbench
  - What about portability and re-usability?
  - => UVM (Universal Verification Methodology)
    - Enable faster development of functional verification environment
    - Re-usability of functional verification environment
    - Developed using Systemverilog and derived from OVM (Open Verification Methodology)
    - Set of verification libraries
    - => IEEE standard backed by the big players of the field



Job Profile RTL Design & Verification

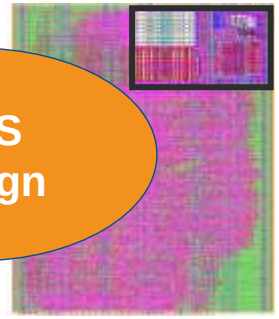
- Input
  - Requirements & Specification
- Output
  - Verified and synthesizable RTL code + testbenches
- Tasks & skills in RTL Design
  - Have an idea of an application or design
  - **Architecture design skills**
  - **HDL coding skills** (VHDL, Verilog, Systemverilog)
  - Further complete the steps of the design/development process
    - Simulation, synthesis, verification ….
  - Significant time and effort in RTL design goes into verification
  - Having RTL design and coding skills is a good starting point



From Spec to Chip – AMS Design

Specification

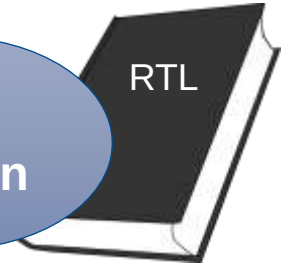
AMS Design



Custom Mixed Signal Block (Abstracts)

+

SOC Design

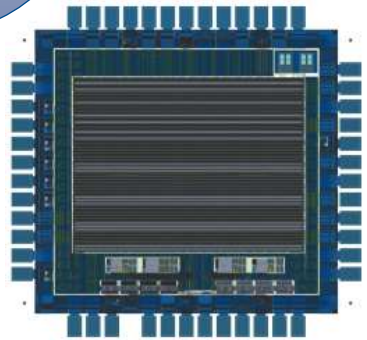


Logic description (hardware related code)

RTL2GDS

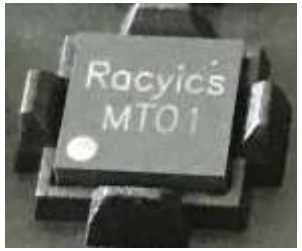
Physical Implementation

DFT



Geometric shapes

PRODUCTION



Real silicon in package

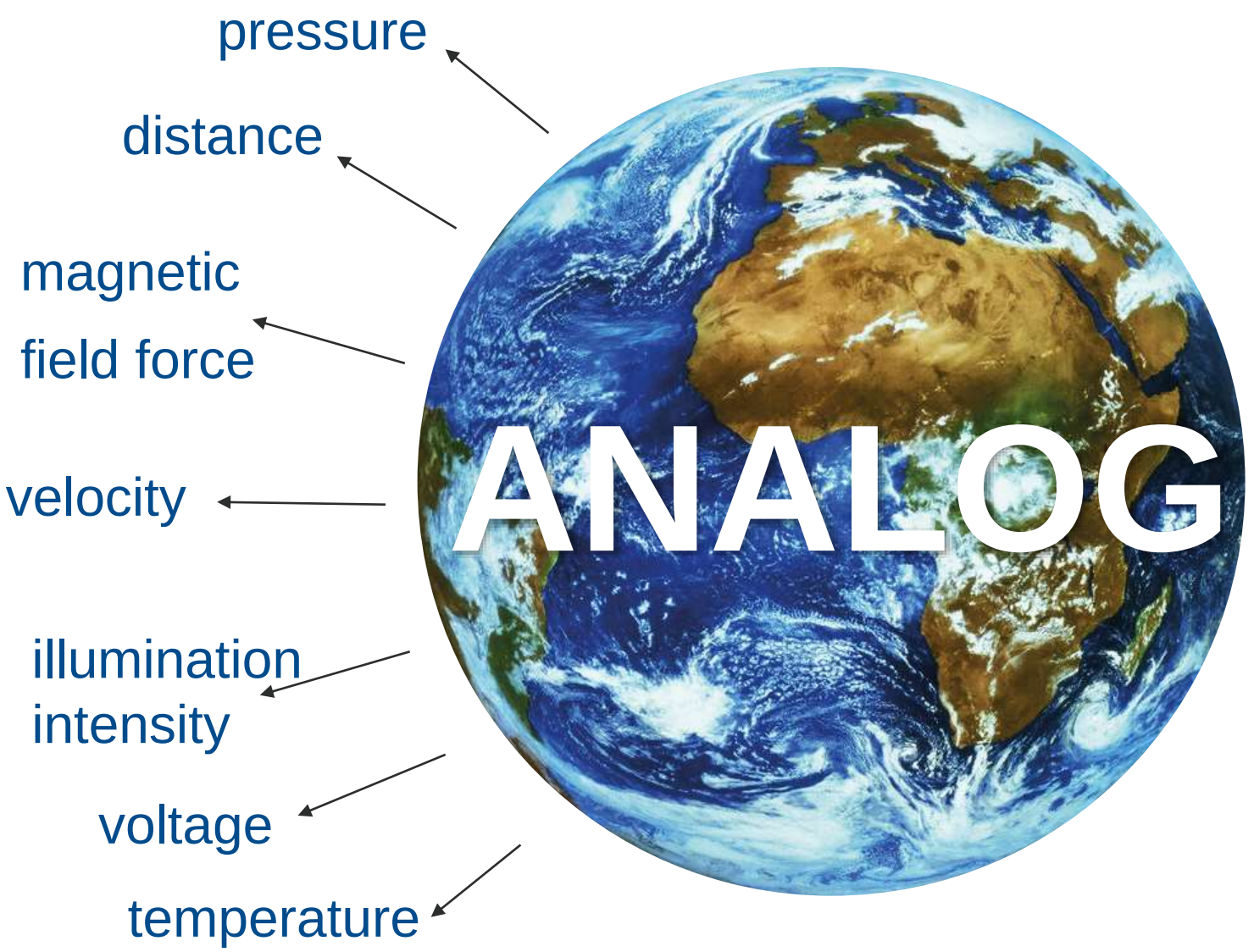
Testing & Validation

Package Design

Application and Product Engineering

Analog Design

- SoCs usually also need analog components
- Typical examples are PLL for internal clock generation, Analog-Digital and Digital-Analog Converters (ADC & DAC) etc.



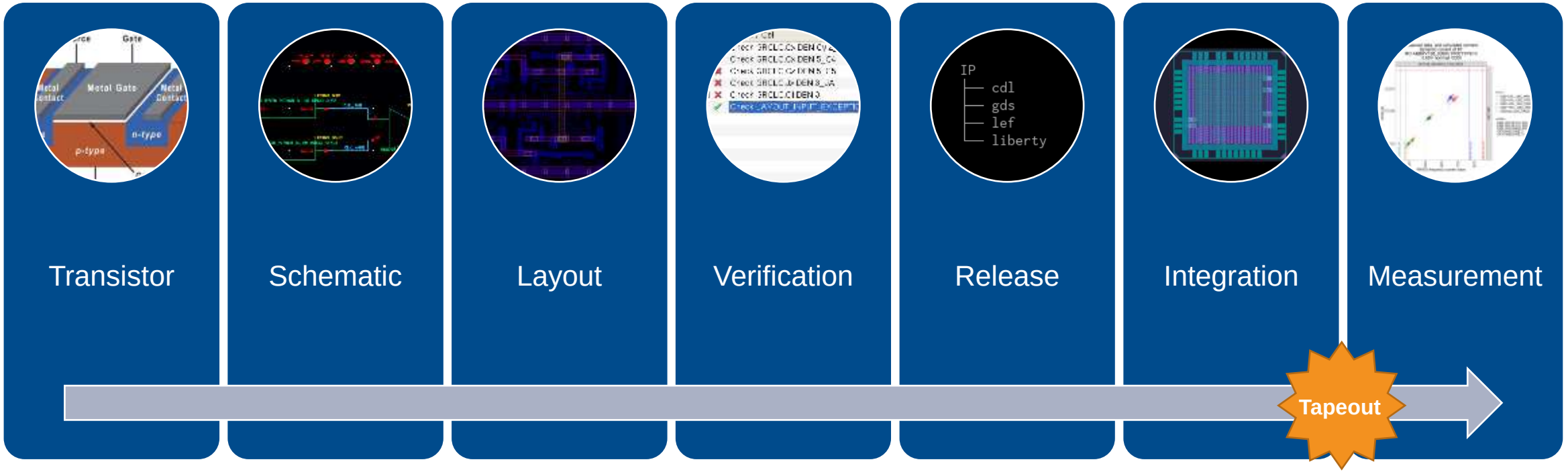
Analog vs. Digital Design

- Analog and digital semiconductor design employ different design methodologies.
- Analog design typically involves a more iterative and manual process, often requiring custom circuit design and optimization for specific performance requirements ➔ Full Custom Design
- On the other hand, digital design is often driven by hardware description languages (HDL) allowing for a more modular and automated design process.

