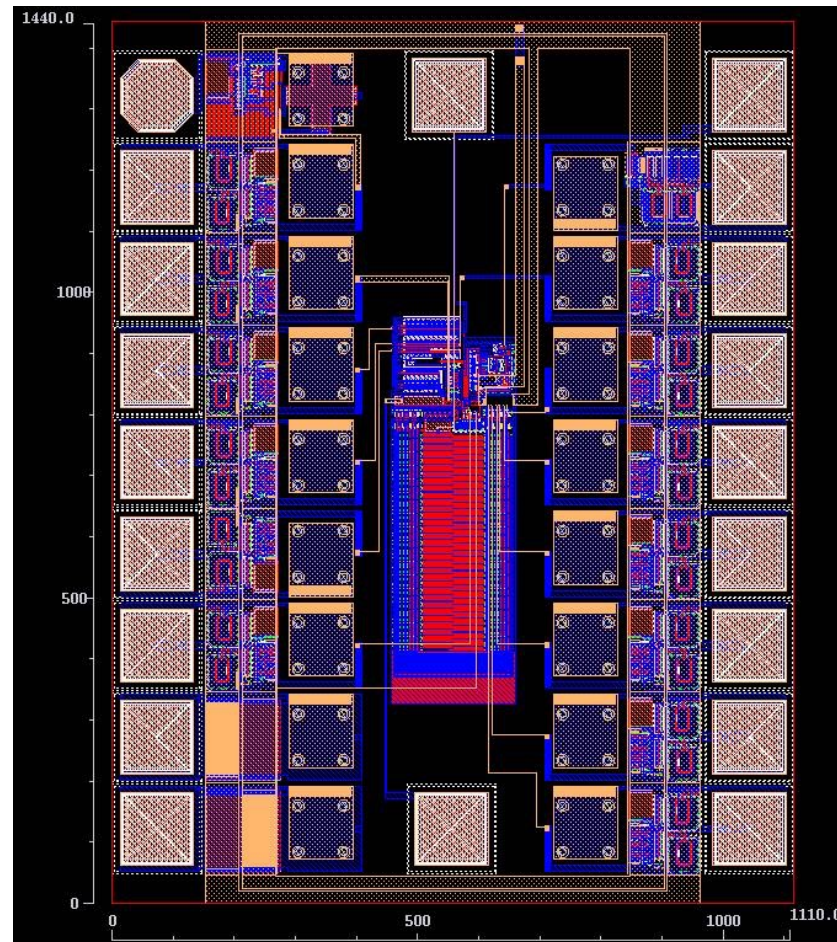


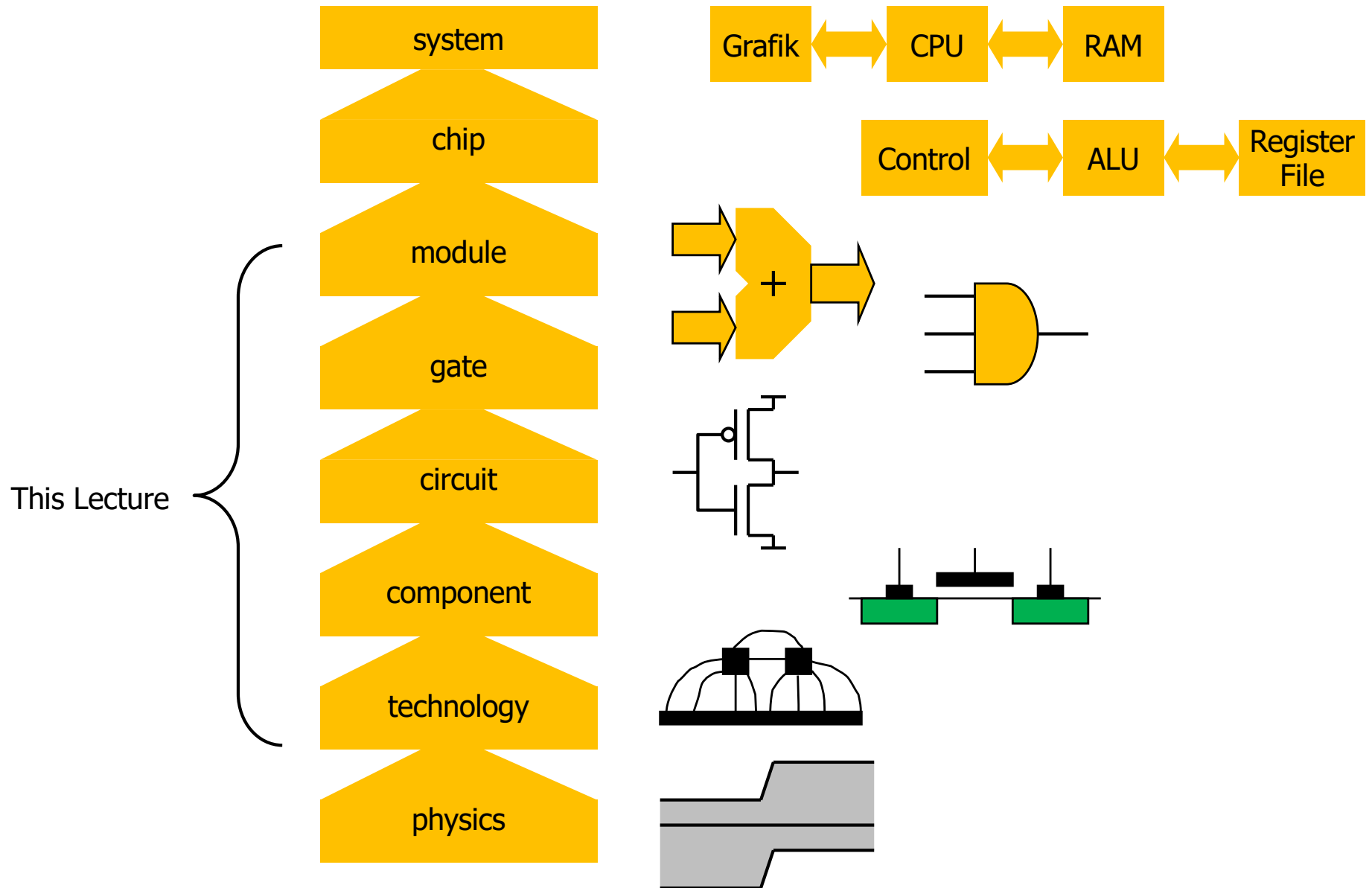


# Design Flow





