# Digital ASIC design

EDA322 guest lecture

Lars Svensson, larssv@chalmers.se

# Why?

- ASIC design necessary when FPGAs don’t cut it
  - Performance
  - Power
  - Cost
  - Non-digital features
- Understand when and why to consider

# Overview

- CMOS technology (ASIC + FPGA)
- ASIC vs FPGA:
  - Construction
  - Performance
  - Design flows
- Future

# Electronic systems

- Cabinets, racks, PCBs, chips
- Chips do the actual work!  Examples:
  - General-purpose processor (with software)
  - Special-purpose processor (DSP, GPU, ASSP)
  - Programmable logic (e.g. FPGAs)
  - Application Specific Integrated Circuit  (ASIC)

- All built on CMOS technology

# CMOS technology

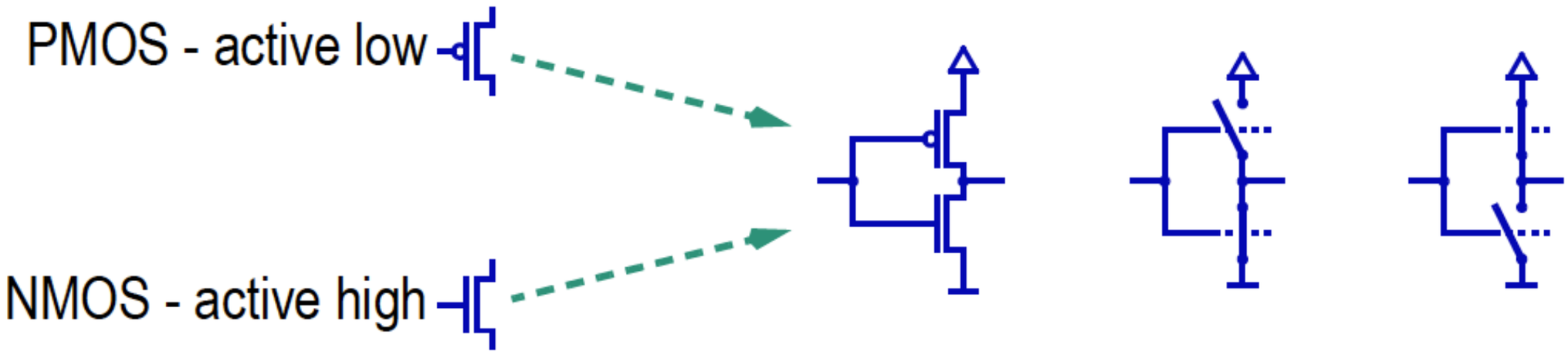
# Technology overview

- "All" digital chips these days built in Complementary MOS, or CMOS, technology
  - Processors, FPGAs, ASICs, …
  - Similar manufacturing (some variations for memories, etc)
  - In programmable logic, mfg is "hidden"
    - Not so in ASIC design!

# Complementary MOS

Metal-Oxide-Semiconductor
Field Effect Transistor
(MOSFET)

Complementary MOS
=
CMOS

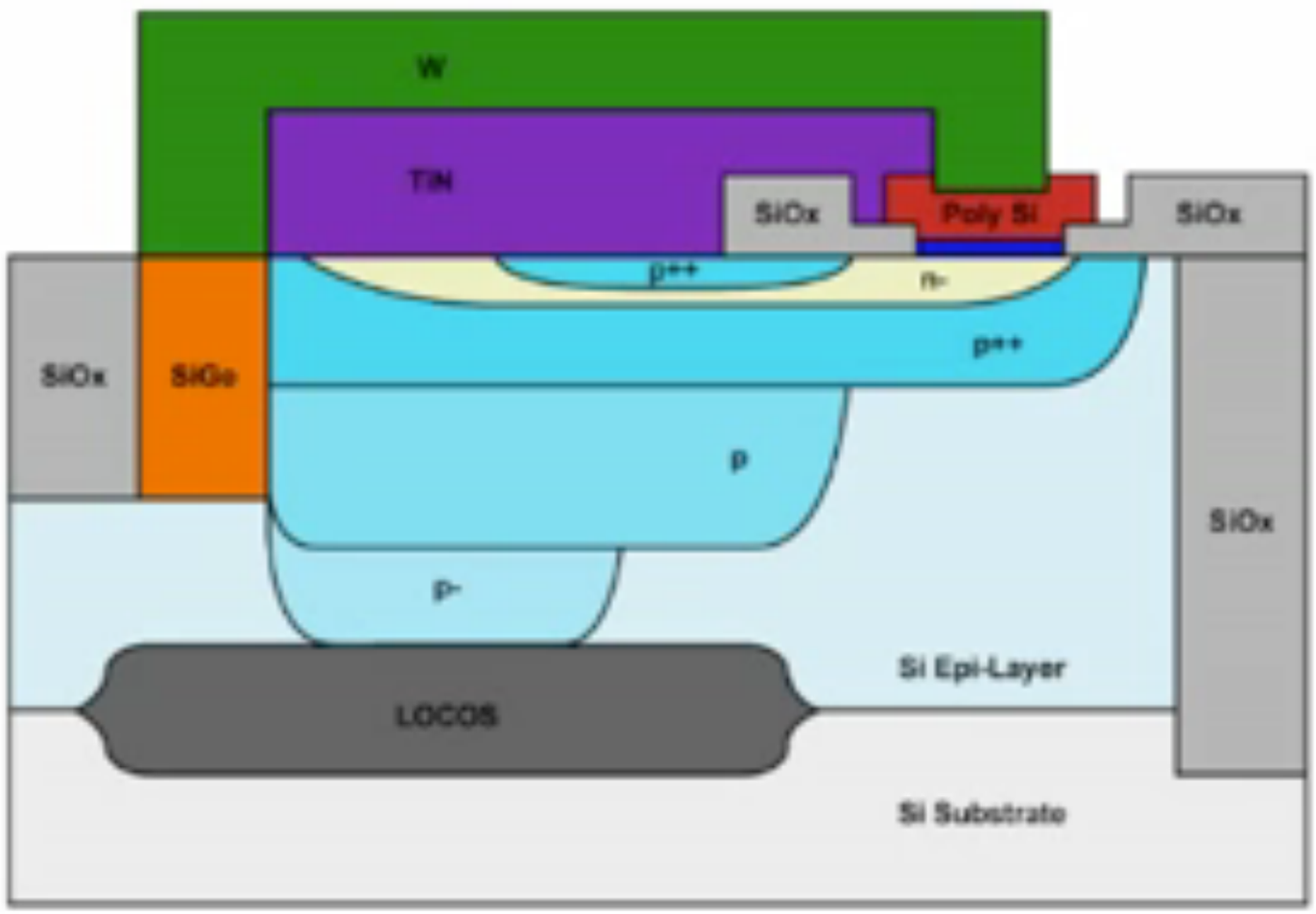



- Here: inverter; high input → low output (etc)
- Transistors (NMOS, PMOS) act as switches
- Connect output to $V_{dd}$ or to GND