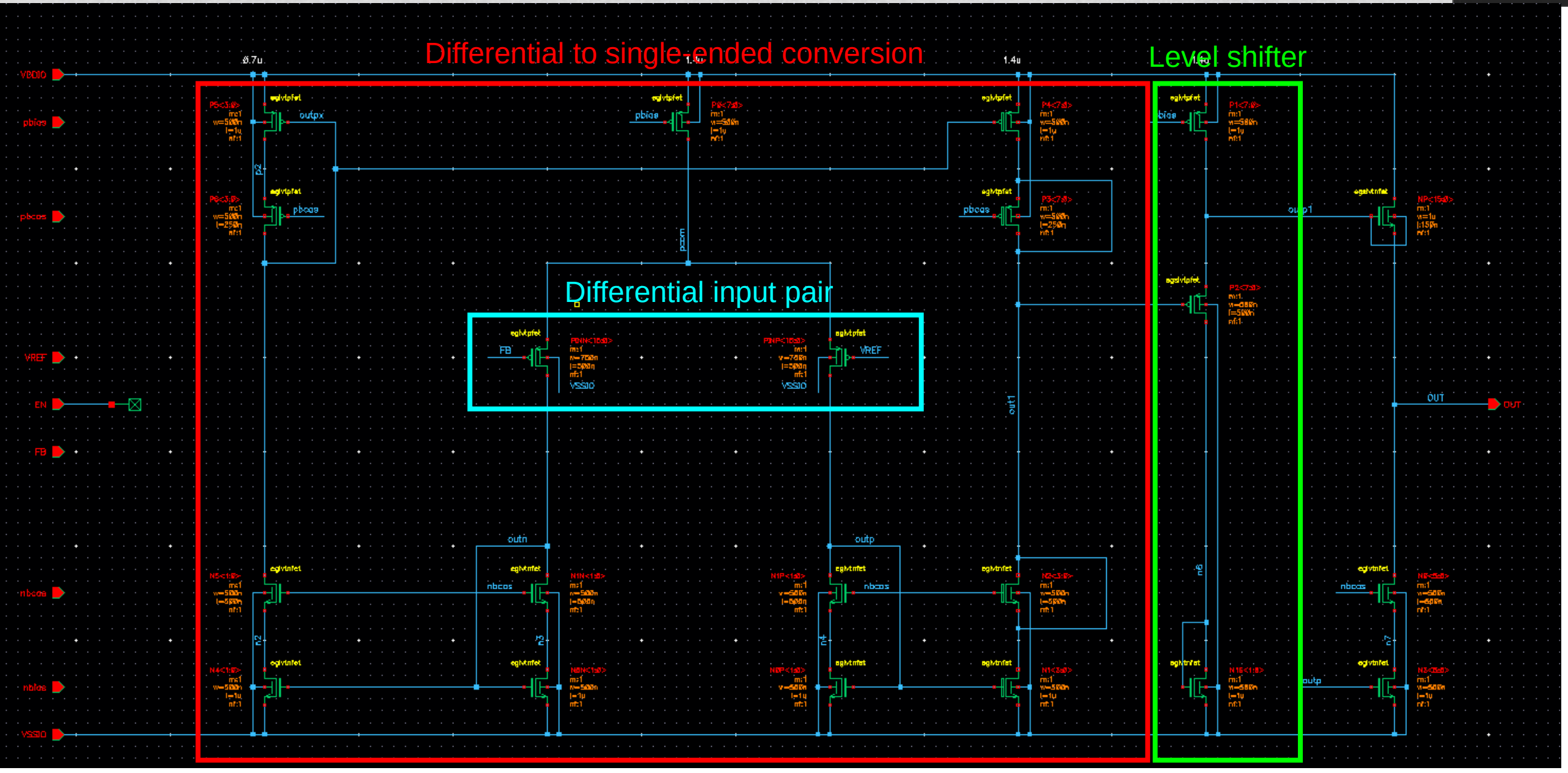
Analog Mixed Signal Design

- Analog Mixed-Signal (AMS) design involves the integration of both analog and digital circuitry into one module (often called IP – Intellectual Property)
- AMS design enables the implementation of complex systems that require both analog and digital functionalities, such as data converters, sensor interfaces, and wireless communication systems.
- The digital part of AMS design is usually rather small, but can still be challenging to implement due to strict timing requirements on the analog-digital interfaces