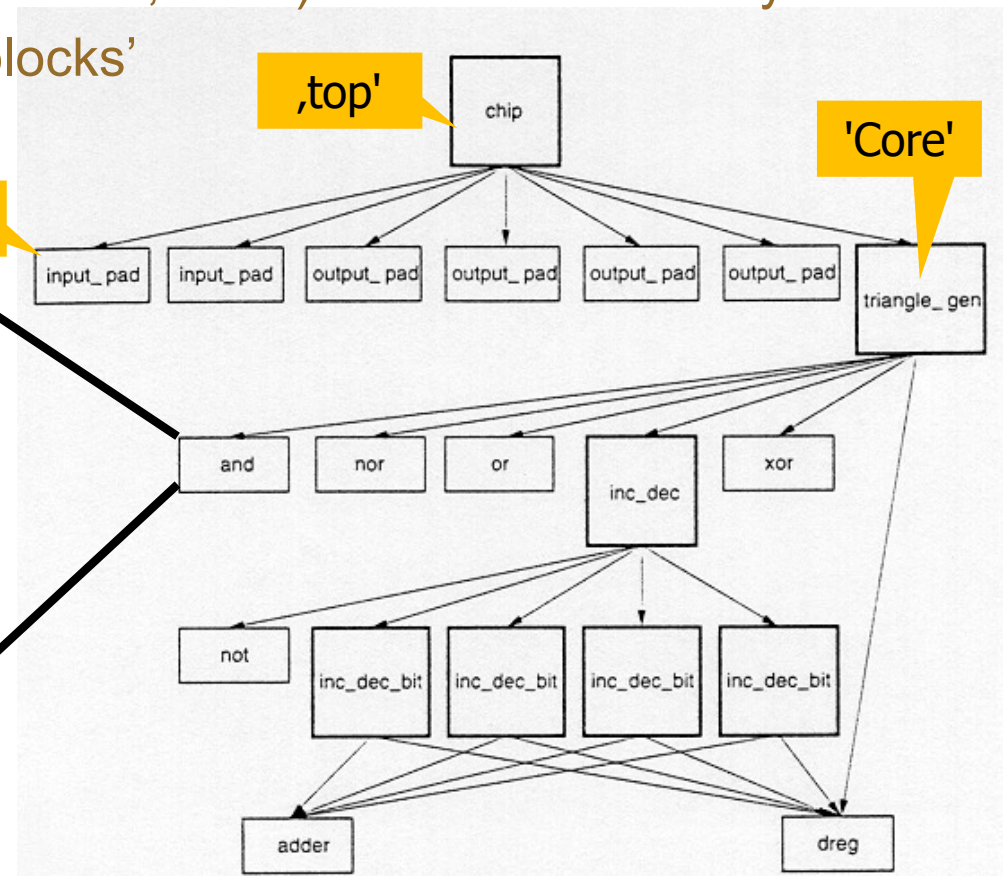
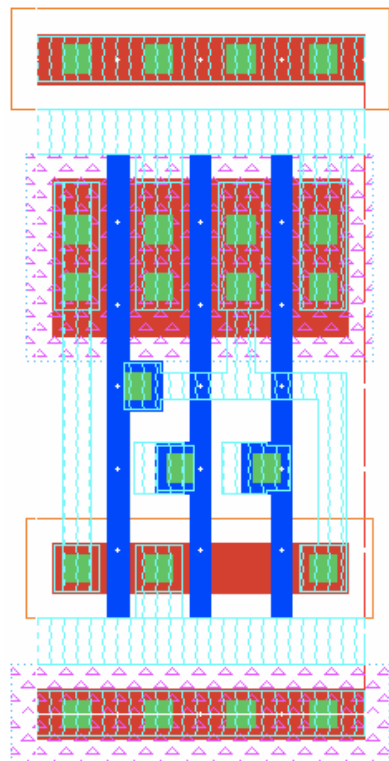
# Levels of Abstraction