# Silicon manufacturing (video)

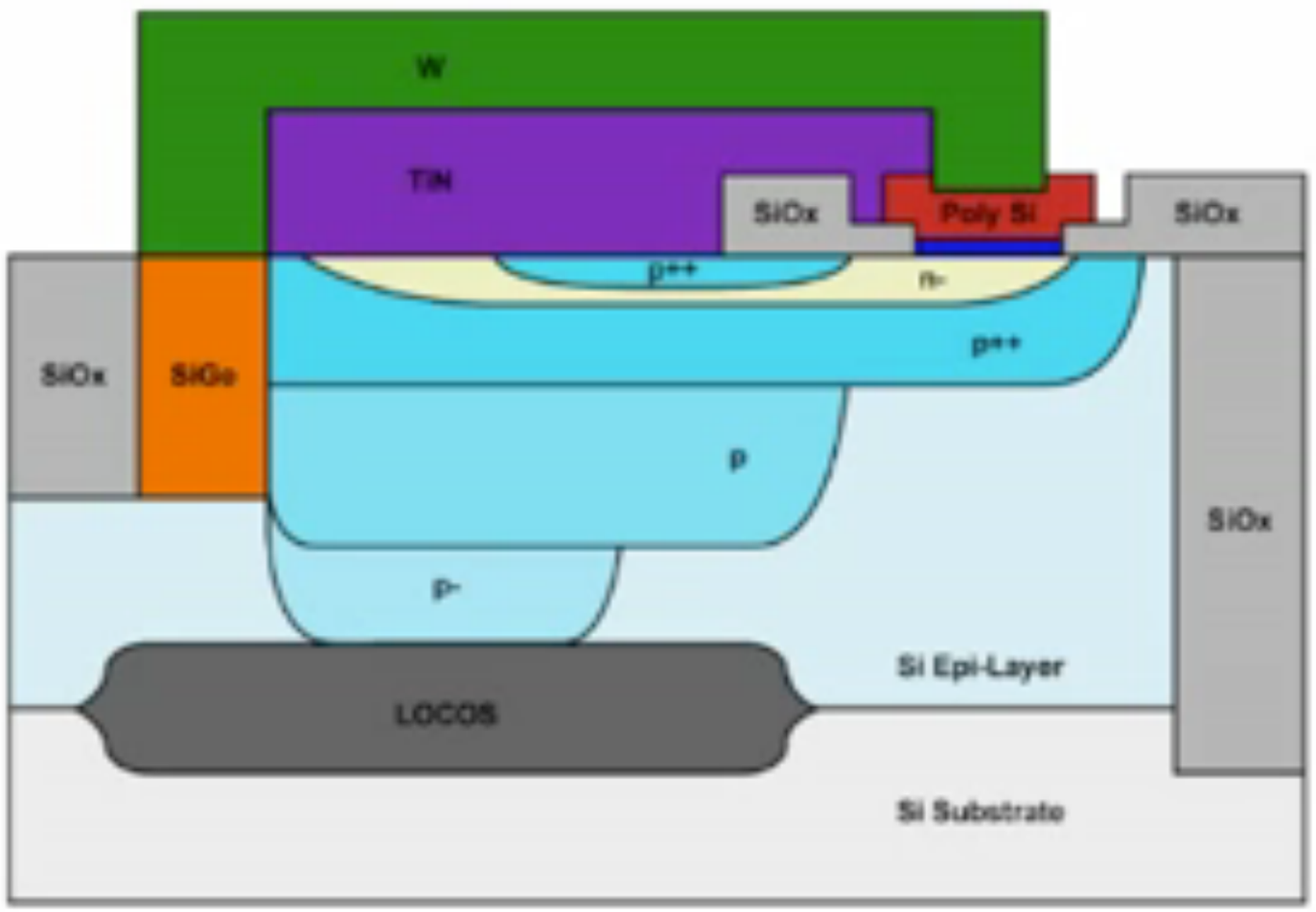
EE 594L
Homework 4
April 17, 2008

Ben McIntosh



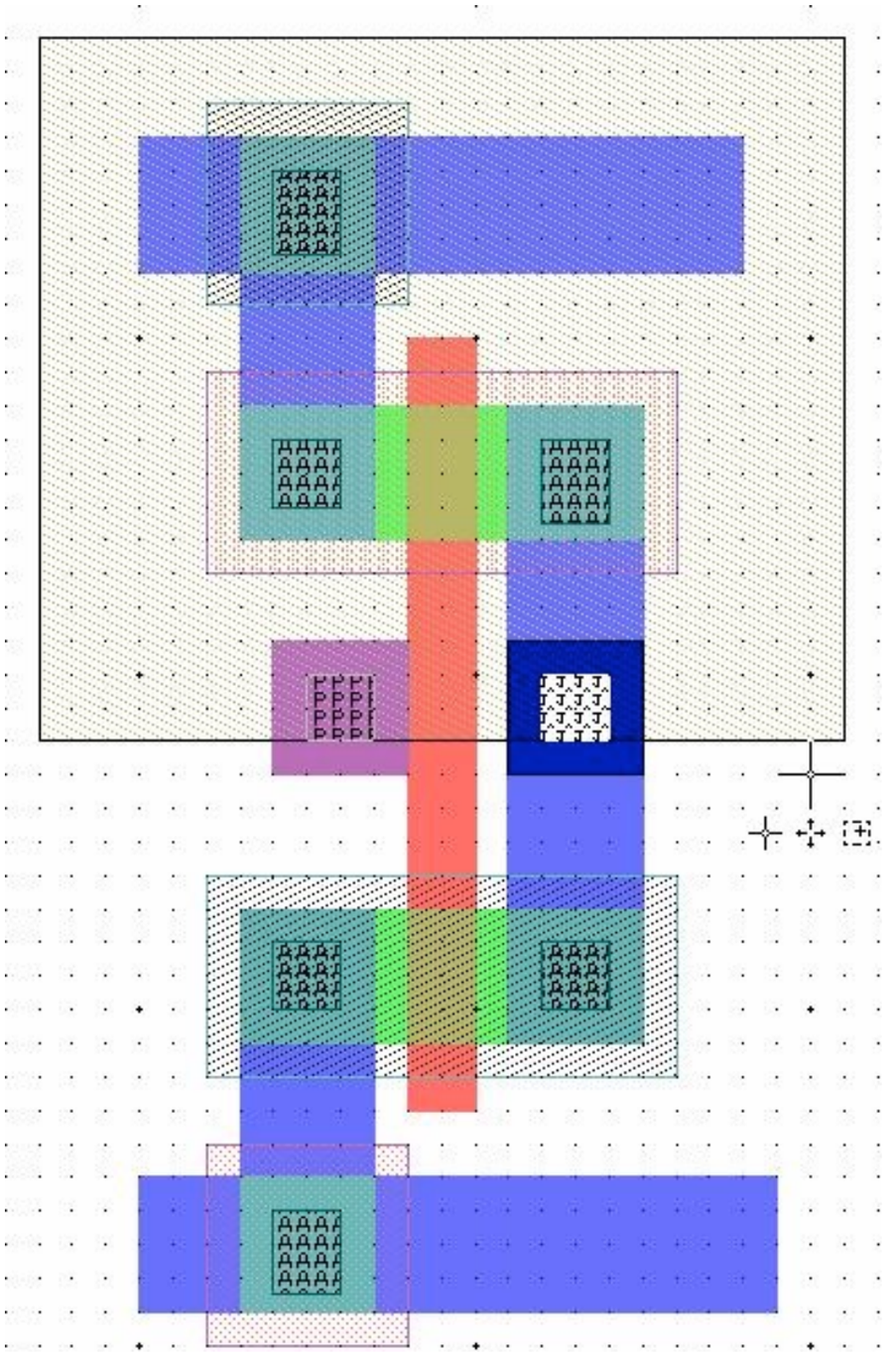
# EE 504L
# Homework 4
# April 17, 2008

## Ben McIntosh

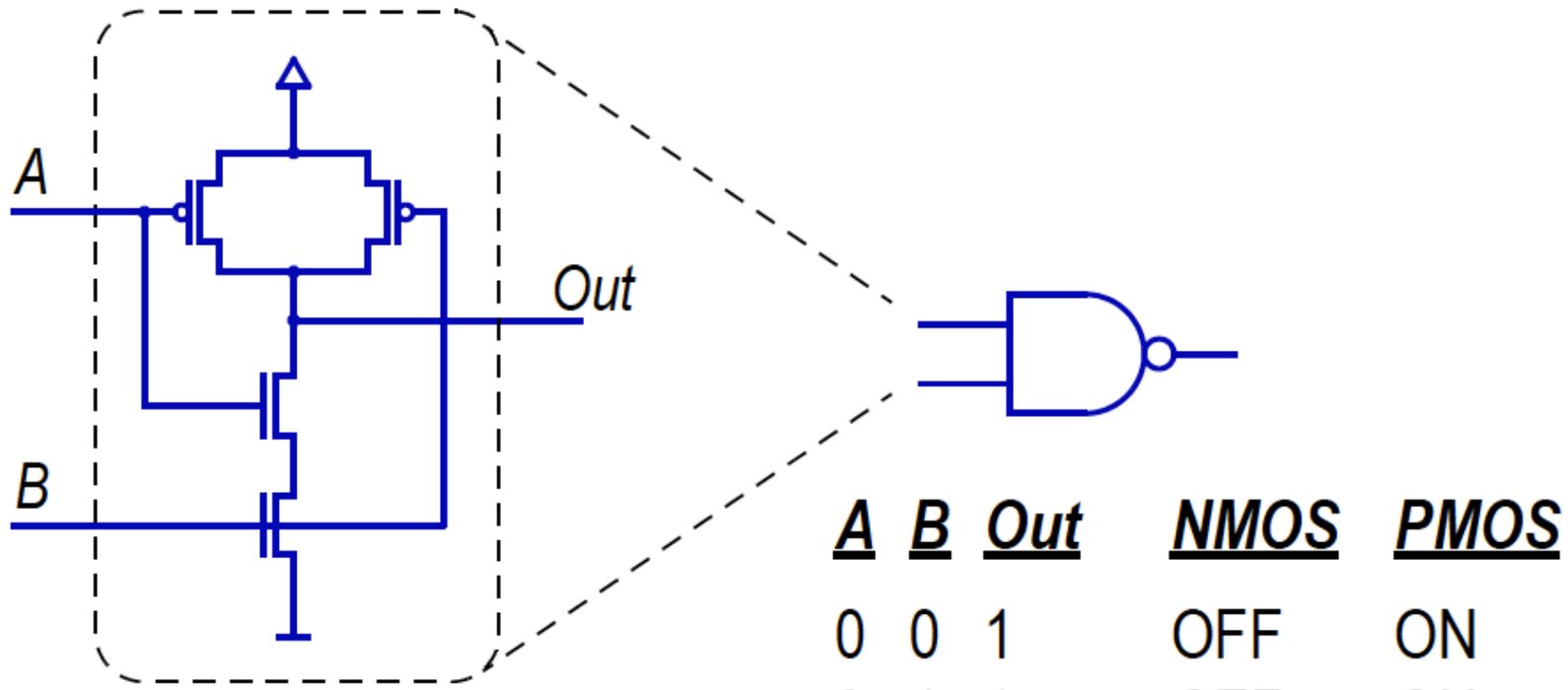


# Inverter cell layout

- Symbolic representation of layers of silicon, metal etc
  - Top (map-like) view
  - Chip layout constructed hierarchically from cells such as this
  - Layout used to create masks for lithography



# Logic gate (here: NAND)




| A | B | Out | NMOS | PMOS |
|---|---|---|---|---|
| 0 | 0 | 1 | OFF | ON |
| 0 | 1 | 1 | OFF | ON |
| 1 | 0 | 1 | OFF | ON |
| 1 | 1 | 0 | ON | OFF |

- Generalized inverter!
- Switches connected in series and parallel
- Function described with Boolean algebra

