Digital Design and Test Automation Flow Lab

Introduction and Overview

Mehdi Tahoori

INSTITUTE OF COMPUTER ENGINEERING (ITEC) – CHAIR FOR DEPENDABLE NANO COMPUTING (CDNC)

KIT – University of the State of Baden-Wuerttemberg and
National Research Center of the Helmholtz Association

www.kit.edu

Objective

- Electronic Design Automation (EDA)
  - Behind all novel electronic systems that we use in our daily lives
    - Such as iPod, smartphones, laptops, TVs, digital cameras, etc.
- The objective of this lab
  - To have a hands-on practice on major steps in digital design and test automation flow
    - From system-level specification to physical design and verification
      - Using industrial EDA toolsets
  - You will work on sample designs and go through all major design and test steps
    - Become familiar with the steps and tool chain in the digital design and test automation flow

Topics

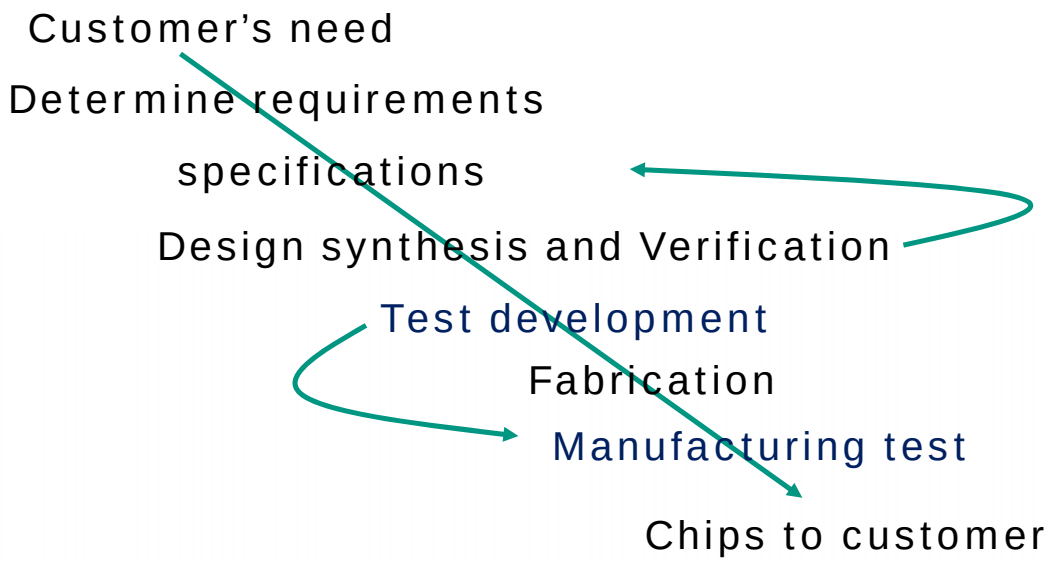
- System-level specification and simulation
- High-level synthesis
- Logic-level synthesis and simulation
- Design for testability
- Test pattern generation and fault simulation
- Circuit design, simulation and verification
- Timing analysis and closure
- Area, delay, and power estimation and analysis

With a flavor of dependable computing

Structure

- Block 1: System level design
- Block 2: RTL design and test
- Block 3: Circuit level design