# Design Hierarchy

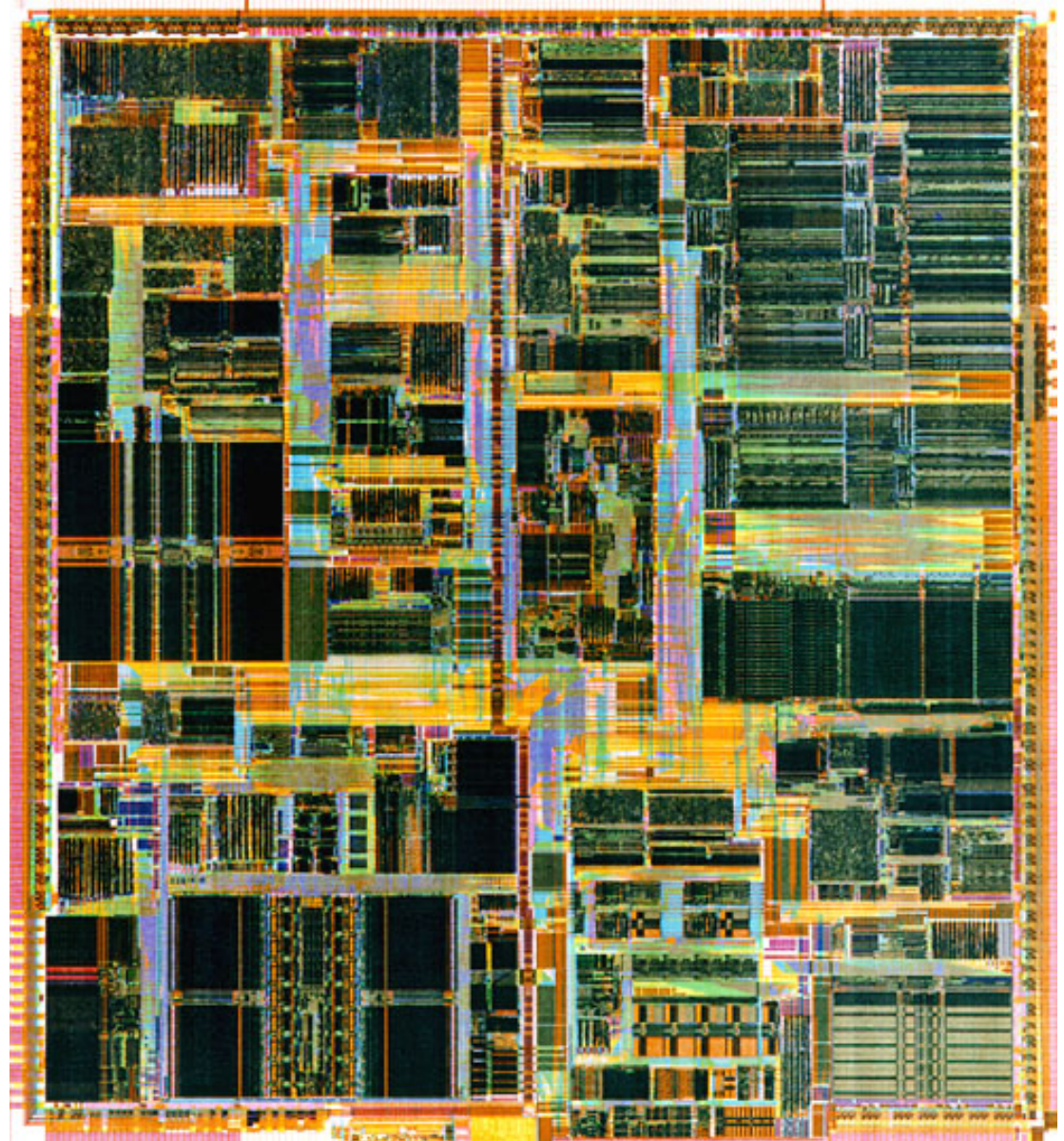
- A chip is subdivided in smaller blocks
  - Described by schematics or hardware description language (HDL)
- At lowest Level:
  - simple logic functions (NAND, MUX) + 'standard cell' layouts
  - Analogue 'full custom blocks'





# A real(y deep) Design Hierarchy

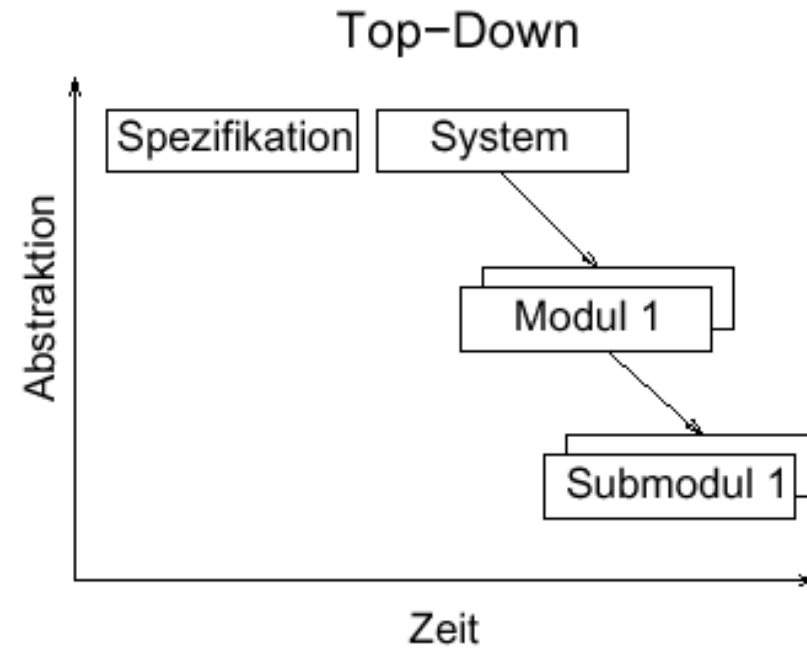
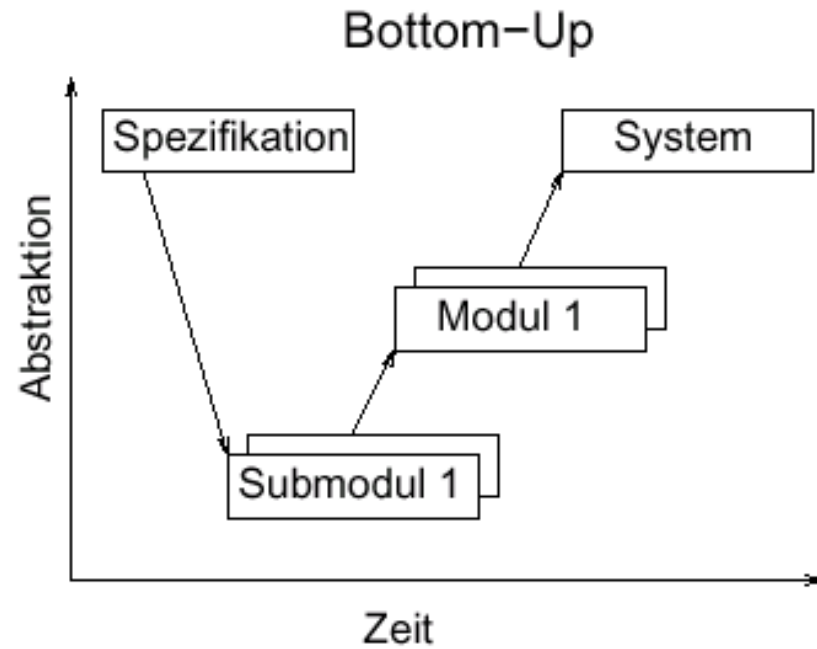
- The hierarchy has MANY levels
- In our chips: max. 10-15 levels