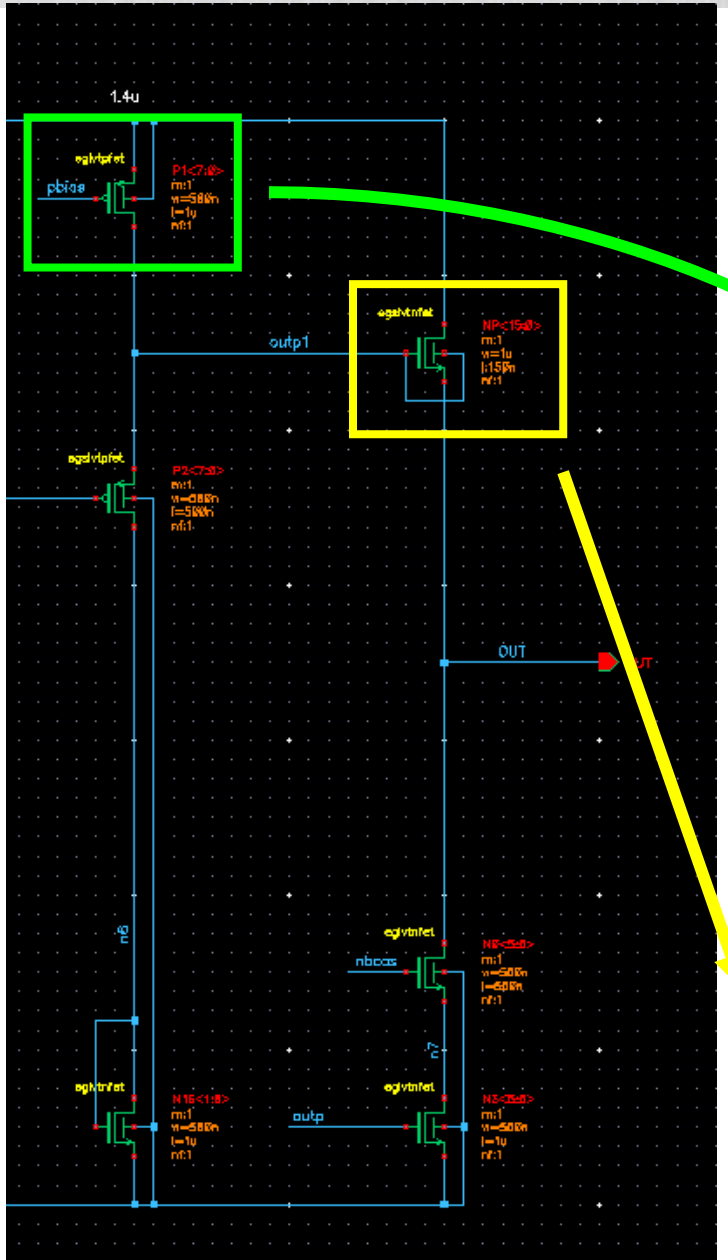
AMS Development Process

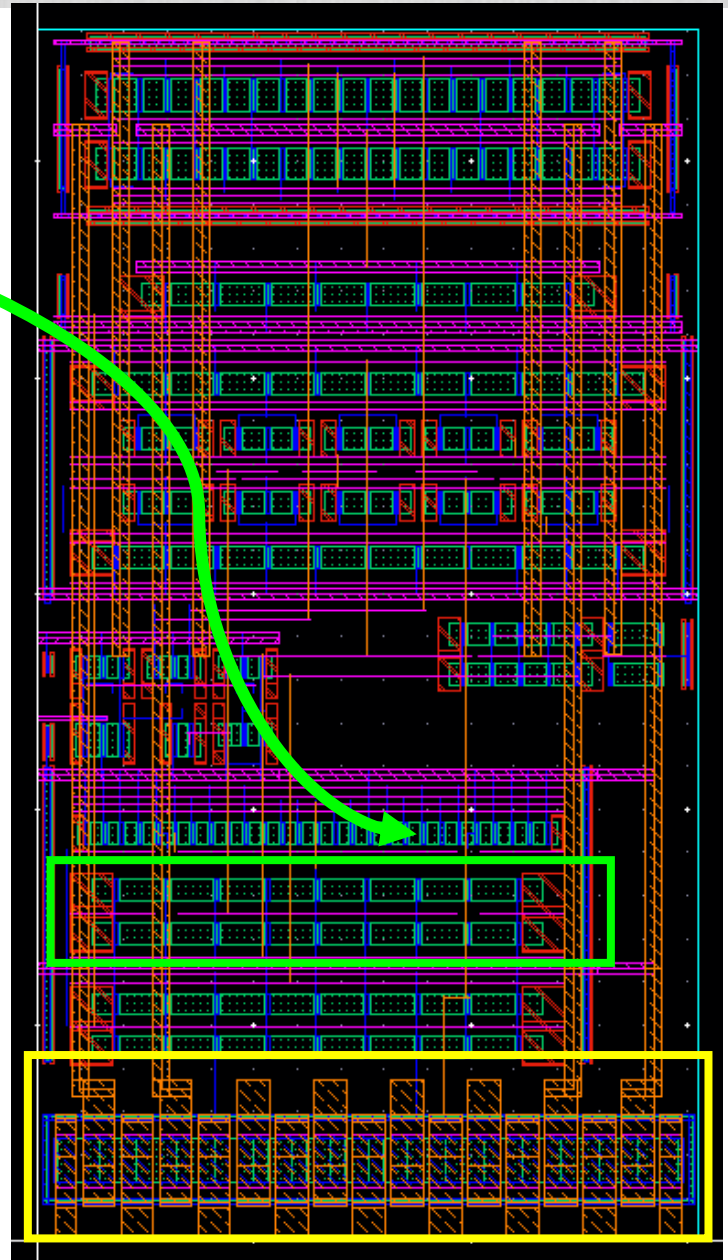


AMS Schematic



AMS Layout





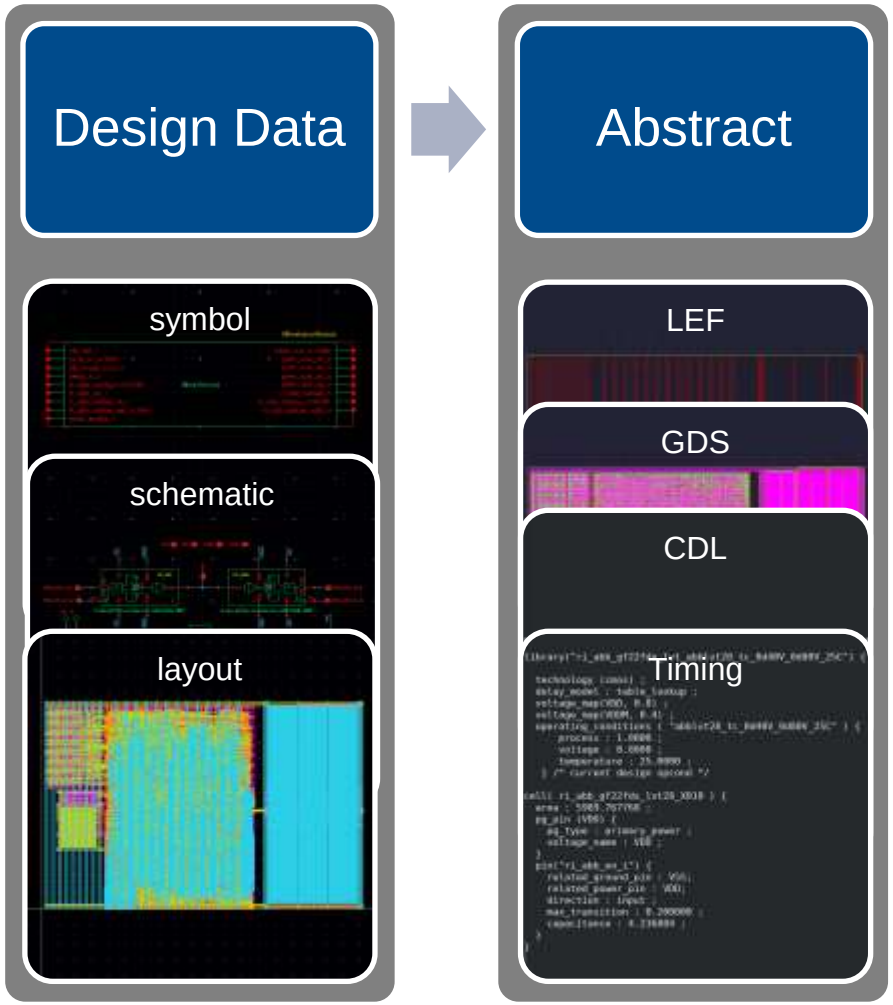
- **Placing** transistors and analog components according to the schematic
- **Connecting** them
  - Wires must be thick enough to carry the current
  - Technology rules must be followed

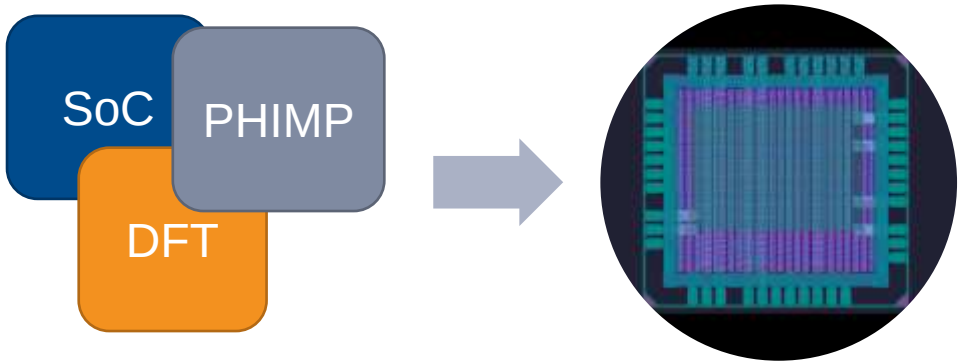
AMS Verification

- Verifying if the circuit performs the intended function for all possible conditions
  - cold and hot temperatures
  - high and low supply voltages
  - with manufacturing variation
  - ..
- Iterating back to schematic and layout if the requirements are not met

Output	Nominal	Spec	Weight	Pass/Fail
Filter	Filter	Filter	Filter	Filter
Phase Margin	72.61 deg	> 60		pass
Gain Margin	15.93	info		
Closed Loop Gain				
Open Loop Phase				
Open Loop Gain dB20				
Output noise; V / sqrt(Hz)				
Open Loop Gain at 100Hz	44.56 dB	> 45		near
Vout				
UGF	17.1 MHz	< 5M		fail
BW 3dB	99.52 KHz	> 250		pass
Idc	13.39u	info		
Power	10.71 uW	< 500u		pass
vbias	523.2 mV	tol 0.52 5%		pass

AMS Integration & Release





- Generate different views of the AMS circuit for the digital design flow
- Hand the views to the other teams to integrate the IP into the chip

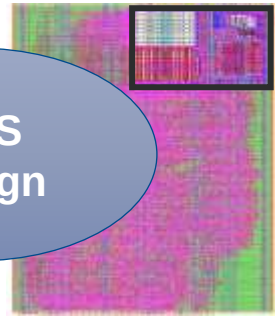
Job Profile AMS Design & Verification

- Input
  - Requirements & Specification
- Output
  - GDS (Mask Data) & Models of AMS IPs
- Tasks & skills in AMS Design
  - Strong knowledge of analog circuit design principles and device physics.
  - Proficiency with analog EDA tools for circuit design, analysis and optimization.
  - Familiarity with full custom layout techniques.
  - Understanding of digital interfaces and mixed-signal integration.
  - Ability to develop test plans and perform circuit verification and characterization.

From Spec to Chip – Design For Test (DFT)

Specification

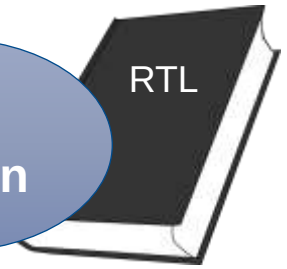
AMS Design



Custom Mixed Signal Block (Abstracts)

+

SOC Design

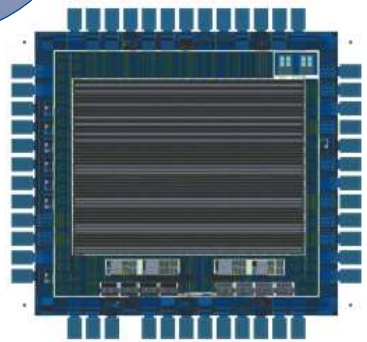


Logic description (hardware related code)

Physical Implementation

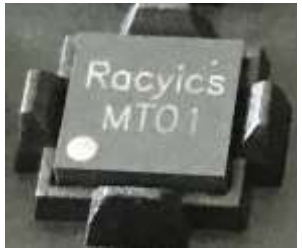
RTL2GDS

DFT



Geometric shapes

PRODUCTION



Real silicon in package

Testing & Validation

Package Design

Application and Product Engineering

DFT: Motivation

- Why do we need to test?
  - Chips can be faulty either due to defects during production or throughout their lifetime due to aging or external effects
- Not detecting defective chips during production would in best case cause significant higher costs, in worst case in a malfunction of a safety-critical application (automotive, avionics, medical,..)
- For such safety critical applications chip failures do not only need to be detected during production but throughout the entire life-time of the chip
- Question: who would like to drive in an autonomous driving car where the functionality of the chips was only tested 5 years ago during production?