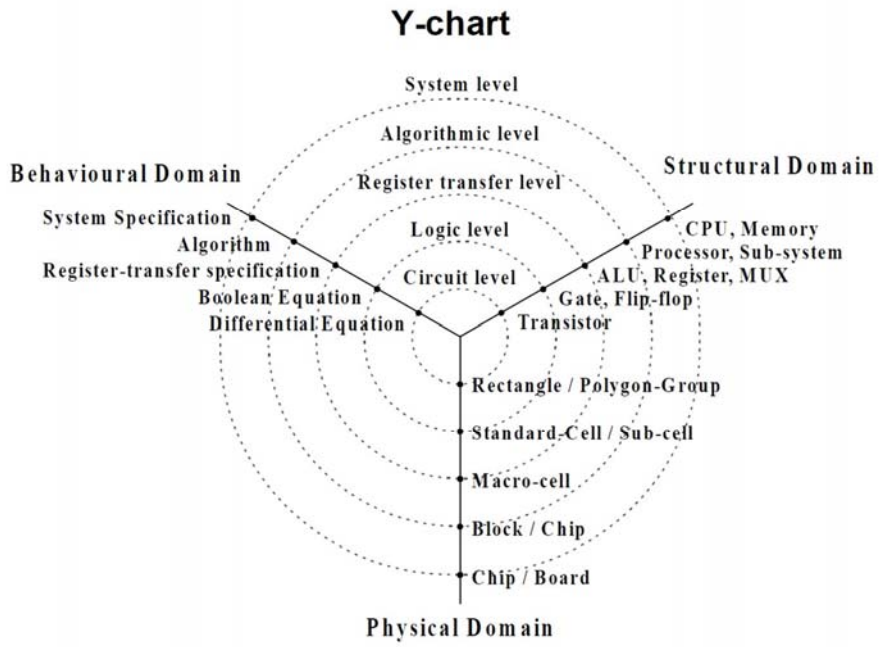
VLSI Realization Process



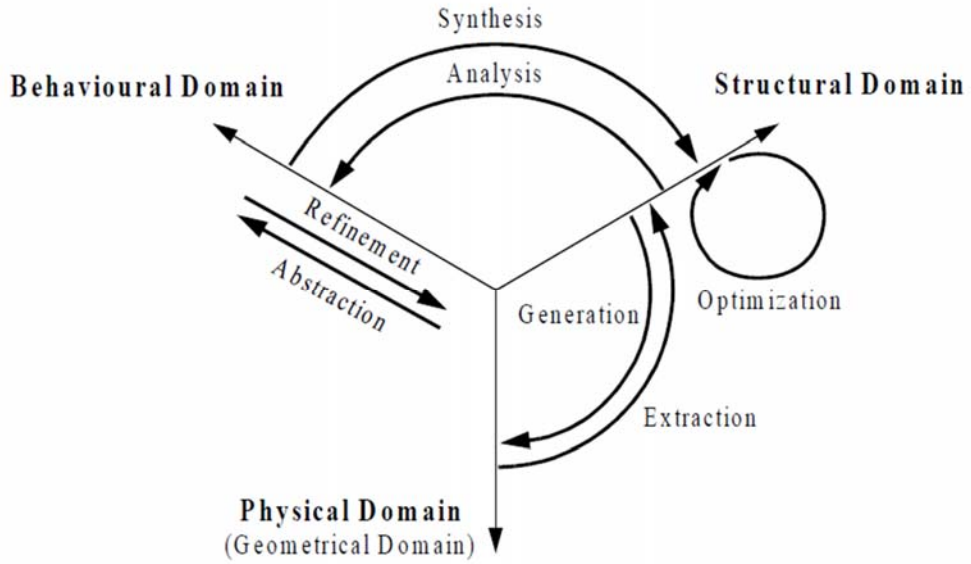
Definitions

- Design synthesis:
  - Given an Input-Output function, develop a procedure to manufacture a device using known materials and processes
- Verification:
  - Predictive analysis to ensure that the synthesized design, when manufactured, will perform the given Input-Output function