Pentium P6



# Top-Down, Bottom-Up

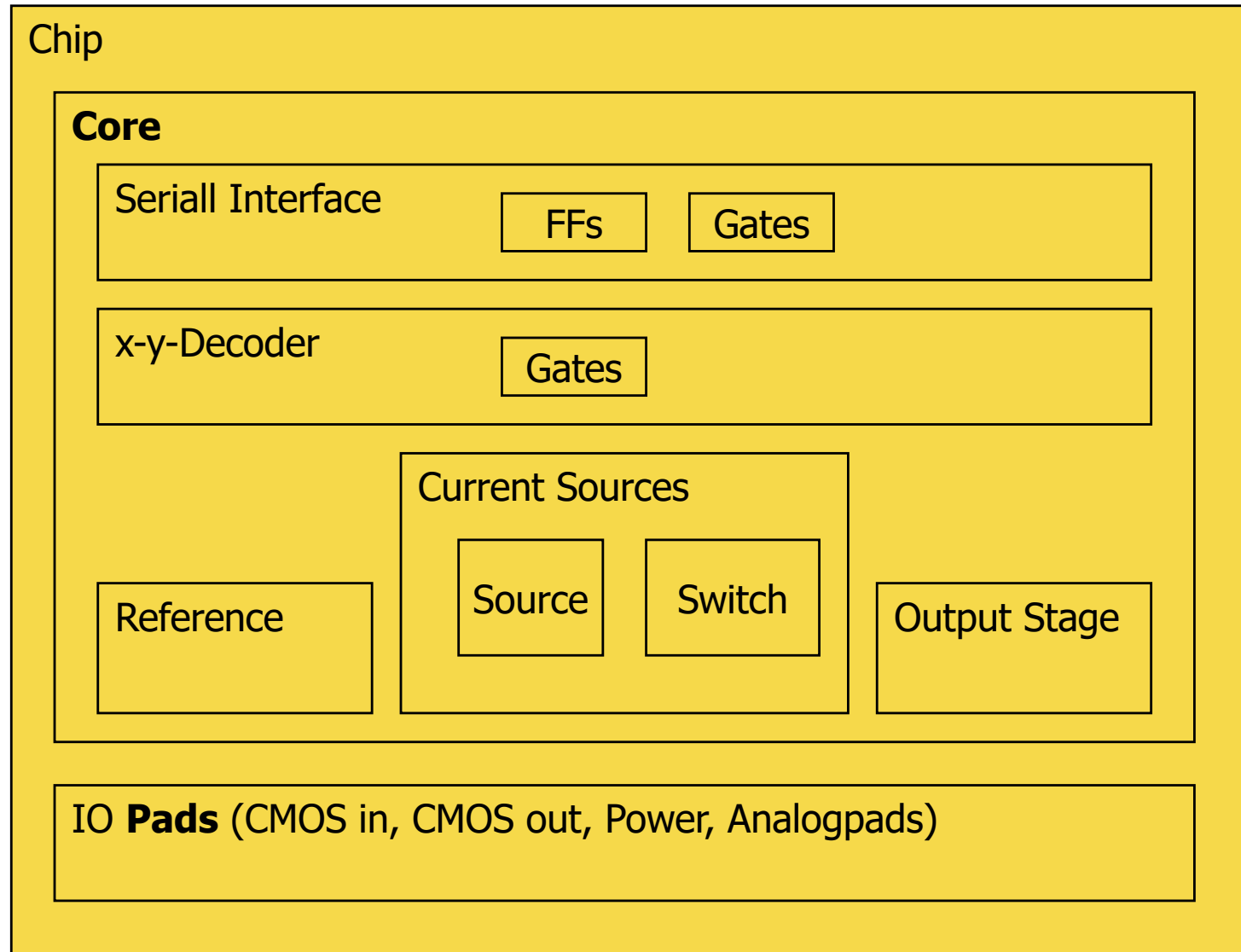


- I believe: both methods have drawbacks
- Try to come from both sides!