Electromigration

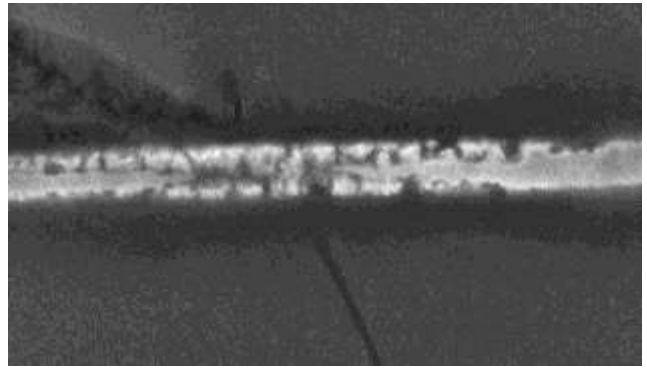
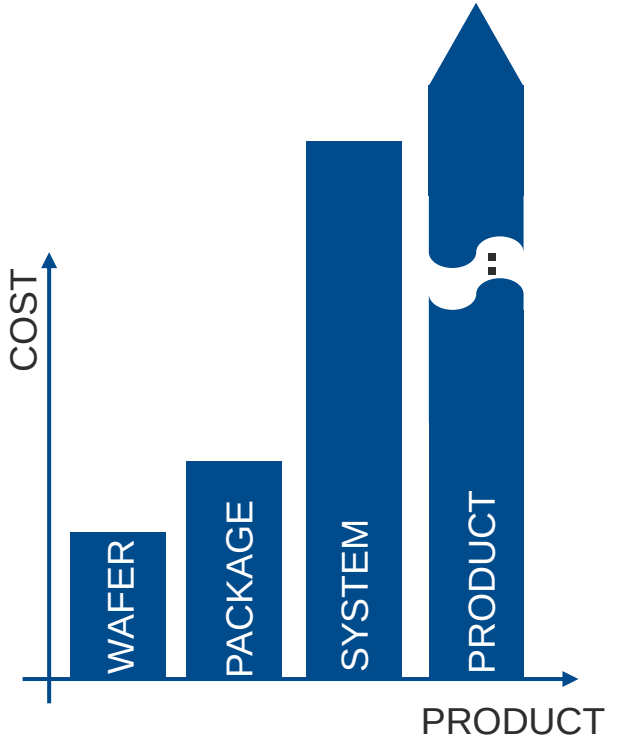


Photo: by Patrick-Emil Zörner, CC-SA



DFT: Functional vs. Structural Testing

- Ok, so chips might have defects – can't we just run some functional tests on them to find the bad ones..?
- Answer: NO, because..
- .. Today's chips contain many millions of transistors, contacts, wires, ... Testing all of them with functional test would either not be possible or take too much time
- .. high coverage of complex SoCs cannot be achieved purely via functional testing
- .. testing time is expensive

So, if functional testing alone can't do the job – what is the solution?

DFT: Structural Testing

- The solution is option 2: **Structural testing** instead of functional testing
- Functional testing
  - tests whether the design meets the specified functional requirements
  - requires detailed knowledge about the functionality of a design
  - focuses on the external behavior of the design, without considering its internal structure
- Structural testing
  - tests the internal structure of the design (transistors, vias, wires, …), rather than its external behavior
  - does not require detailed knowledge about the functionality of a design

DFT: Structural Testing

- So, the idea is:

  if you have **verified** once, that the function of your chip would be correct under the condition that is does not have any structural defects, you don‘t need to do functional testing anymore. Structural **testing** then can ensure the correct function of the chip.

DFT: Design For Test / Testability

- DFT is the process to add additional features to a chip to make it structural testable
- Usually this is done by adding a complete new “hidden” test operation mode on top of the functional / mission mode of a chip.
- After entering this test mode, the chip enables internal DFT logic and testing functionalities.
- Typical testing mechanisms are Scan Test, Logic Built-In Self Testing, Memory Built-In Self Testing, Boundary Scan ..

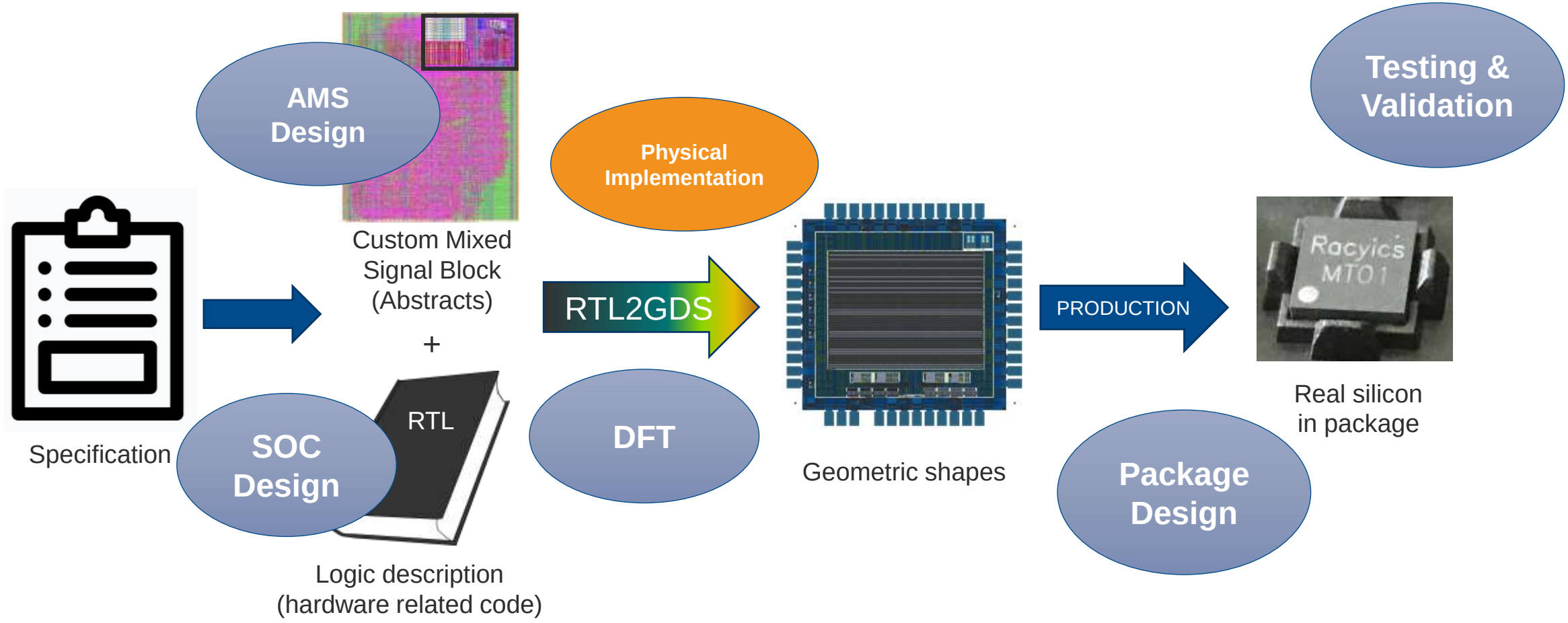
SoC RTL vs. DFT Design

- RTL design
  - concentrate on the functional mode of a chip
  - Coding in HDL
- DFT Design
  - Concentrate on the structural testing mode of a chip
  - Big part of the HDL code is generated by tools and not handwritten
  - Coding is mostly in TCL to write scripts that control the DFT tools
  - Finding and fixing testability issues in a design in close collaboration with the RTL design team
  - Advanced Test Pattern Generation (ATPG) & verification

Job Profile DFT Design & Verification

- Input
  - Functional RTL without DFT Logic + Testability Specification
- Output
  - RTL or Netlist with additional DFT structures + Constraints for additional DFT modes
- Tasks & skills in DFT Design
  - Good understanding of Digital Logic
  - Interest in Coding (mainly TCL)
  - DFT engineers need to collaborate a lot with other Teams, so a good understanding of the entire design flow is very helpful

From Spec to Chip – Physical Implementation



Application and Product Engineering

Physical Implementation

- Physical Implementation of a semiconductor refers to the process of transforming a digital circuit design into a physical layout that can be fabricated on a semiconductor wafer.
- It involves translating the RTL code into a gate level netlist and then creating a layout representation, specifying the exact locations and interconnections of all components on the chip.

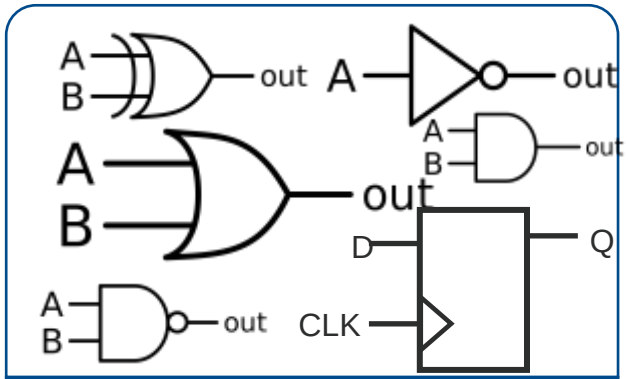
RTL to Gate-Level Synthesis

```
always @(posedge clk or negedge reset_q_i)
begin
    if (reset_q_i == 1'b0)  begin
        r_result          <=1'b0;
    end
    else begin
        r_result<=input0_i && input1_i;
    end
end
```

RTL

- Clock Frequency
- Input/Output Delays
- False paths
- Case Settings
- …

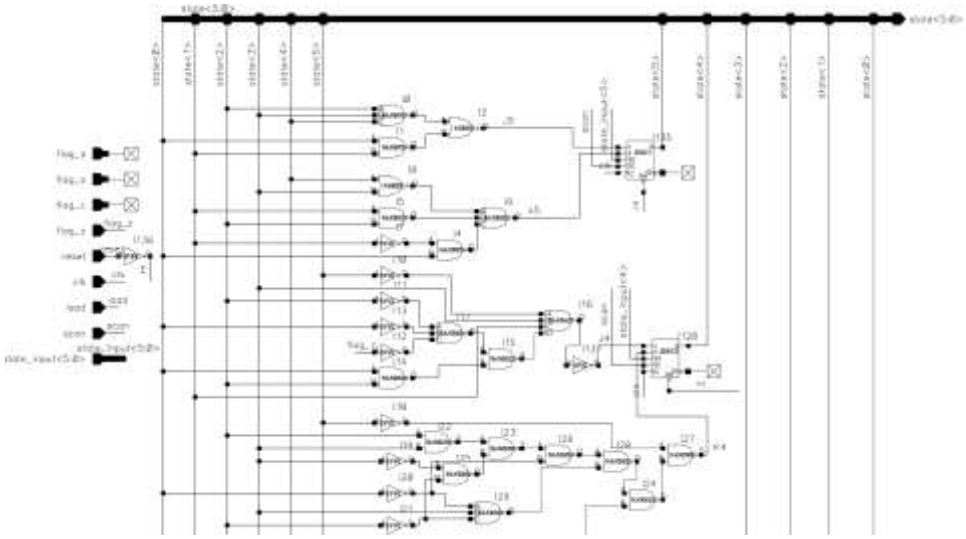
Constraints



Standardcell Lib
(+ AMS Libs)