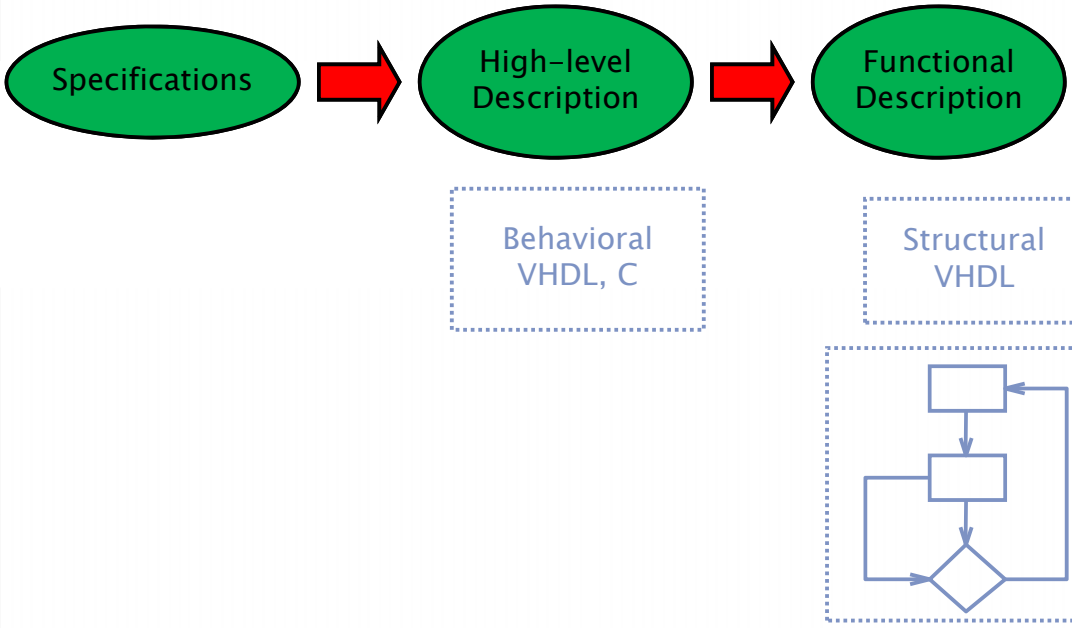
Y Chart



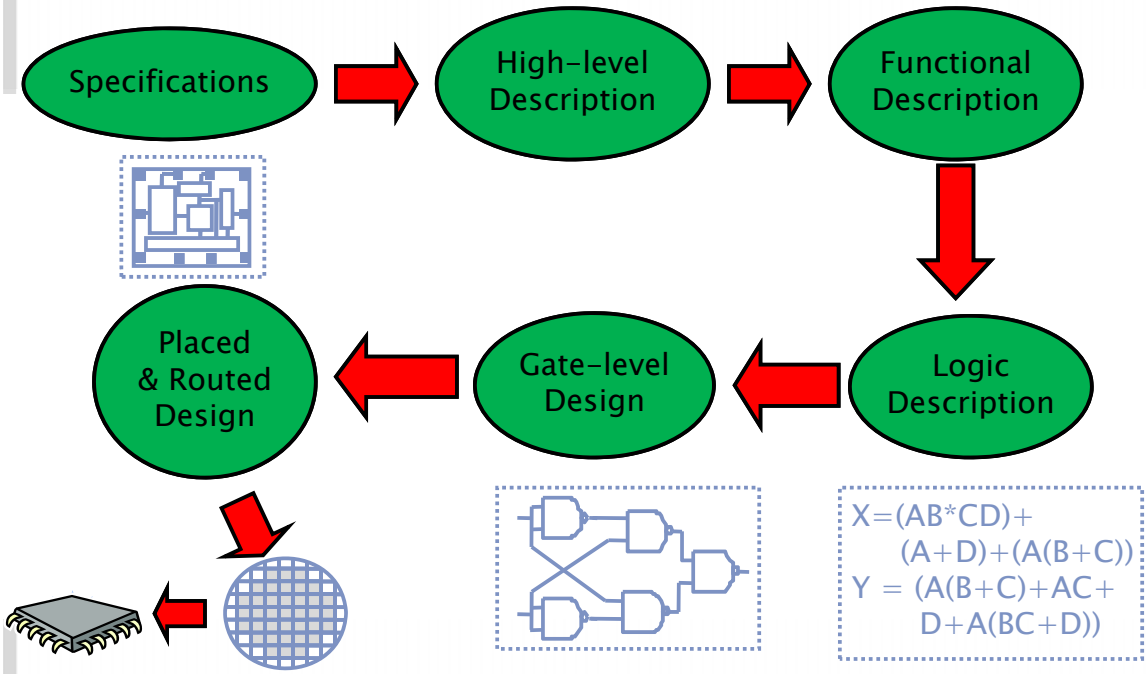
Y transformations



Design Steps



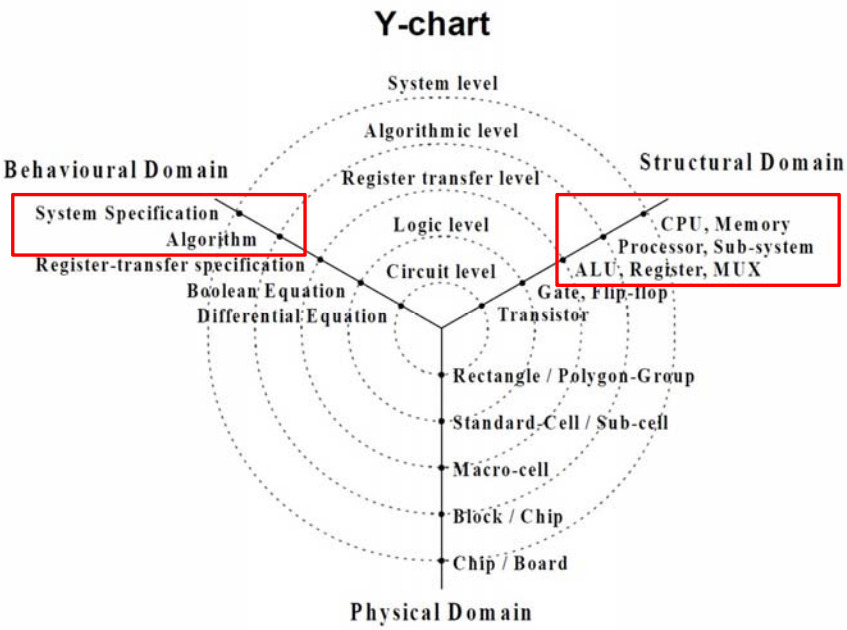
Design Steps (cont)