# Example: DAC

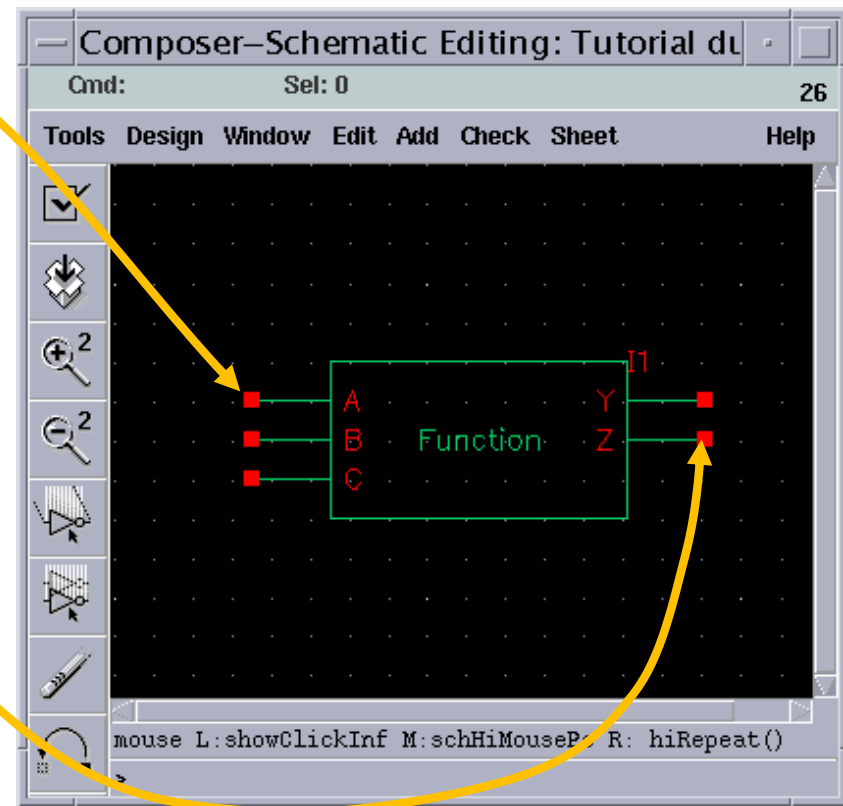
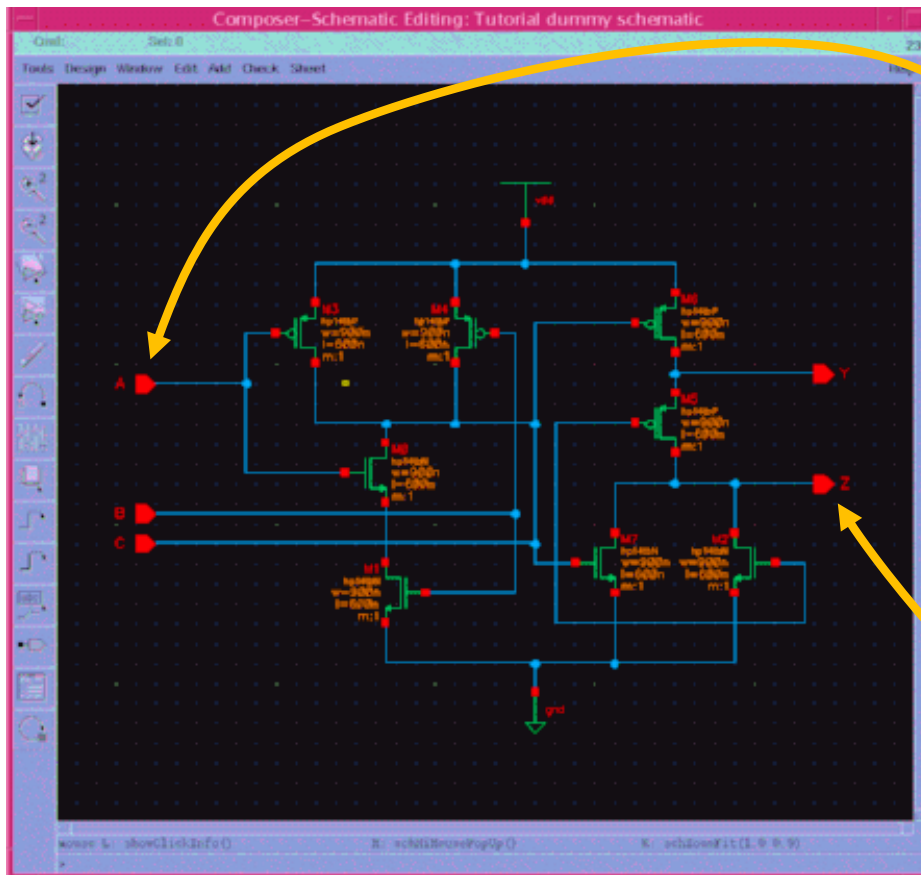
- Digital Analogue Converter: Binary Code  $\rightarrow$  Current





# 1. Schematic and Symbol

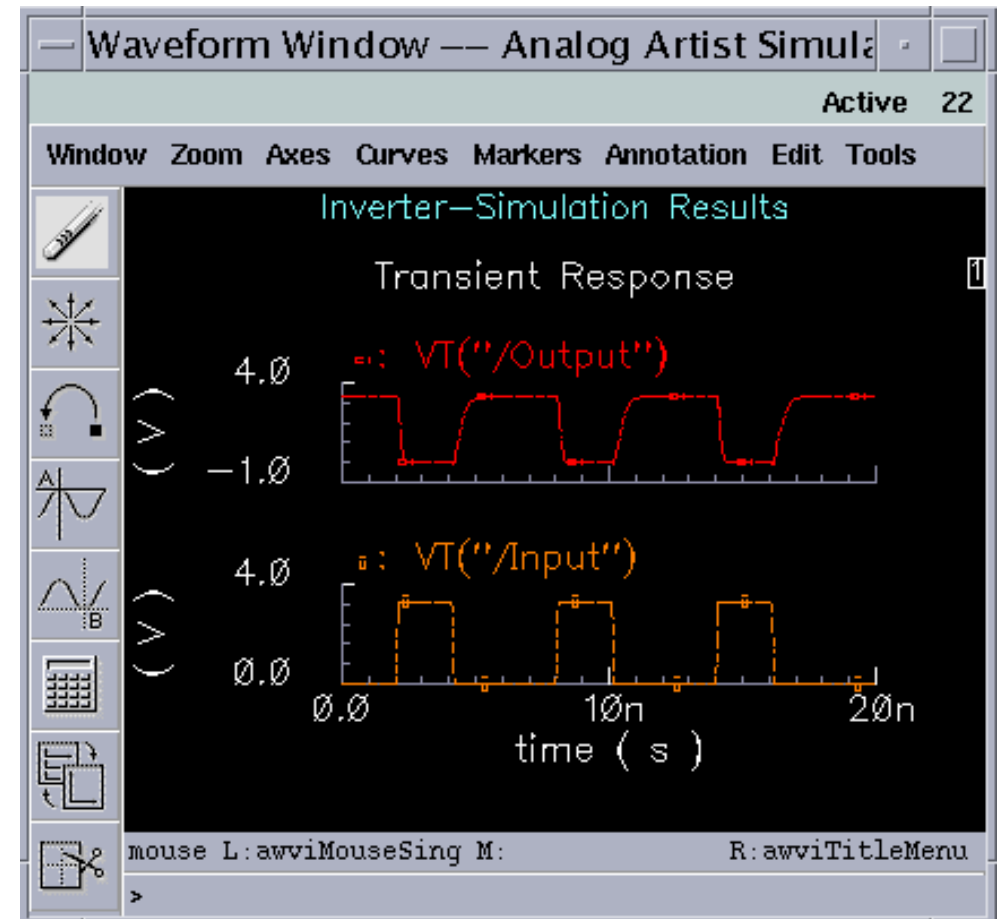
- Schematic describes circuit with other cells & transistors
- A ‚symbol‘ of the cell contains all pins
  - It is later used to represent the schematic in the hierarchy