Synthesis Tool

Netlist

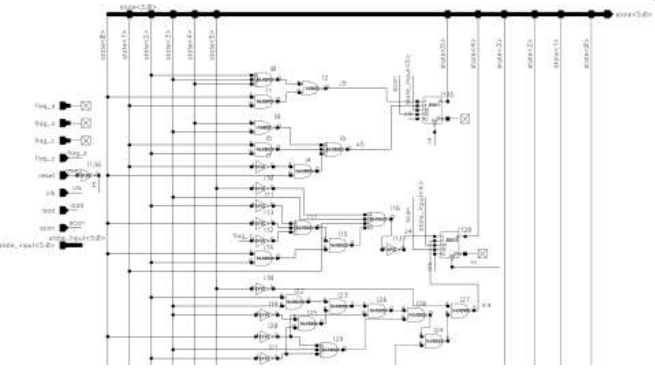


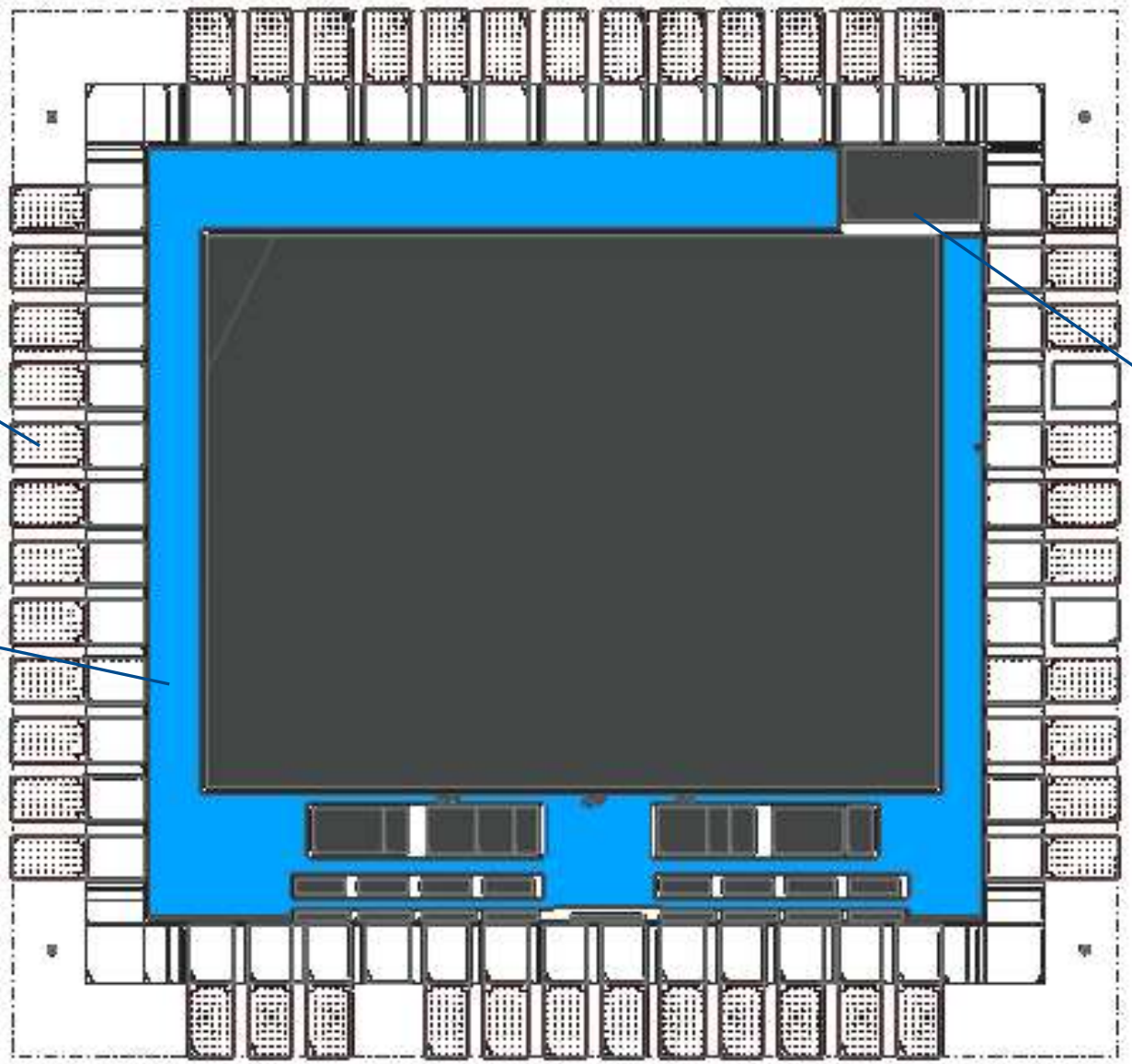
Floorplan

Interface
to outside

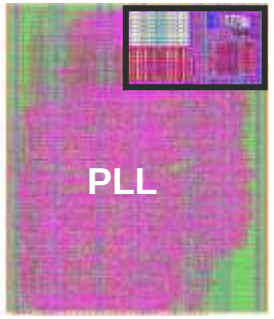
IO cells

Standard cell area



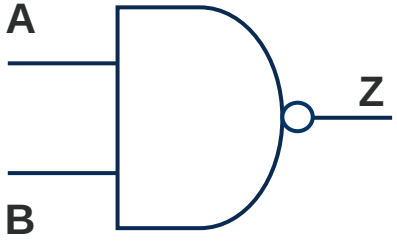


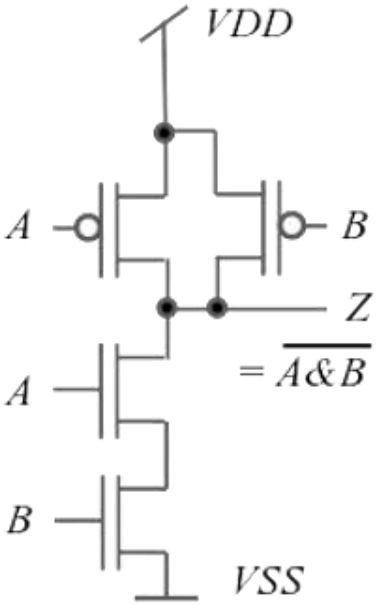
Macros



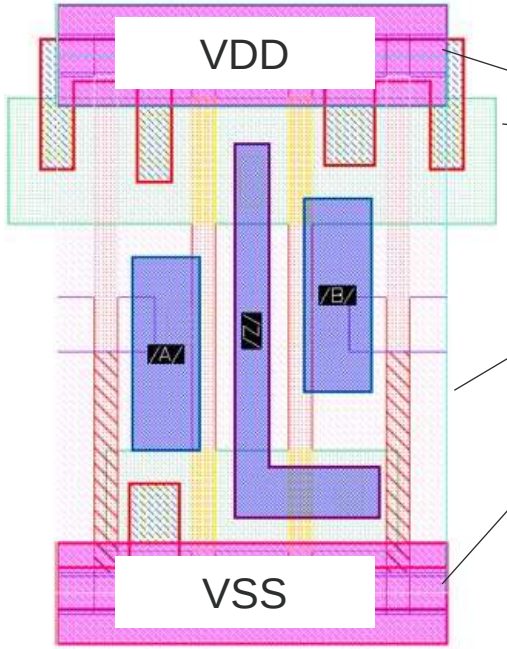
Standard cells and their rows

Example of a NAND gate:

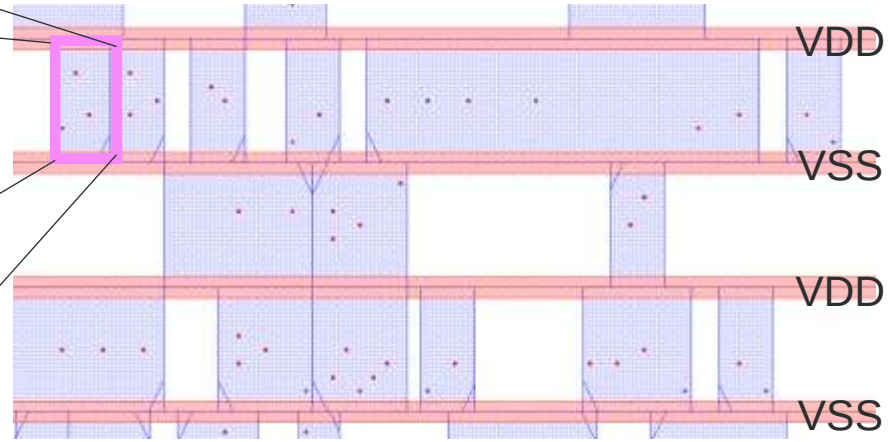




CMOS circuit



Layout



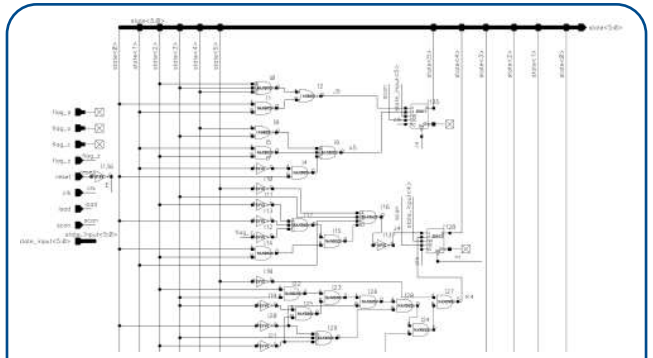
Rows are created during floorplanning to form rails for the placer tool