Block1 : System level design

- We will focus in this block on high level abstraction
- C++ for high level design of a microprocessor.
- Benefits:
  - Faster implementation of the specifications.
    - The behavior of the system can be practiced in early phase.
  - Early discovery of errors in the specifications.
    - Before the manufacturing begins
  - Early performance estimations are possible
    - Find possible bottlenecks
    - Easy design space exploration

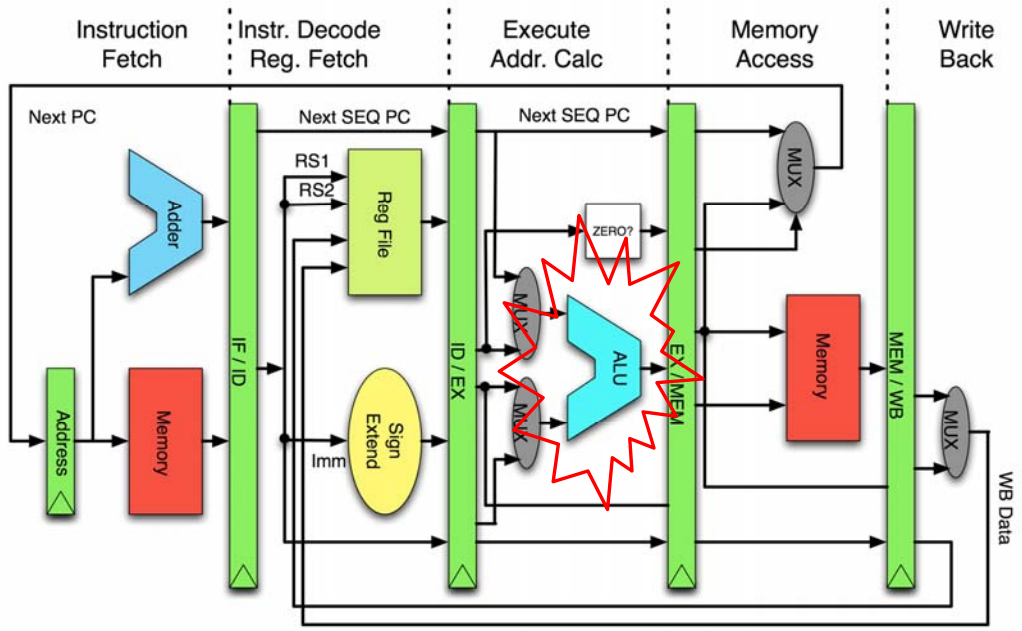
Behavioral Domain



Example: MIPS processor

- MIPS comes from **M**icroprocessor without **I**nterlocked **P**ipeline **S**tages
- Reduced Instruction Set Computer (RISC) design
- 32-bit and 64-bit versions
- Used mainly in embedded systems
  - Windows CE devices
  - Routers
  - Video game consoles
    - Sony PlayStation 2 and PlayStation Portable
- Many manufacturers are using the MIPS license:
  - ATI & AMD
  - Infineon
  - Toshiba