## 2. Simulation

- Mostly analogue (for us)
- Models are different for each vendor! Are provided by vendor. Should be verified..
- Optimize circuits
- Results can be used to parameterize the cell.
- For instance, digital cells can be modelled by delay (depending on  $C_{load}$ , VDD, Temp,..)
- $\Rightarrow$  digital (VERILOG) Model

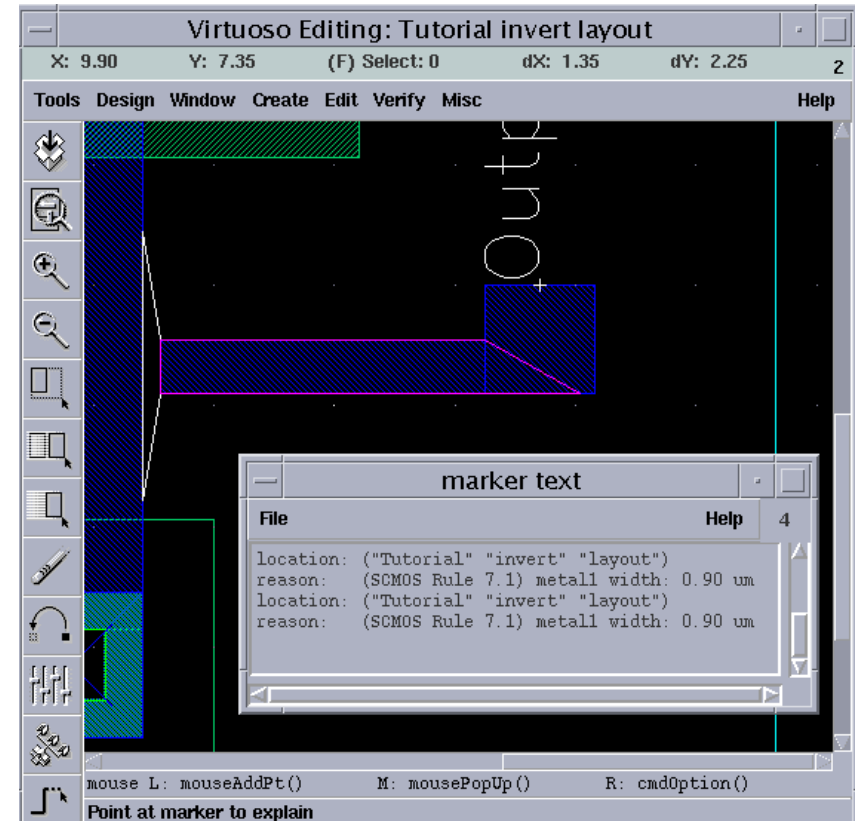
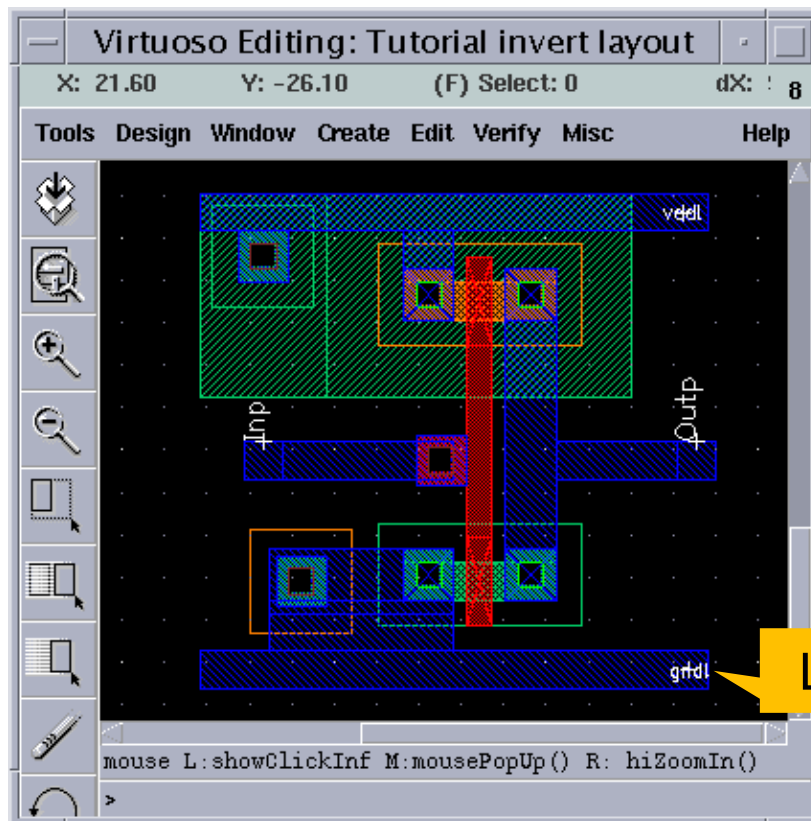