# Other building blocks

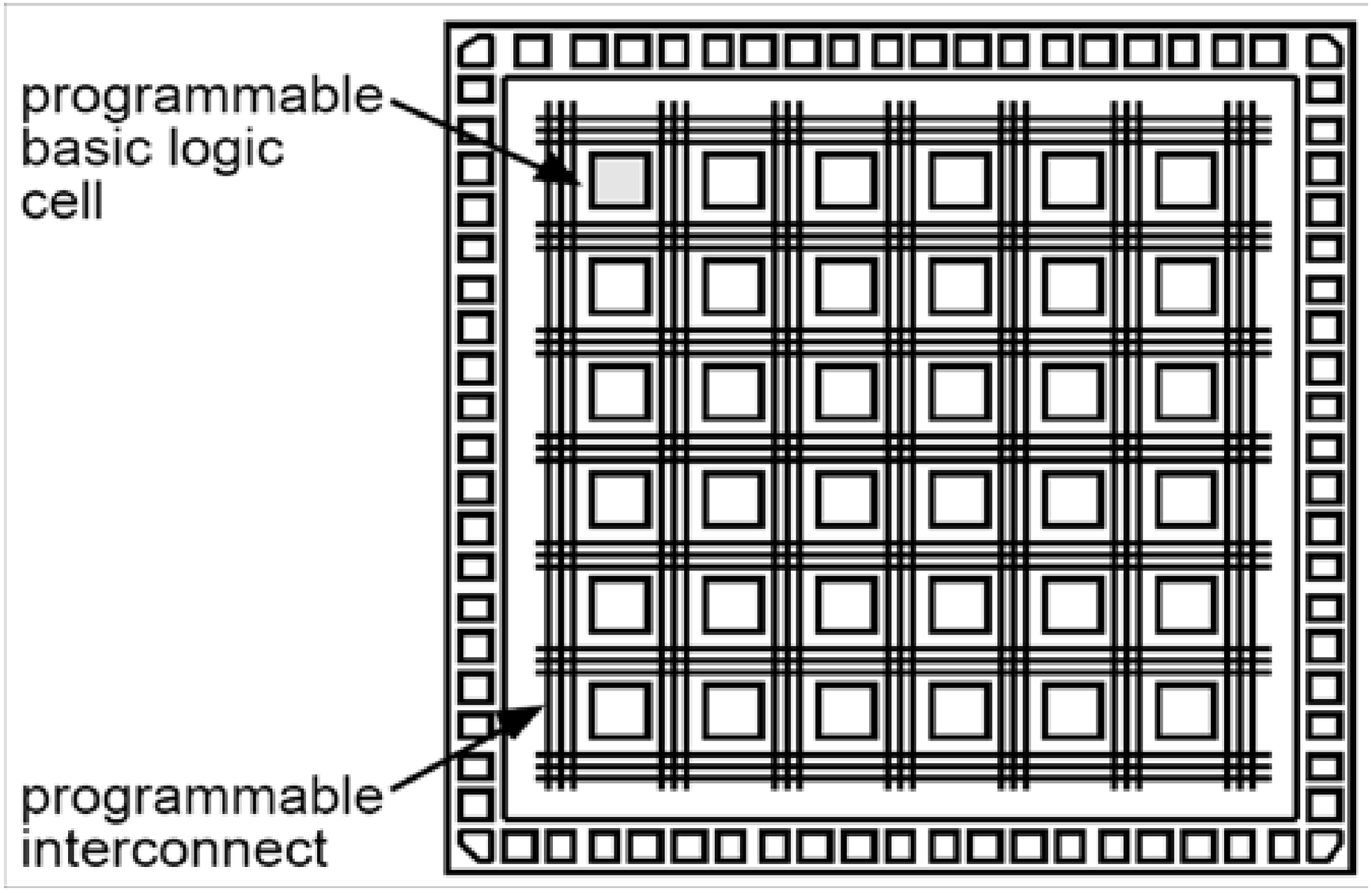
- Built with same set of transistors, etc
  - Logic gates
  - Latches, flip-flops
  - Memories
  - Analog / mixed-signal / special-purpose
- … and then many layers of metal wires

# Design construction

# General approach

- Most often language-based, just as for FPGAs
  - VHDL, Verilog, etc.
- Similar CAD tasks to be carried out …
  - … plus a few more…
  - …motivated by the structural differences in the “substrate”