MIPS pipeline structure

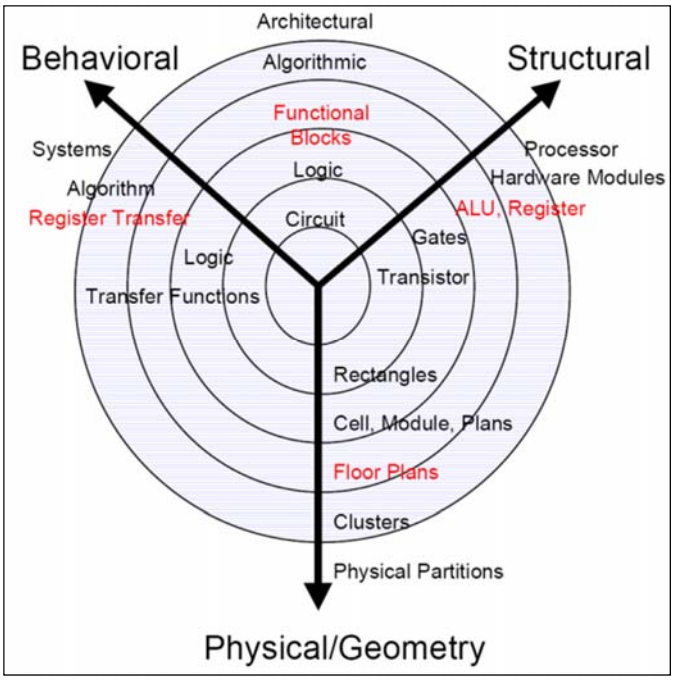


Main tasks in Block1

- Re-implementation of the MIPS ALU using bit level
- Adding fault detection and recovery strategies to the ALU.
- Implementation of a fault injection mechanism
  - To test the efficiency of the added techniques

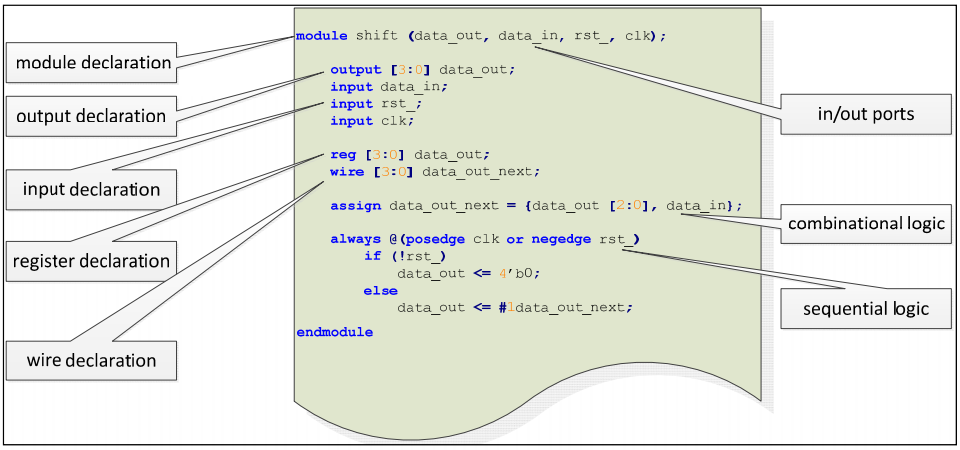
- Main tools:
  - Qt Creator development environment
    - Cross-platform
    - Qt GUI
    - C++
    - Free version available
  - SPIM Simulator
    - SPIM simulator
    - Implemented in Qt Creator
    - Open source

Block2: RTL Design and Test



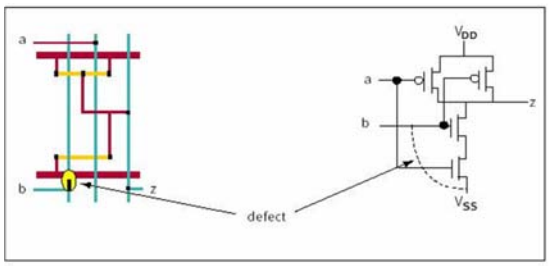
Main content of lectures (1)

- RTL design and verification
  - Design flow, design methodology
  - Verilog basics, synthesizable coding
  - Modelsim demonstration

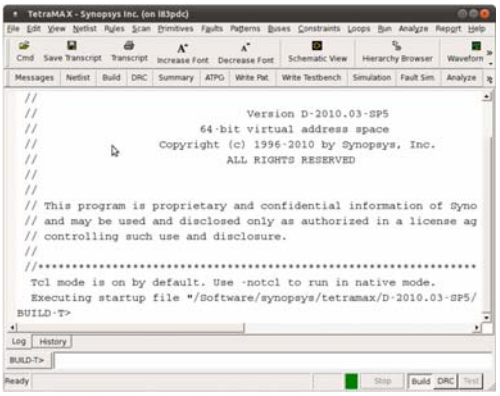


Main content of lectures (2)

- Test fundamental
  - Digital circuit test basics
  - TetraMax tutorial
  - Demonstration