Placement of Standardcells in finished Floorplan

Floorplan with empty Standard Rows

- No overlapping
- Correct orientation
- Timing constraints
- …

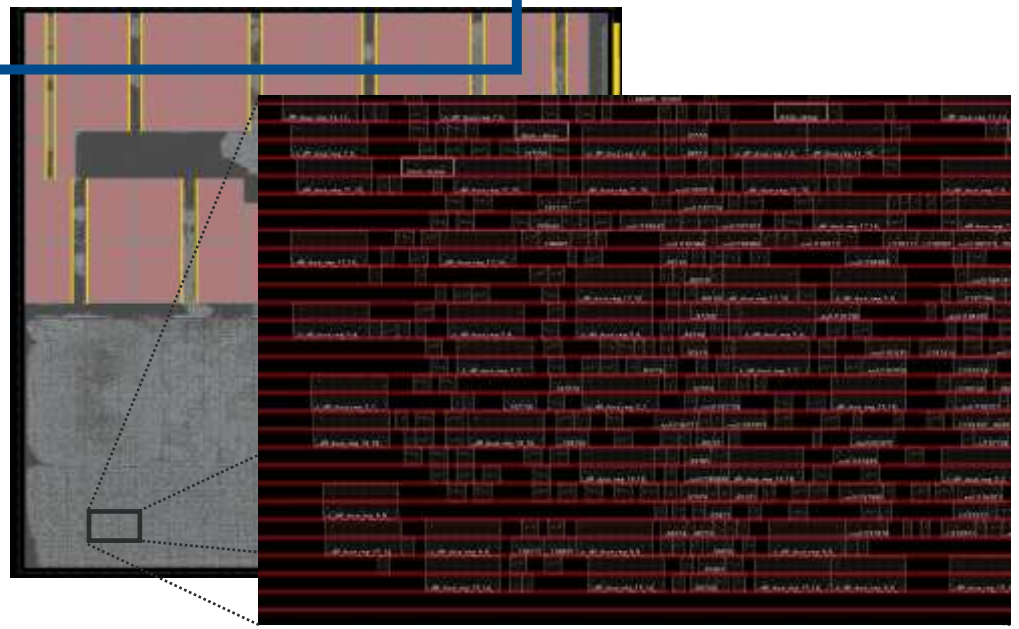
Constraints



Standard Cells and logical connections

P&R Tool

Floorplan + Placed Std.-Cells



Time for a Quiz!

How many standard cells are shown in the yellow area?

A: 100 - 1,000

B: 1,000 - 10,000

C: 10,000 - 100,000

D: >100,000

55 nm technology

Time for a Quiz!

How many standard cells are shown in the yellow area?

A: 100 - 1,000

B: 1,000 - 10,000

C: 10,000 - 100,000

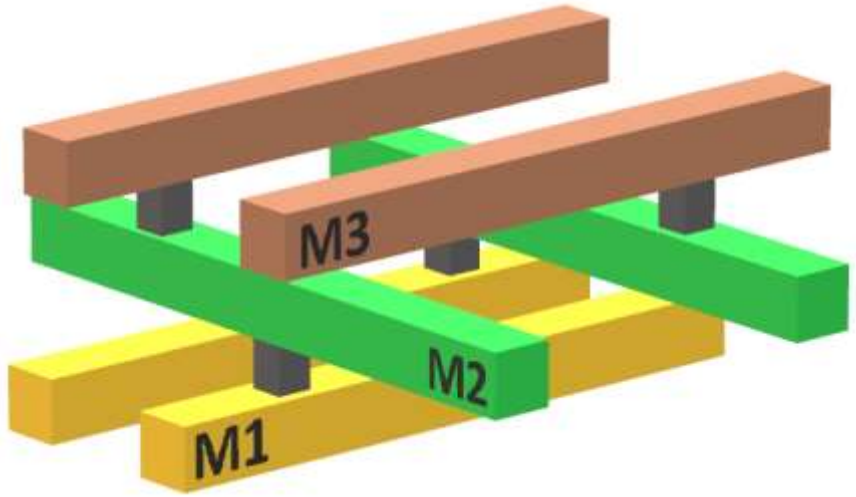
D: >100,000

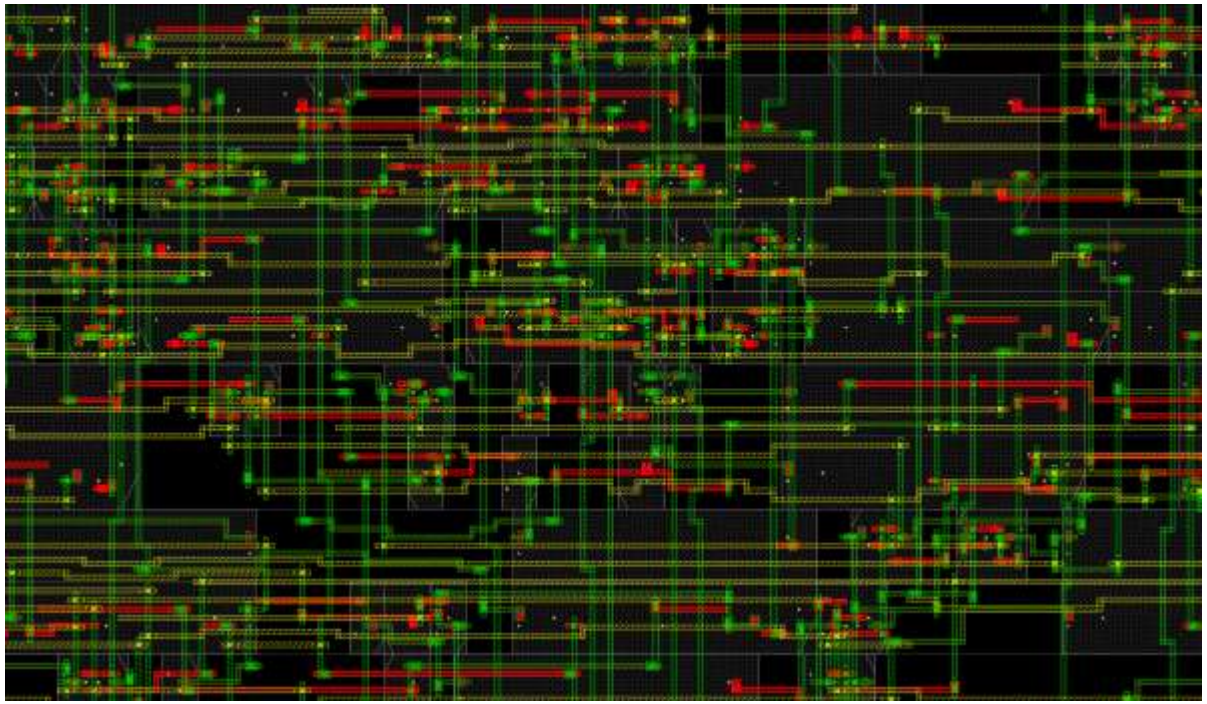
55 nm technology

Routing

Process to draw the logical connectivity of all cell pins in the circuit

- Semiconductor processes offer multiple metal layers that can be used to establish the connectivities
- Metal layers are
  - assigned to routing tracks for automated routing
  - alternating in preferred routing direction