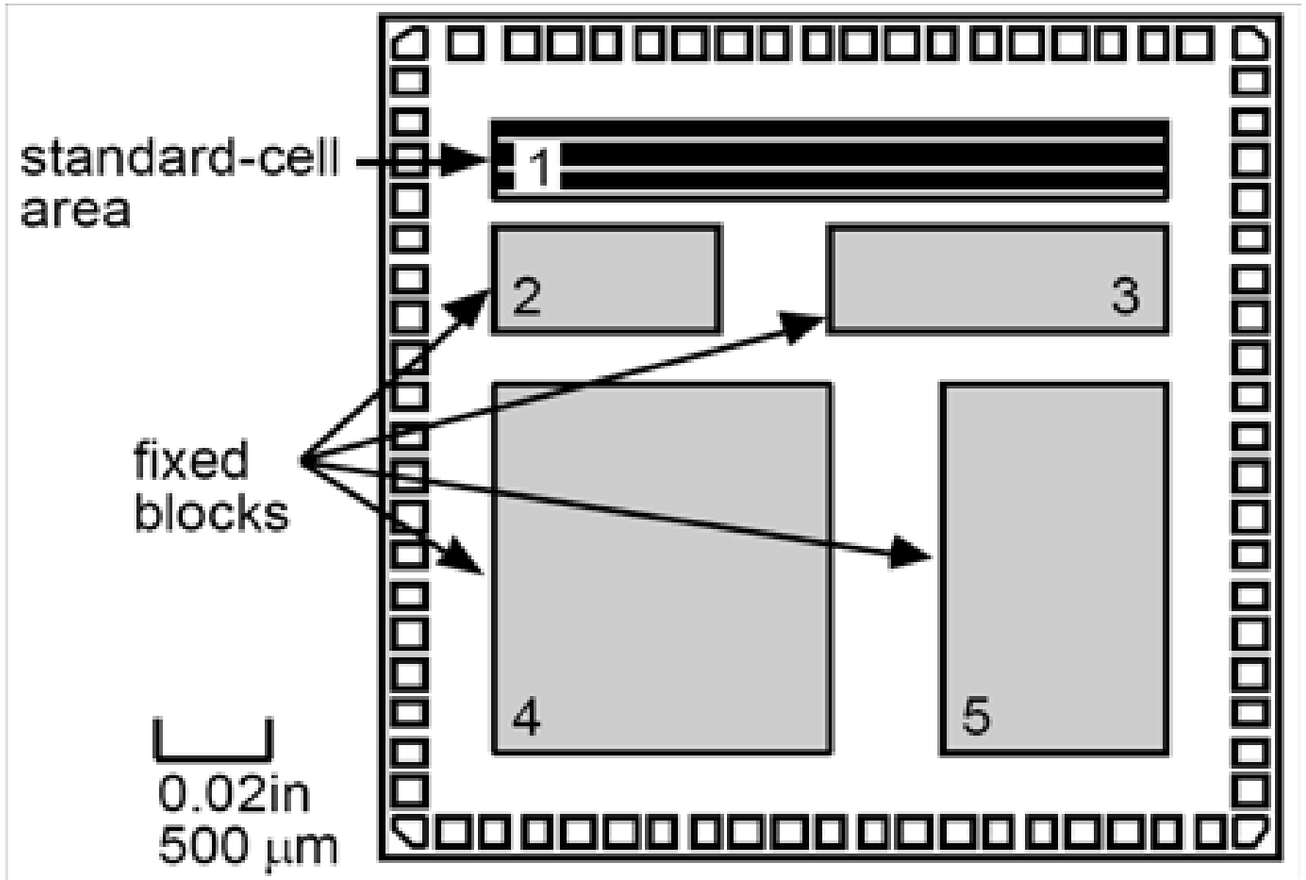
# Basic FPGA structure




- Regular; functionality through configuration
  - Refined variations abound…

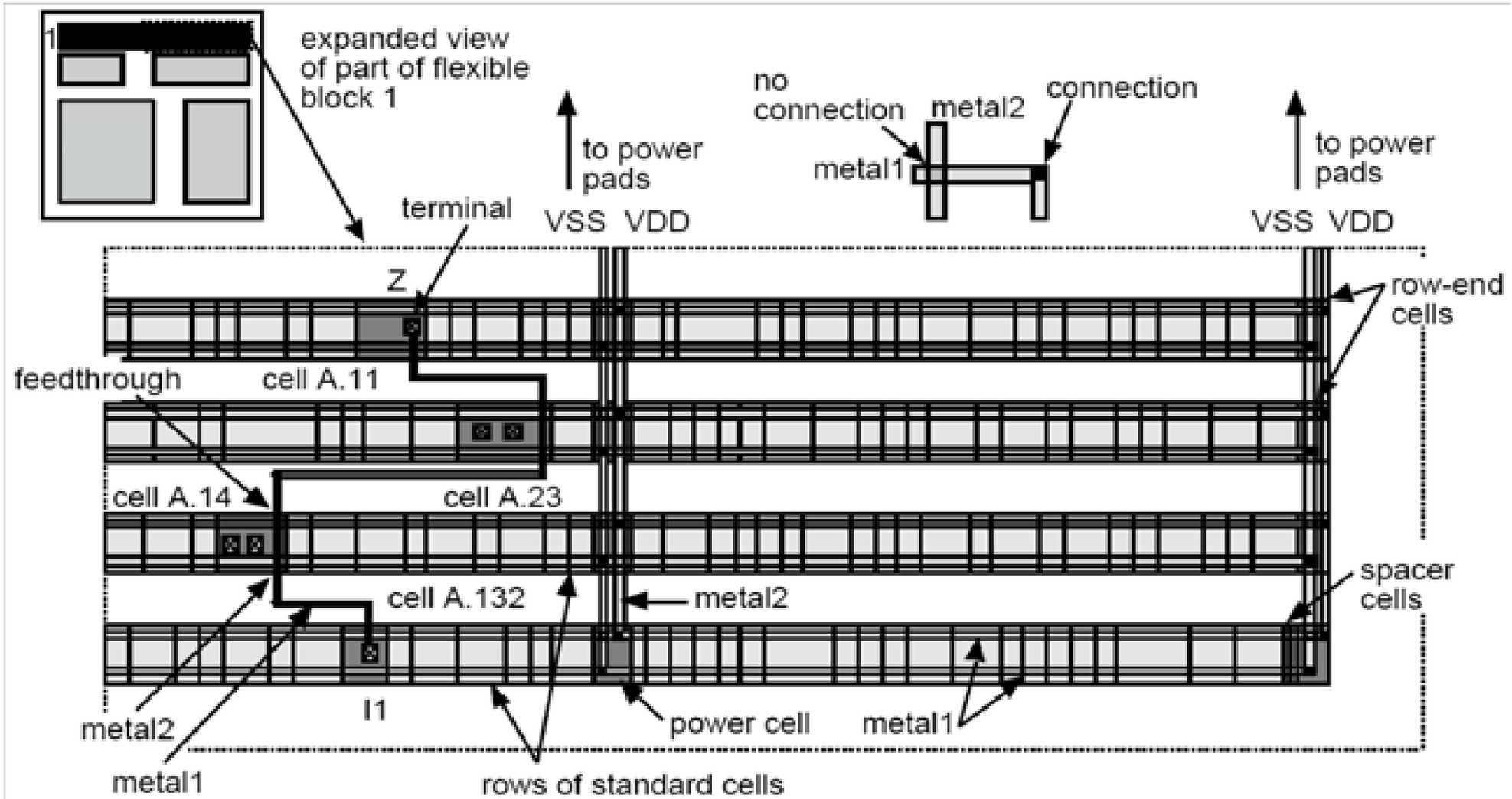
# Basic ASIC structure

- Uses pre-designed, fixed blocks (datapaths, memories, etc)
  - In-house / licensed
- “Glue logic” done with standard cells
  - Licensed libraries
- All placed on “empty canvas”



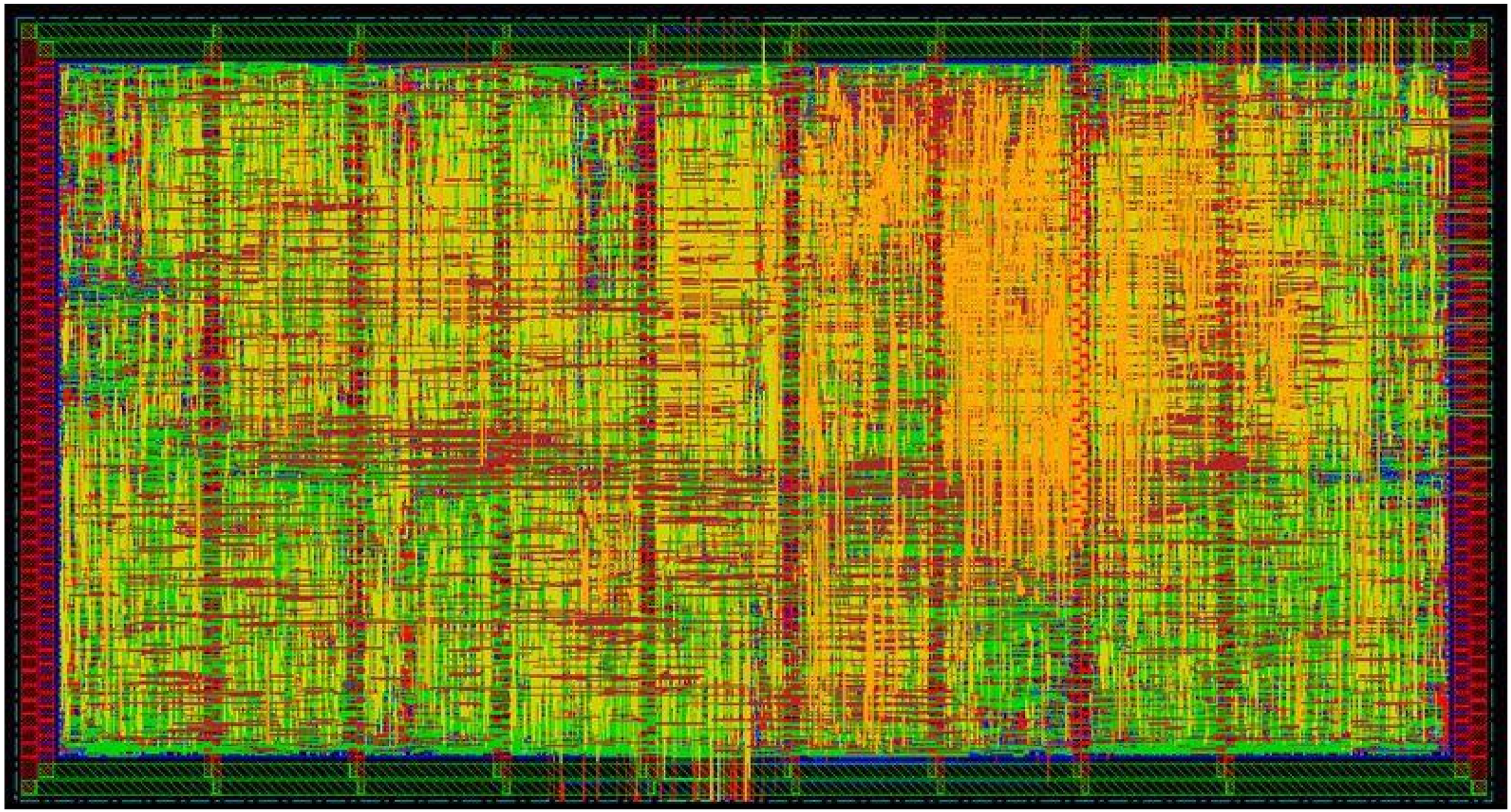

source: M J S Smith

# Standard-cell areas




- Each cell may contain single logic gate or flip-flop
- Computer support needed to place and interconnect cells

# Standard-cell processor layout

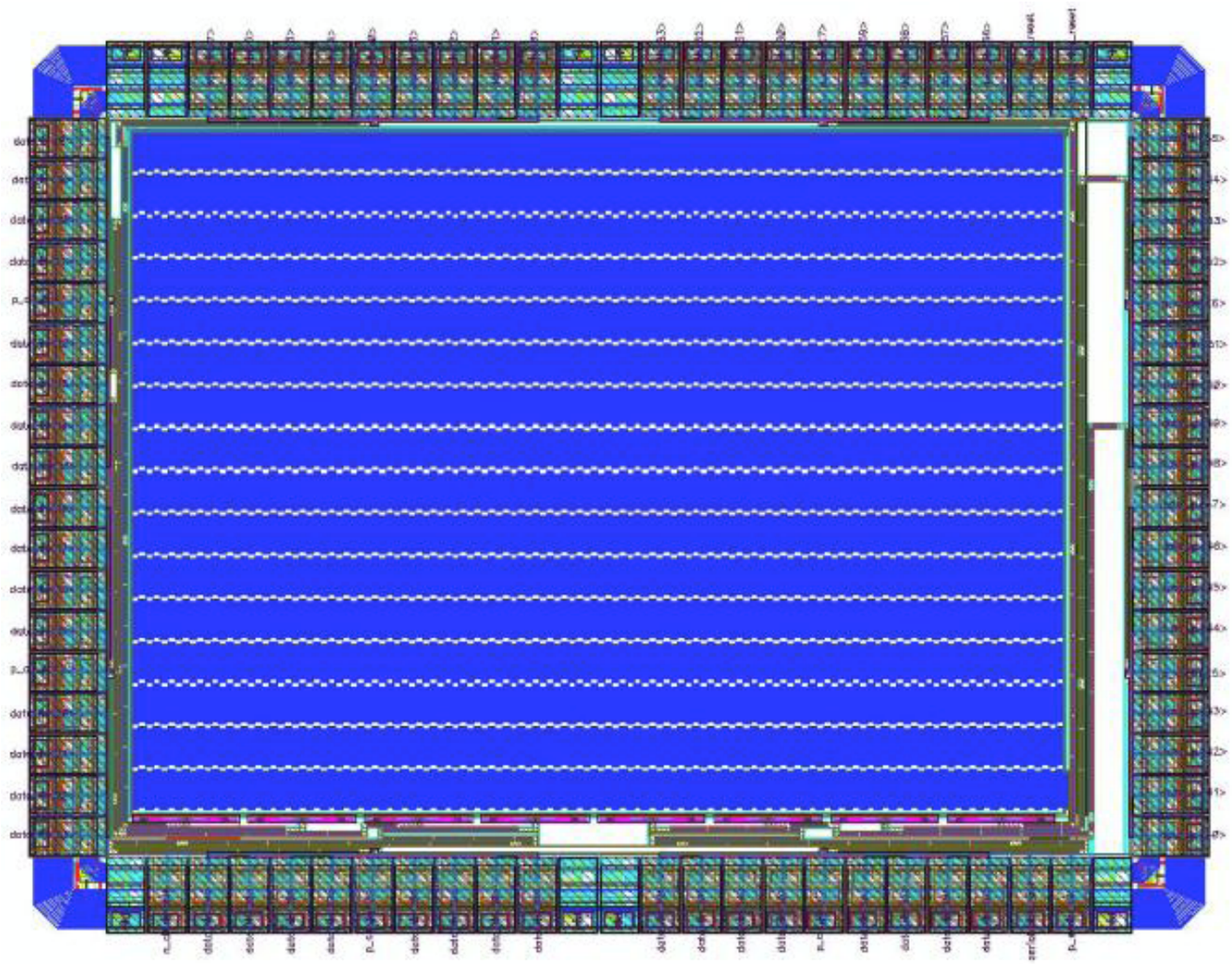


- Not manually doable…

# “Full-custom” ASIC

- Even lowest-level cells designed in detail
- Common for analog design
- Rarely used for digital design… except:
  - Extreme performance / area requirements
  - Critical blocks
  - …and for design of FPGA substrates!