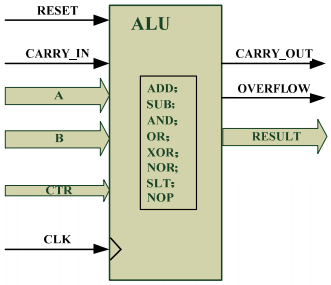


http://web.iiit.ac.in/~mohanbvm/VlsiTestingTool.htm



Lab tasks and arrangements

- Session 1 ~ 2
  - Verilog design of a simple ALU

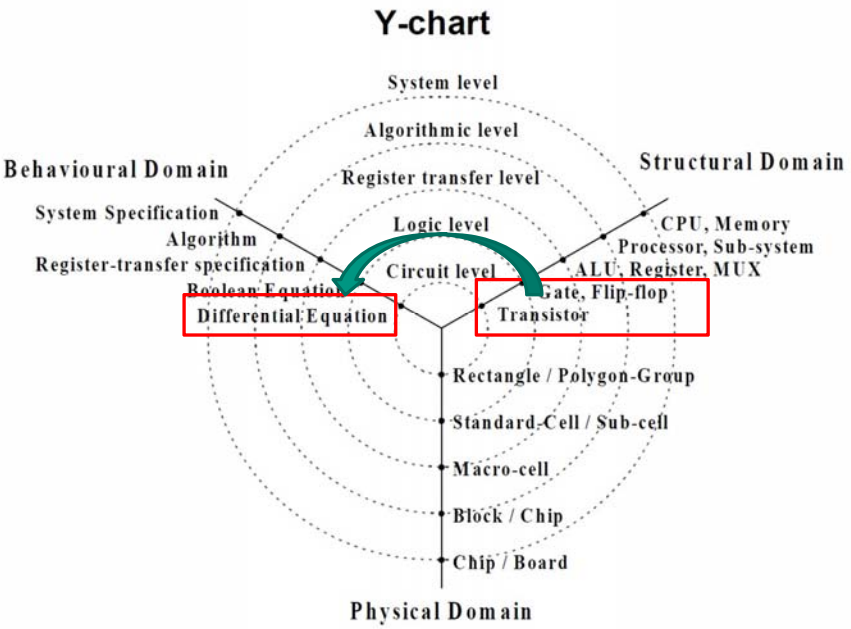


- Session 3
  - Logic synthesis of designed ALU
- Session 4
  - ATPG and fault simulation about the synthesized ALU

Block3 : Circuit level design

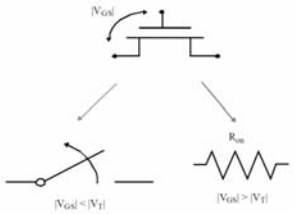
- We will focus on transistor level design
- HSPICE simulations of CMOS gate
- Motivation:
  - Scaling the technology → everything becomes analogue
  - More accurate simulations
    - Power measurement of basic cells
    - Delay measurement of basic cells
  - Optimize basic cells for meet in the middle strategy
    - For area, delay and power
  - Physics based analysis is possible
    - Process Variation analysis
    - Reliability analysis

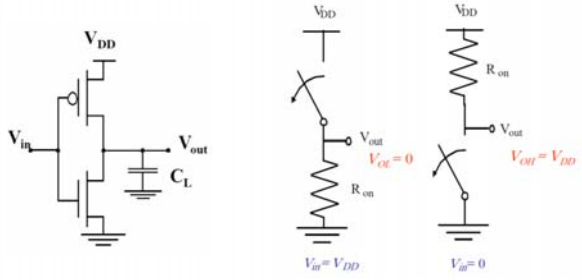
Behavioral Domain



Introduction of CMOS technology

- **MOSFET**
  - MOS : Metal Oxide Semiconductor
  - FET: Field Effect transistor
  - CMOS: Complementary MOS:
    - which means using both NMOS and PMOS transistors to form a circuit.
- Transistors are not ideal switches.
  - NMOS cannot transfer V_{DD} ideally
  - And PMOS cannot transfer 0 ideally.
- CMOS inverter:





Main tasks in Block3

- Introducing the structure of basic CMOS cells
  - What are the challenges
  - How to overcome
- Use Hspice as analogue simulator for basic cells:
  - Design
  - Verify the functionality
    - DC analysis
    - Transient analysis
  - Delay measurement
  - Power measurement
- Basic cells are
  - Simple structures: NOT, NAND, NOR
  - More complex structures: Full Adder and Ripple Carry Adder
- Main tools:
  - Hspice

Dates ???

- As a block Lab (recommended)
  - At the end of the semester
  - 5 full days from 16-20 July

- Wednesdays 14:00 – 18:00 or
- Thursdays 14:00 – 18:00