# Layout, Design Rule Check

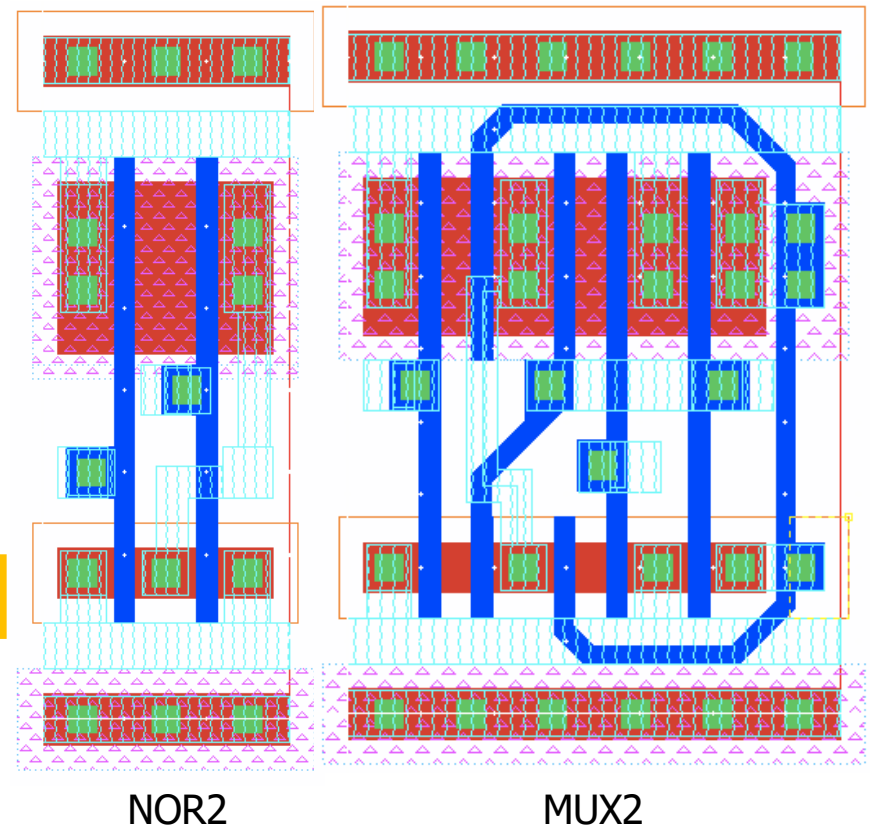
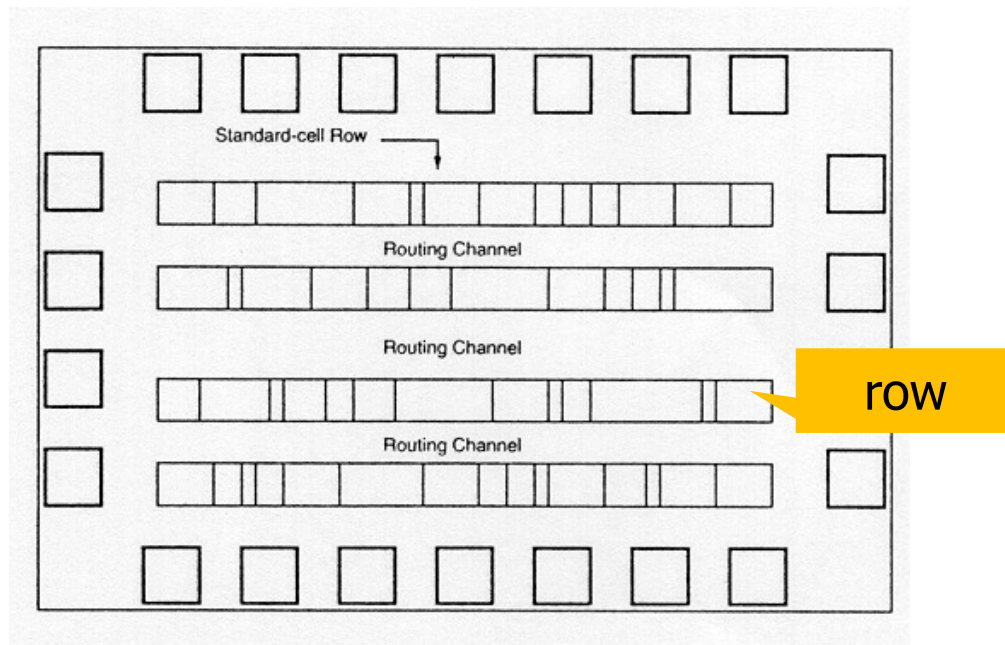
- Layout (mostly by hand) on transistor level
- Check of vendor-given rules with Design Rule Check (DRC) (z.B. conductor widths, spacing,...)
- Design rules are different for each technology!
  - The are collected in a (text) file.





# Standard Cells

- (Digital) standard cells have equal heights and identical (power) connections, so that they can be arranged in rows.
- In technologies with  $>2$  metal layers: routing on top of cells
- 'Optimal' placement of cell is challenging task!
- Wires introduce delays which are not a priori known by simulation!



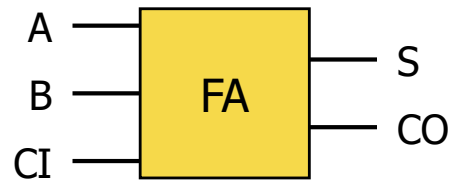


# Special Layout : Bit-Slice

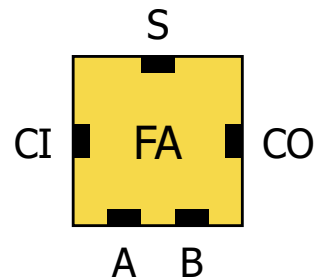
- For repetitive structures: pads are arranged so that touching cells connect correctly
  - compact layout, low delays, best performance