# Cross-correlator ASIC (full-custom)

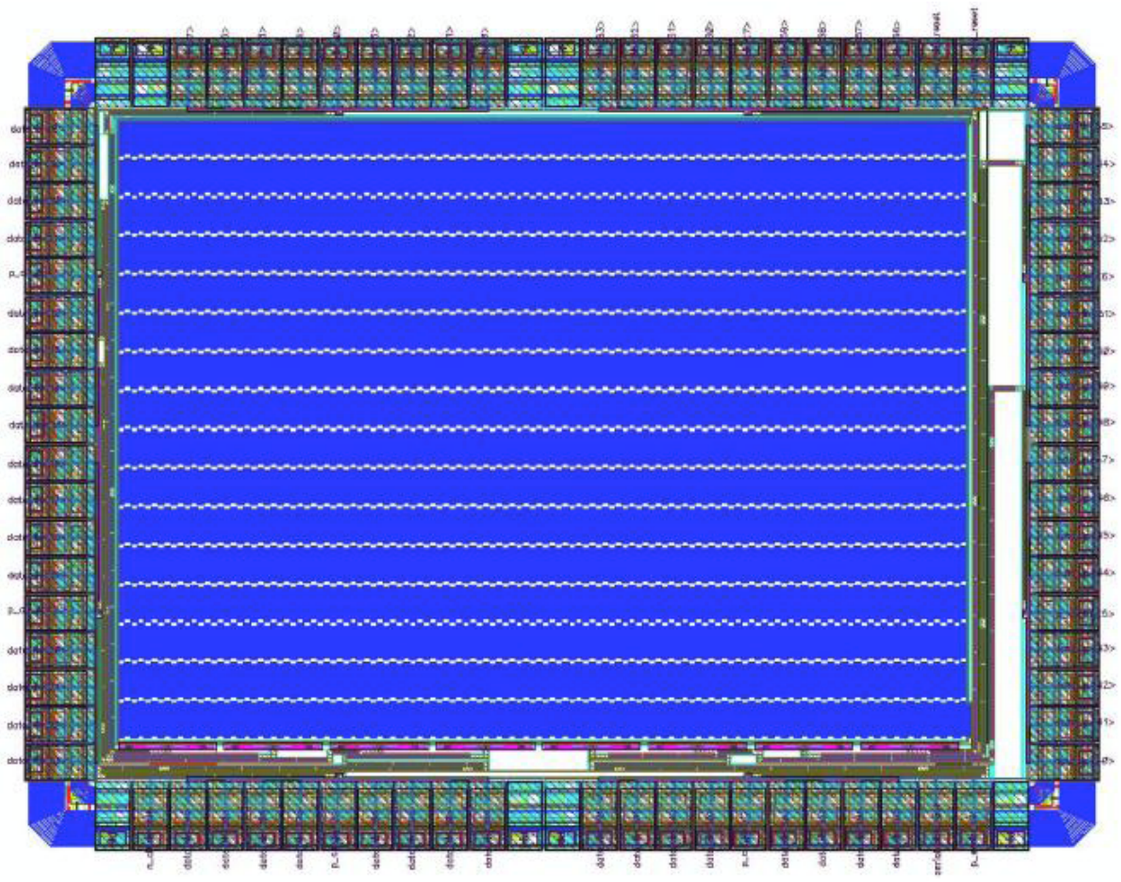


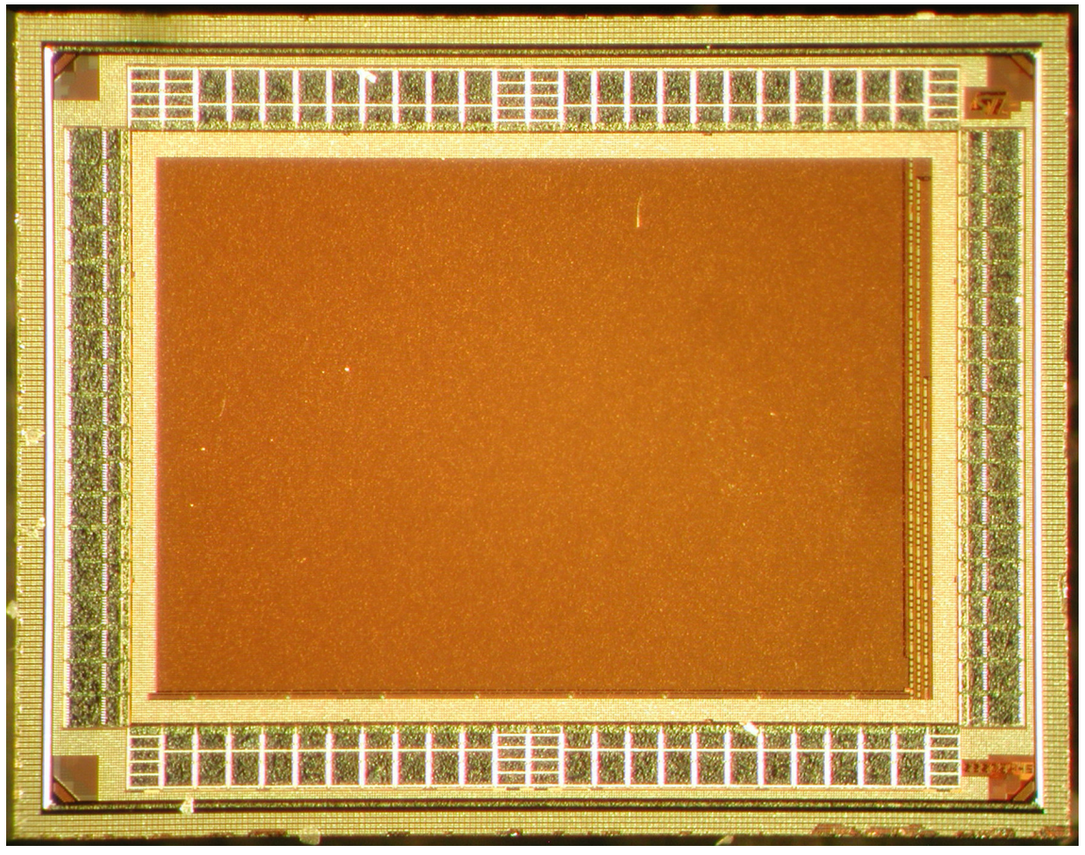
- 3 mm$^2$ in 65-nm process
  - ~3M transistors, ~3 GHz
- ~1 man-year, SEK 250K for prototype mfg



[Erik Ryman, ongoing PhD project]

# Cross-correlator ASIC (full-custom)





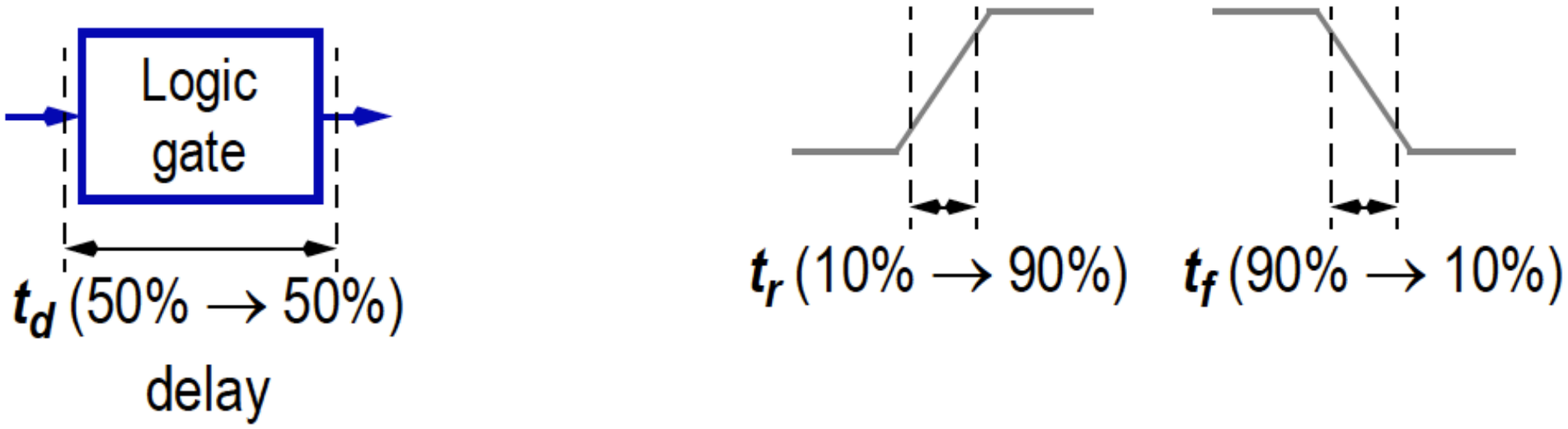
- 3 mm$^2$ in 65-nm process
  - ~3M transistors, ~3 GHz
- ~1 man-year, SEK 250K for prototype mfg