Sign-Off

Synthesis

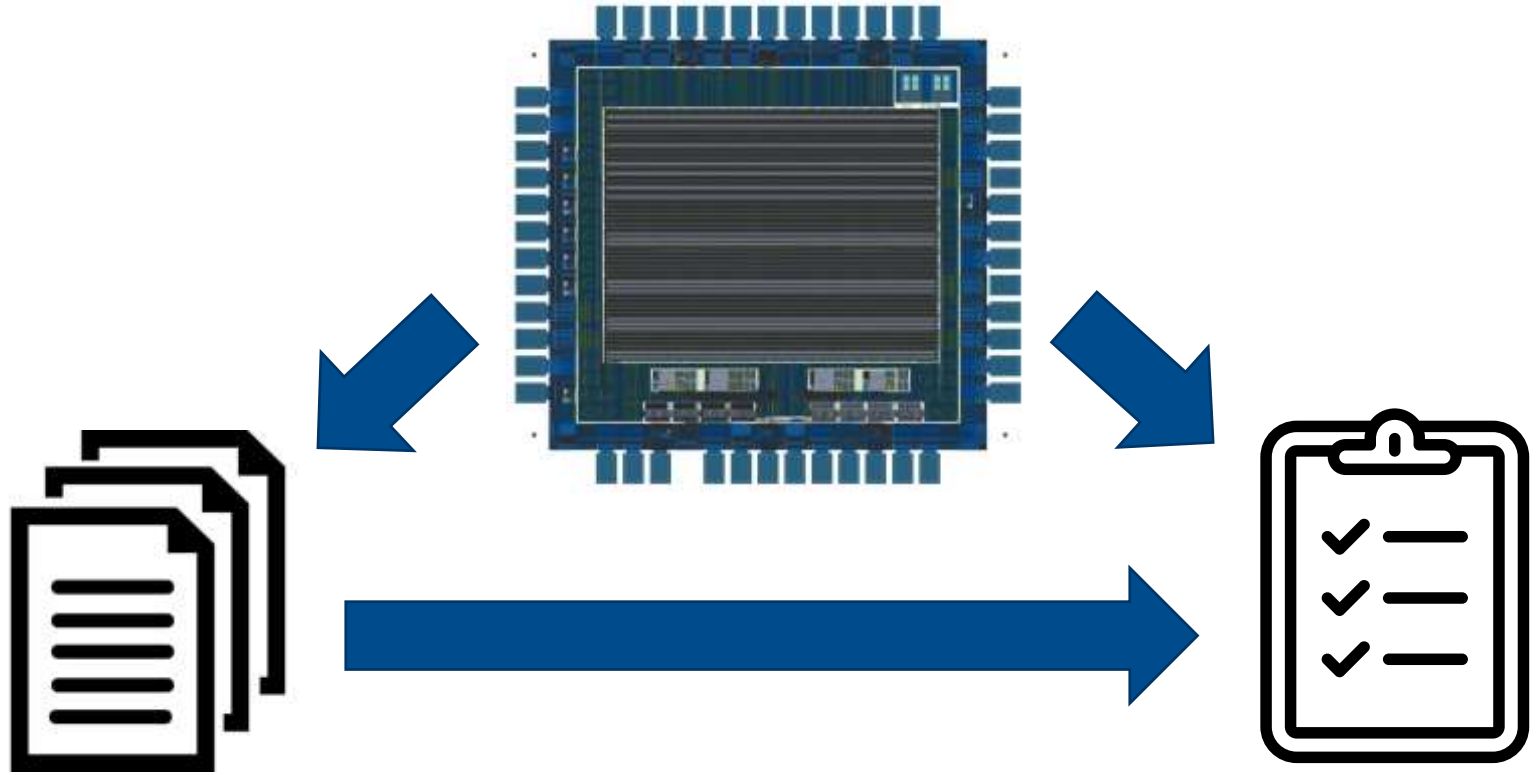
Floorplan

Place & Route

Sign-Off

Process to extract and calculate results

and to verify the correctness of the implementation

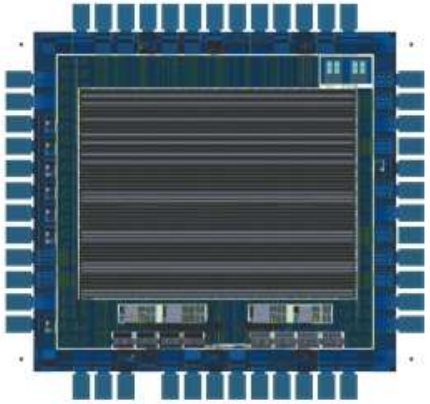


Sign-Off

- Logical and Functional checks:
  - Logical Equivalence Check (LEC)
  - Simulations
- Timing checks: Static-Timing-Analysis (STA)
- Physical checks:
  - Design Rule Checks (DRC)
  - Antenna
  - Electrical Rule Checks (ERC)
  - Layout-vs-Schematic (LVS)
- Power consumption calculation
- Power network checks:
  - Voltage-drop analysis and Electromigration checks (EMIR)

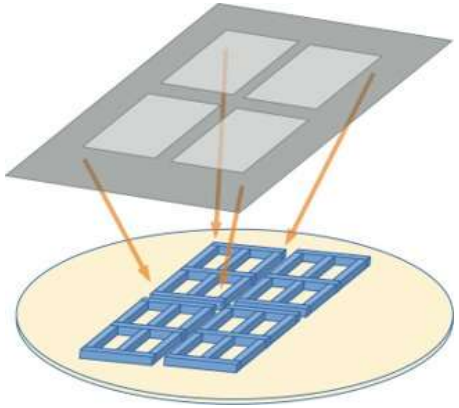
GDS to Fab

When all Sign-Off checks have passed



GO for manufacturing

"Tapeout"

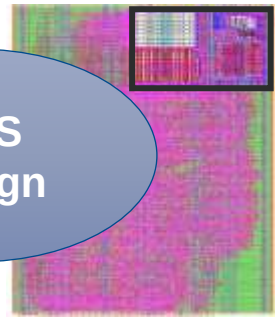


Job Profile Physical Implementation

- Input
  - RTL and Constraints (power, timing, ..)
- Output
  - Layout data for production (GDS or OASIS)
- Tasks & skills in Physical Implementation
  - Interest in Coding (mainly TCL)
  - Graphical design (Floorplanning)
  - Analytical thinking and problem solving skills
  - Working with big data (data science)

From Spec to Chip – Package Design

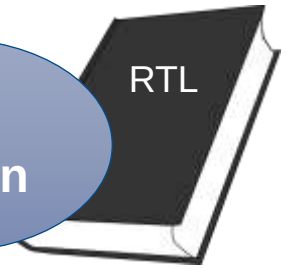
Specification



AMS Design

Custom Mixed Signal Block (Abstracts)

+



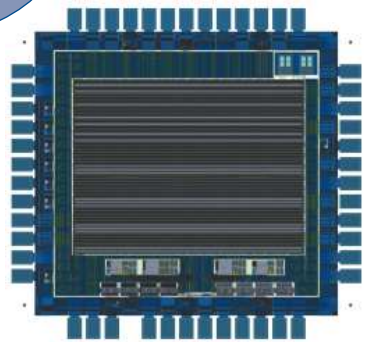
SOC Design

Logic description (hardware related code)

Physical Implementation

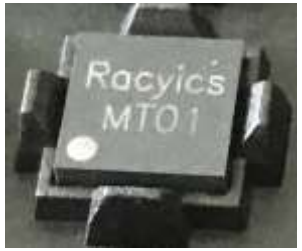
RTL2GDS

DFT



Geometric shapes

PRODUCTION



Real silicon in package

Testing & Validation

Package Design

Application and Product Engineering

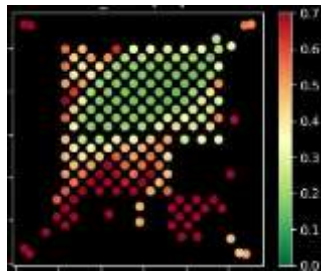
Package Design

Chip design

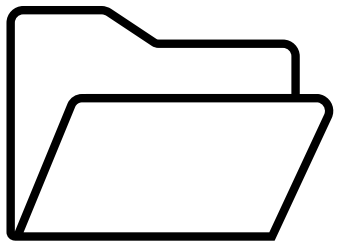
Tape Out

Package design

Layout



Verification



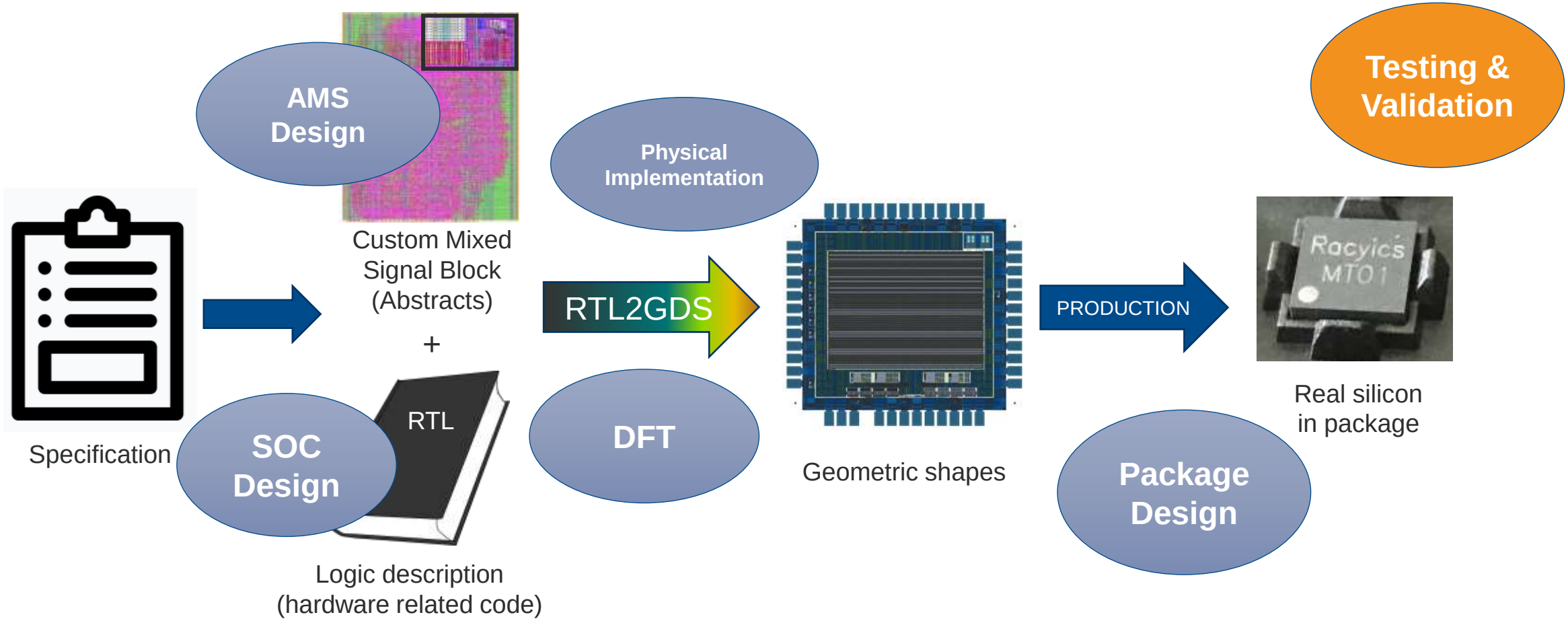
Production Data

Final product

Job Profile Package Design

- Input
  - Requirements & Specification
  - Die pad-out
- Output
  - Production Data (e.g., gerber, ODB++, GDS)
  - Simulation Models for System Verification, s-parameters, SPICE
- Tasks & skills in Package Design
  - Good understanding of analog basics
  - Signal & Power Integrity
  - Scripting skills (Python, ..)