## Full Adder

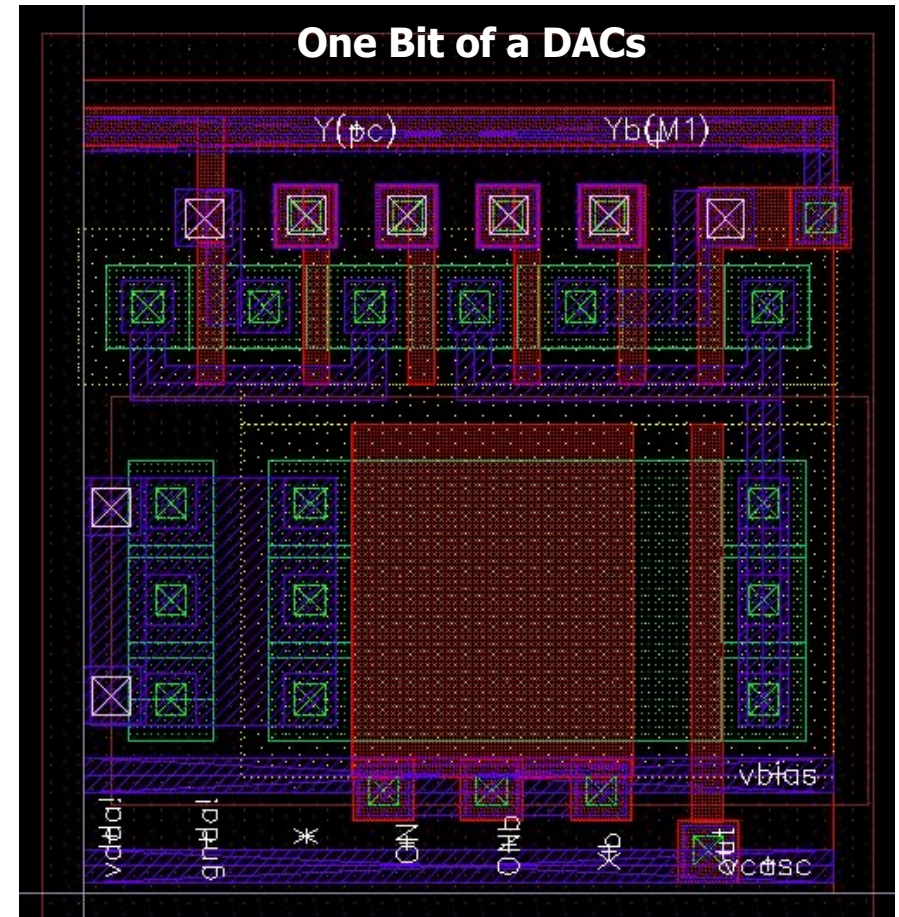
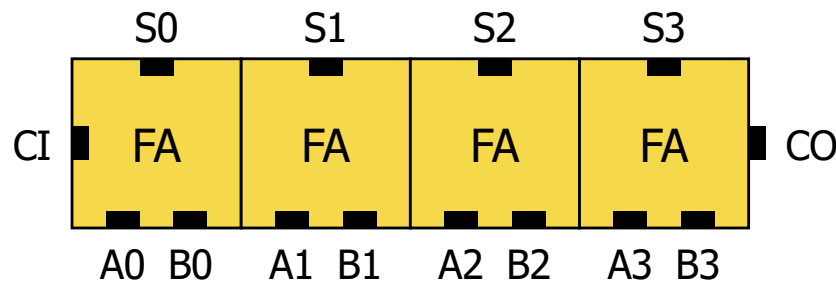
Symbol for 1 bit:



Geometry for 1 Bit:



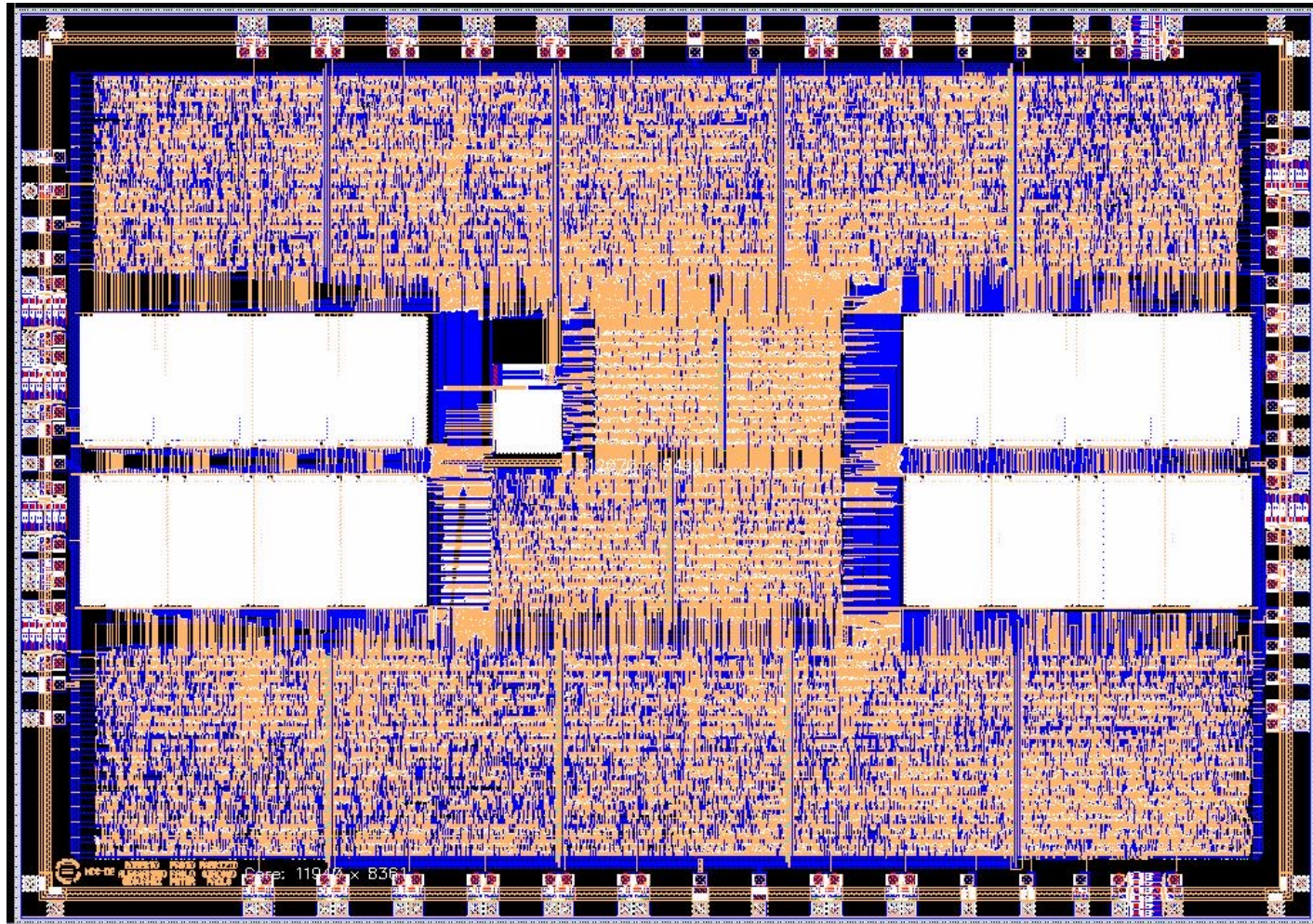
4 Bit Adder:





# Full Chip

- Standard Cells + Full custom cells (here: RAMs, Delays) + Pads ('frame')





# Design Flow