[Erik Ryman, ongoing PhD project]

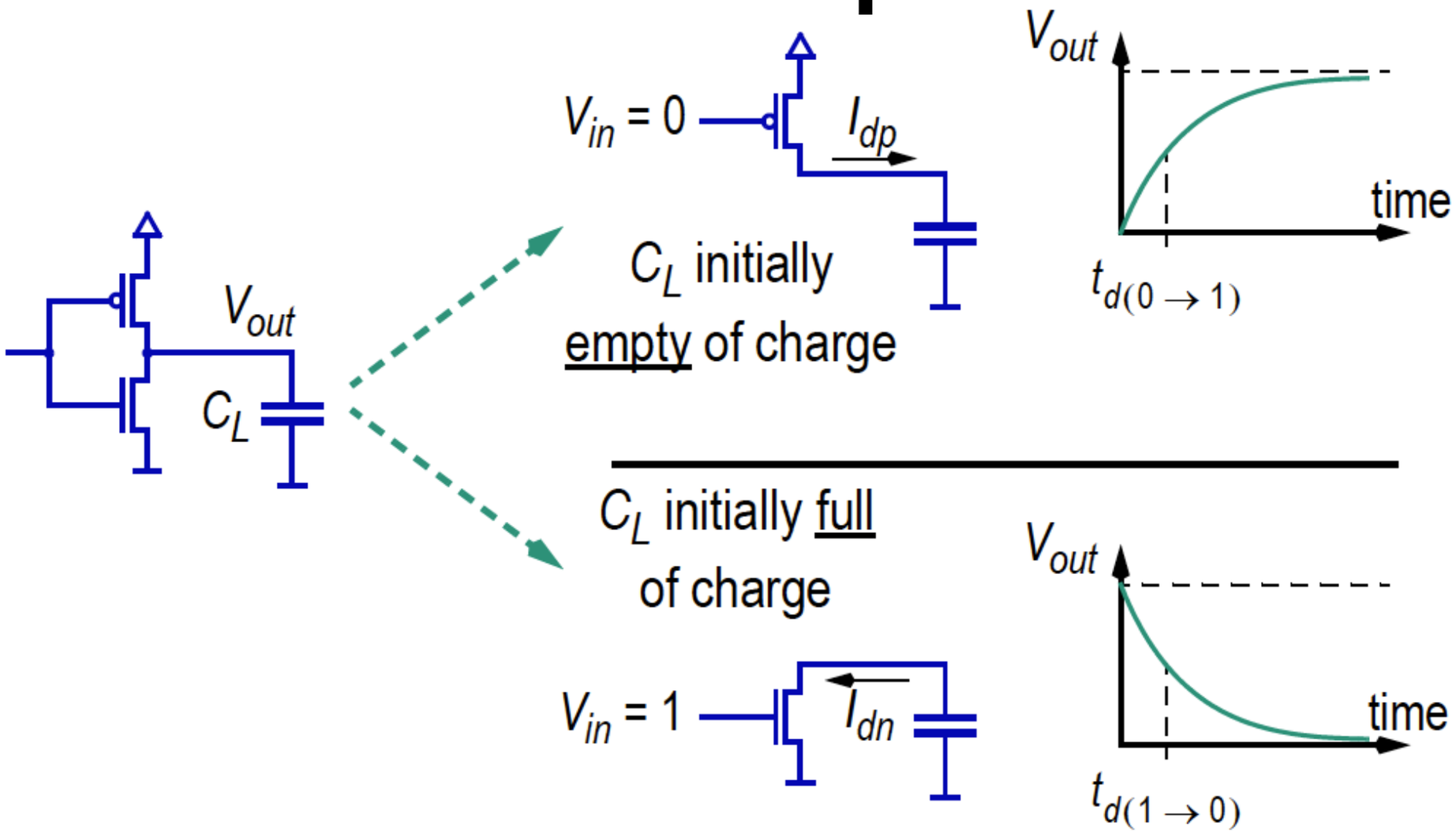
# Performance

# Time



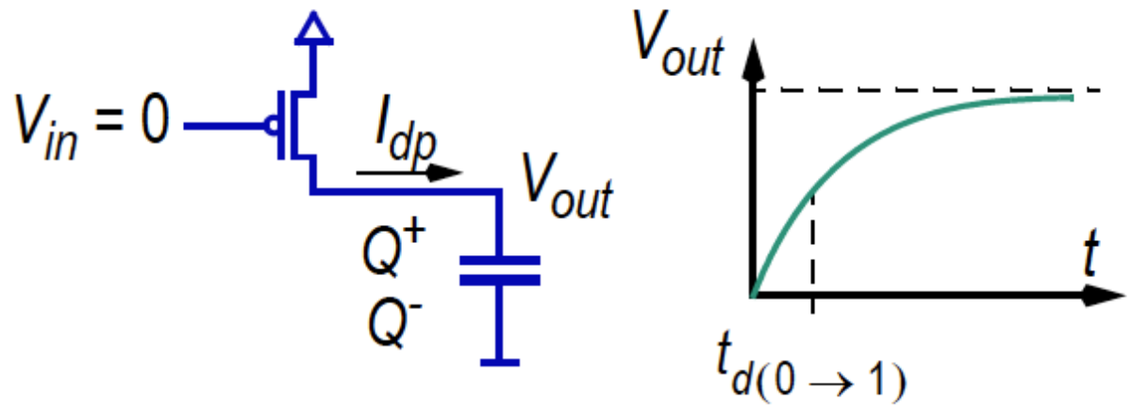

- Delays, transition times
  - Determine computation speed

# CMOS speed




- Capacitive load $C_L$
  - Consists of other-gate inputs, stray caps from interconnects

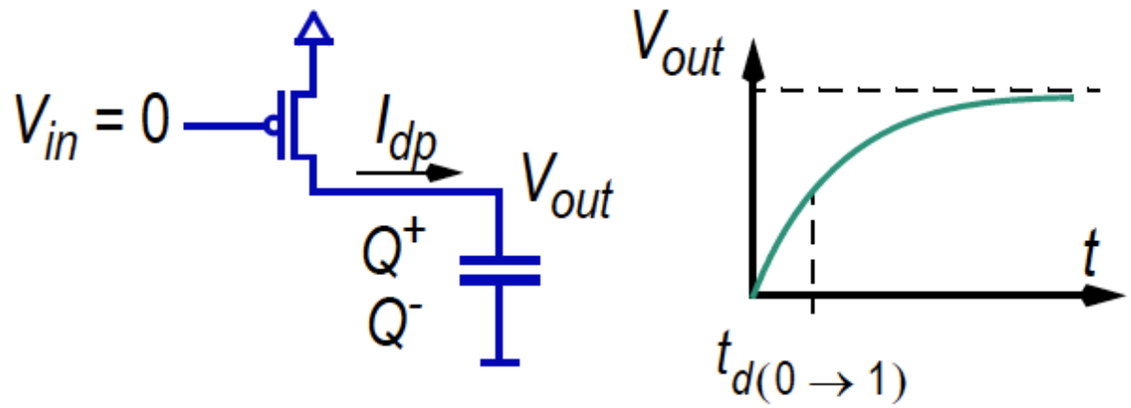
# CMOS speed, cont.




$$t_{d(0 \rightarrow 1)} = \frac{C_L}{k \cdot V_{DD}}$$

- $t_d \sim Q / I_{dp}$; $Q \sim C_L V_{DD}$; $I_{dp} \sim V_{DD}^2$
- k is property of transistor
- In an ASIC, $C_L$ may be minimized and k may be tuned
  - High performance possible!

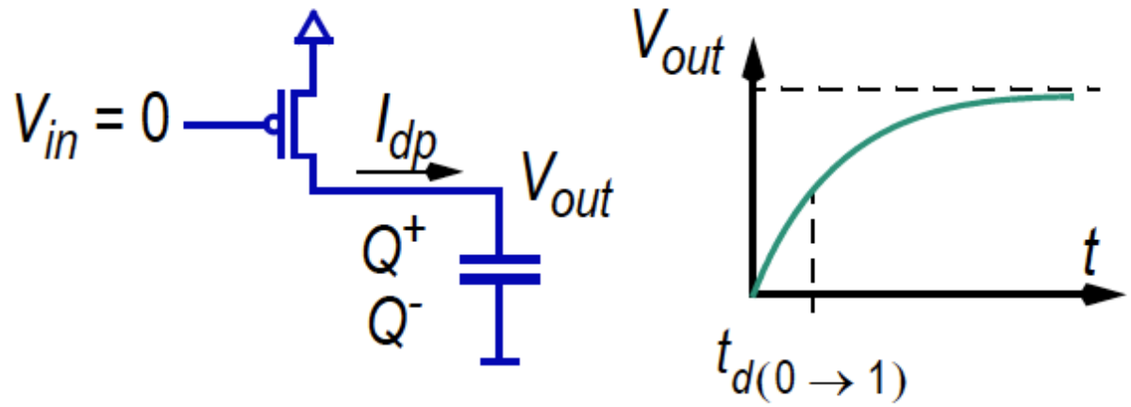
# CMOS speed, cont.




$$t_{d(0 \rightarrow 1)} = \frac{C_L}{k \cdot V_{DD}}$$

- $t_d \sim Q / I_{dp}$; $Q \sim C_L V_{DD}$; $I_{dp} \sim V_{DD}^2$
- k is property of transistor
- In an ASIC, $C_L$ may be minimized and k may be tuned
  - High performance possible!