From Spec to Chip – Testing & Validation

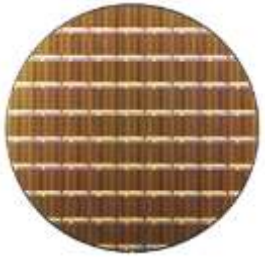


Application and Product Engineering

Testing & Validation

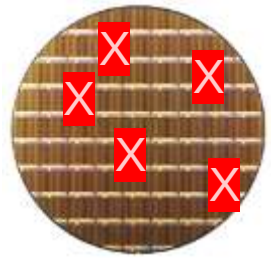
Blank wafers

processing steps



Patterned wafers

Die tester



Tested wafers

Dicer

Individual dies

Packaging

Packaged dies

Part tester

Tested dies

Ship to customer

Validation

Verification vs. testing vs. validation of ICs

- **Verification** aims to ensure the correctness of the chip's design and implementation
  (e.g. by running simulations before production)
- **Testing** of microchips focuses on evaluating their functionality, performance, and reliability through various tests.
  (e.g. by testing the produced dies)
- **Validation** is the process of checking if the end product has met the customers true needs and expectations

- Verification, testing and validation are essential to ensure the quality and functionality of microchips but are usually done in difference steps of the design process by different roles.

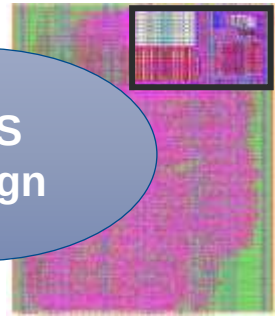
Job Profile Validation

- Input
  - Requirements & Specification
  - Packaged product
- Output
  - Validation report
  - Measurement data
- Tasks & skills in Validation
  - PCB Design knowledge
  - Understanding of testing and measurement techniques and equipment
  - Hardware-related programming (C, Python, ..)

From Spec to Chip – Application & Product Engineering

Specification

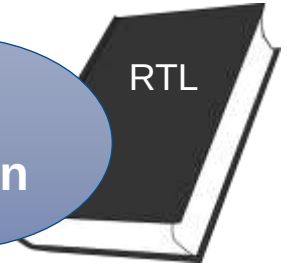
AMS Design



Custom Mixed Signal Block (Abstracts)

+

SOC Design

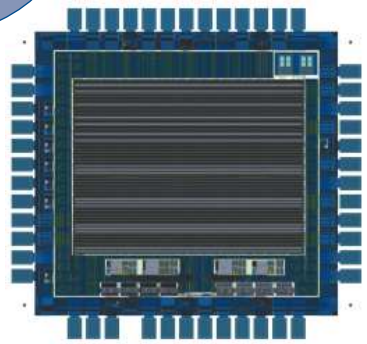


Logic description (hardware related code)

Physical Implementation

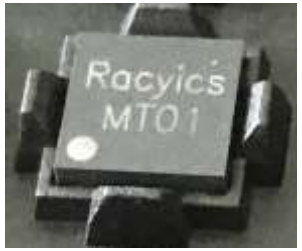
RTL2GDS

DFT



Geometric shapes

PRODUCTION



Real silicon in package

Test & Validation

Package Design

Application and Product Engineering

Job Profile Application & Product Engineering

- Input
  - DUT works on the bench, but failures at customer
  - (Customer) System level requirements
  - Field returns
- Output
  - Working chip
  - Stable mass-production
- Tasks & skills in Application & Product Engineering
  - Experience in “what can go wrong?”
  - System understanding (up to Layer 5/6)
- Tasks & skills in Product Engineering
  - Passion to measure in all different ways, incl. Production Tester or OBIRCH

Thank you very much for listening!

LEADING EUROPEAN ADVANCED NODE SOC/ASIC DESIGN CENTER

Racyics is an experienced System on Chip design service provider founded in Dresden in 2009. We provide many different services, including

- **SoC DESIGN SERVICE** with support for all the major design tasks
- **CUSTOM IP** development for Ultra-low power mixed-signal IP in advanced nodes
- **IP LICENSING** for our world record energy efficient 22FDX® ABX platform for Ultra-Low Voltage implementations down to 0.4V

www.racyics.com/jobs

LEADING EUROPEAN ADVANCED NODE SOC/ASIC DESIGN CENTER

Racyics offers various benefits, including

- **Childcare allowance** for your kindergarten kids
- **Flextime and Mobile Work** to help you maintaining your work life balance
- **Close Mentoring** to support the start of your professional career in the best possible way
- **Interdisciplinary Work and Close University Cooperation** to broaden your knowledge after graduation
- **Great Team Events!**

www.racyics.com/jobs

LEADING EUROPEAN ADVANCED NODE SOC/ASIC DESIGN CENTER

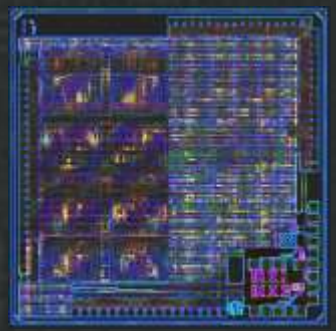
We offer exciting **student positions** *(Internship, Working Student, Thesis Student)* and **job entries** for electrical engineering, information technology, computer science and physics students in

- Digital Design and Verification
- Design for Test
- Physical Design
- Analog & Mixed Signal Design
- Package/PCB Design & Validation
- Application & Product Engineering
- Firmware & Software Development

... and more

World Record Low Power IP for 22FDX

Inhouse Lab & Silicon Validation



Working in Advanced Nodes 10/14/22nm

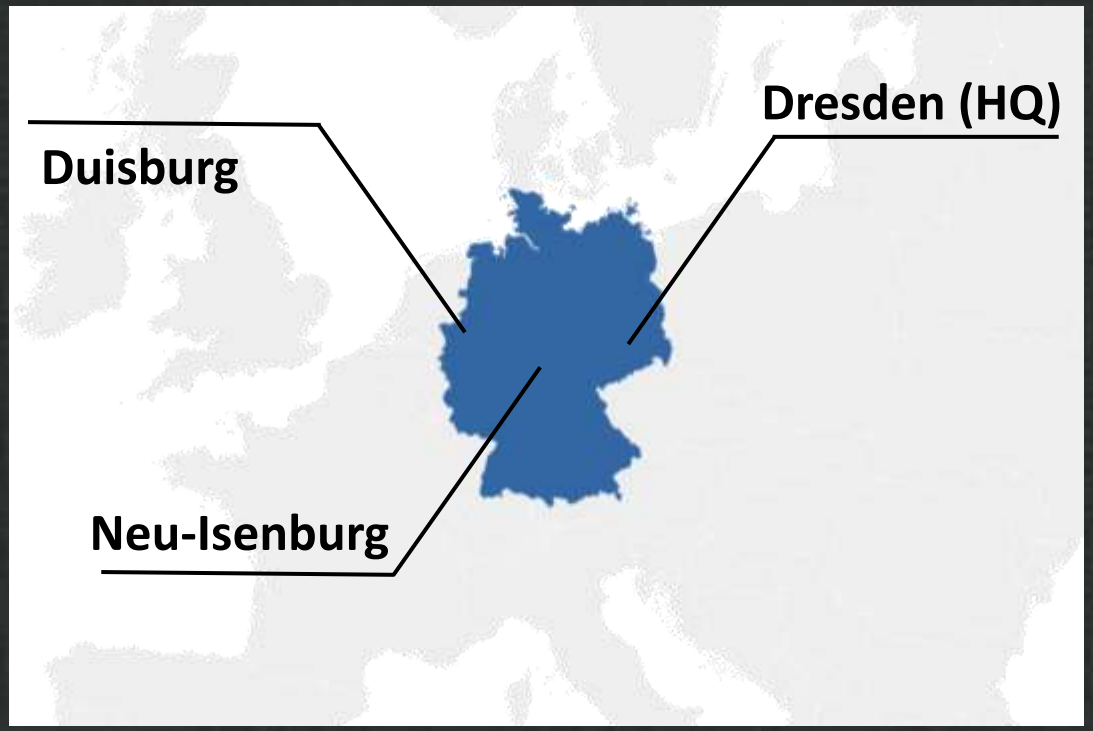
Fast Growing with +100 Employees

>100 successful Tapeouts since 2009

Office Locations in Dresden, Duisburg and Frankfurt/Main

www.racyics.com/jobs

LEADING EUROPEAN ADVANCED NODE SOC/ASIC DESIGN CENTER



www.racyics.com/jobs

The end…

Time for your questions..