# CMOS speed, cont.

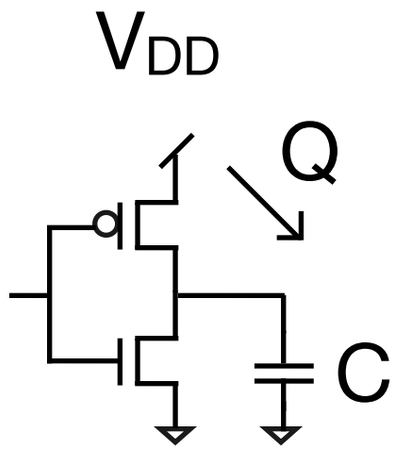



$$t_{d(0 \rightarrow 1)} = \frac{C_L}{k \cdot V_{DD}}$$

- $t_d \sim Q / I_{dp}$; $Q \sim C_L V_{DD}$; $I_{dp} \sim V_{DD}^2$
- k is property of transistor
- In an ASIC, $C_L$ may be minimized and k may be tuned
  - High performance possible!

# CMOS energies

- Again, use inverter as example/model
- Charge C from 0 to $V_{DD}$
- Charge injected: $Q = C \cdot V_{DD}$

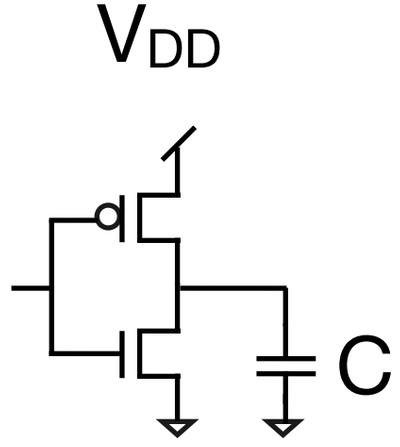



- Energy injected: $E_{inj} = Q \cdot V_{DD} = CV_{DD}^2$
- Energy dissipated: $Q \cdot V_{avg} = Q \cdot V_{DD} / 2 =$
  $= CV_{DD}^2 / 2$
- Energy stored (on C): $CV_{DD}^2 - CV_{DD}^2/2 =$
  $= CV_{DD}^2 / 2$

# Energy vs. power

- Stored energy $CV_{DD}^2$ / 2 dissipated during <u>discharge</u>
  - All of $E_{inj}$ dissipated each charge/discharge cycle!
- Assume f charge/discharge cycles per unit time:

$$P = f \cdot C \cdot V_{DD}^2$$




- Several gates:

$$P_{tot} = V_{DD}^2 \cdot \sum(f \cdot C)$$

- Again: C may be minimized in ASIC; low power possible

# FPGA vs ASIC: power vs performance

- FPGA flexibility requires many alternative signal routes
  - … so more multiplexers, buses, etc
  - … which add capacitive load
- FPGA logic density is worse due to unused alternatives
  - … so wires are longer
  - … so even more capacitance to be driven
- Higher capacitance hurts power and speed

# FPGA vs ASIC: power vs performance

- FPGA flexibility requires many alternative signal routes
  - … so more multiplexers, buses, etc
  - … which add capacitive load
- FPGA logic density is worse due to unused alternatives
  - … so wires are longer
  - … so even more capacitance to be driven
- Higher capacitance hurts power and speed

FPGA: 10x–100x higher power at same performance

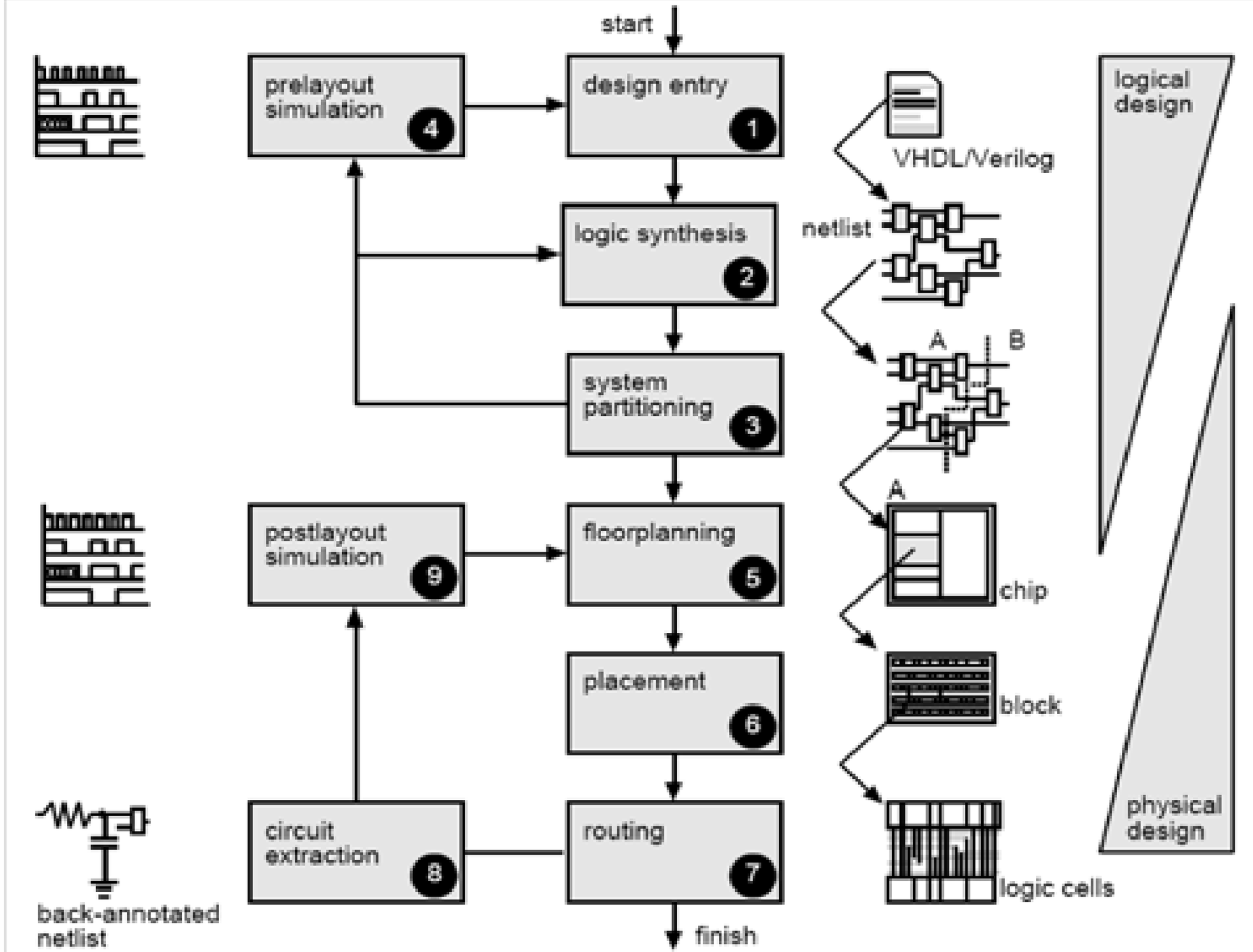
# Design flows

# ASIC vs FPGA

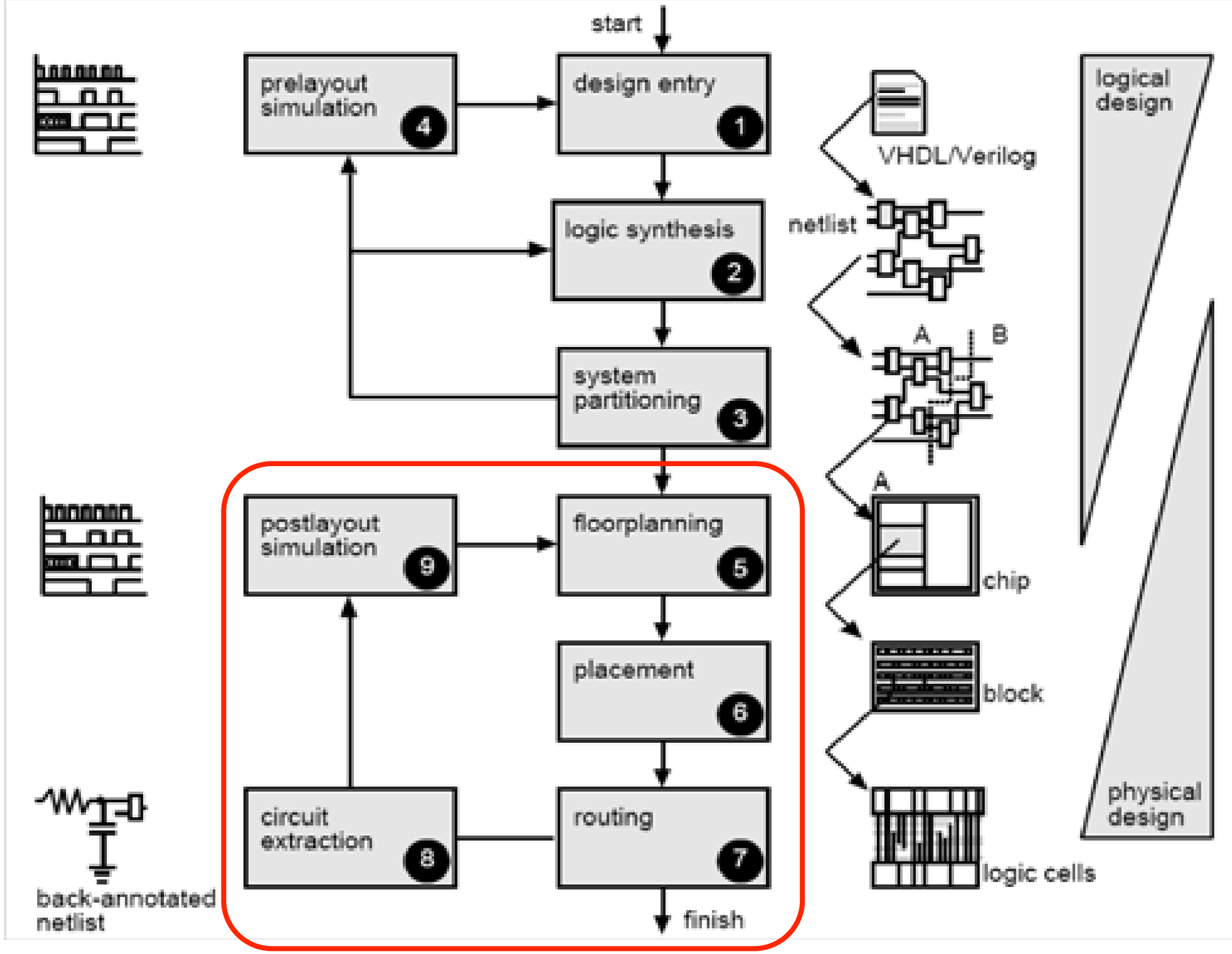
- Both: specification in VHDL (etc.)
- FPGA: designers need not consider circuits; no voltages and currents
  - Simulations etc. on logic level
- ASIC: designers handle circuit details
  - Carry out and review circuit simulations

- Larger (expensive) tool suites for ASICs

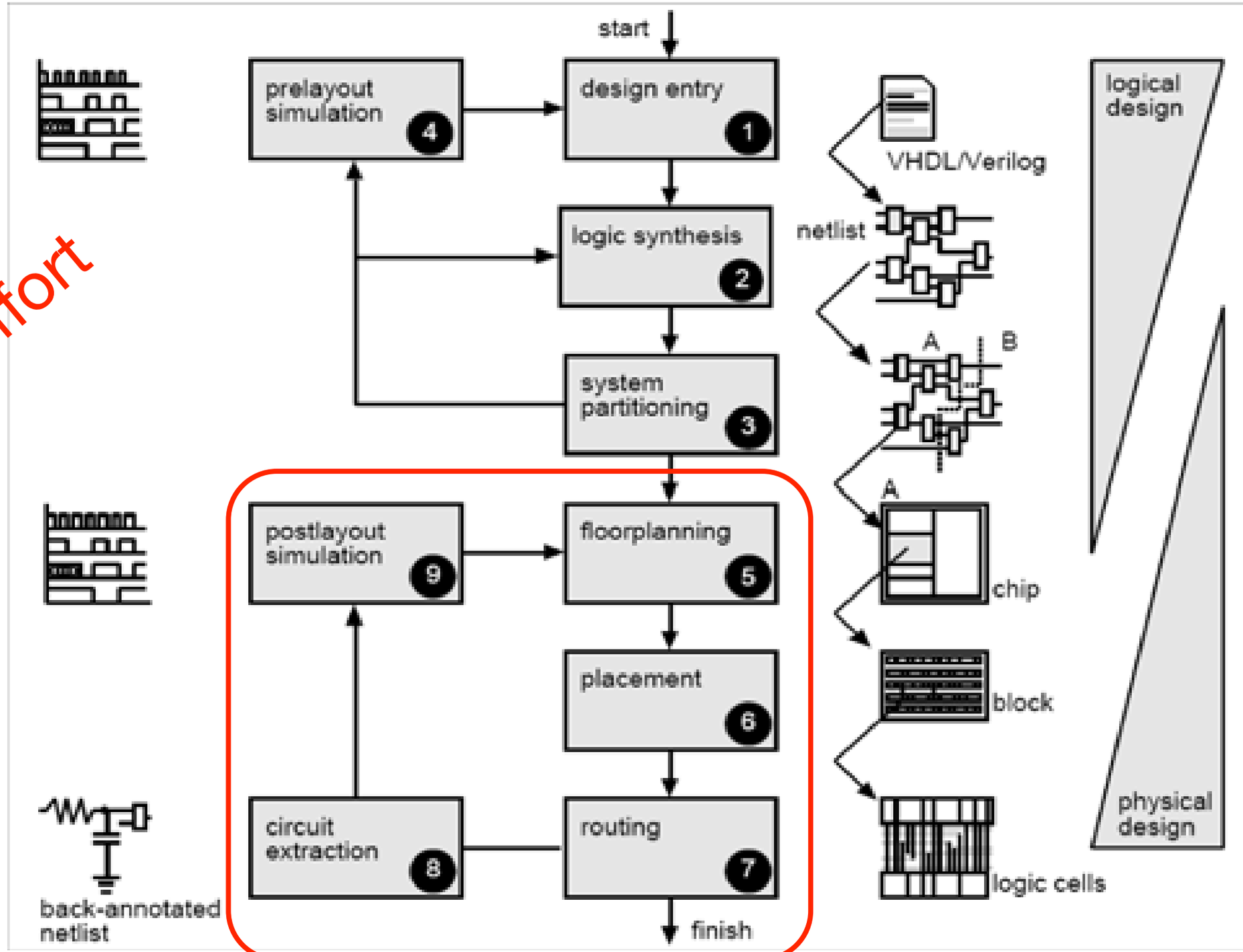
# ASIC design flow



# ASIC design flow

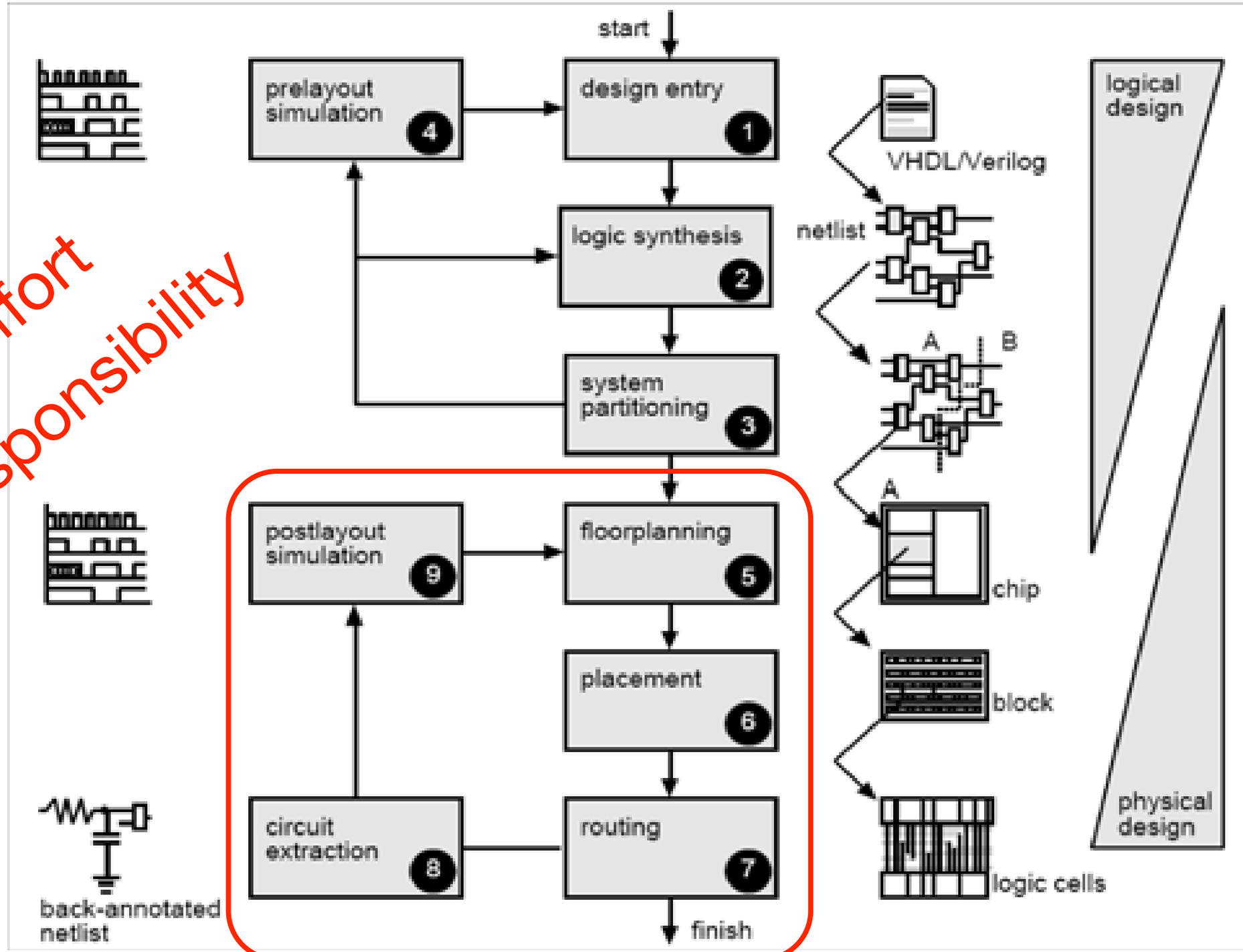


# ASIC design flow




More effort

# ASIC design flow



# ASIC vs FPGA, cont.

- FPGA: edit-compile-test cycle, similar to software development (minutes to hours)
- ASIC: fabrication takes months, costs $$$, development typically involves large teams

- ASIC design a much more expensive prospect
  - …especially when there are errors…
  - …but once done, cost per part is much lower

# ASICs and FPGAs

- Technologies may complement each other in larger project
  - Prototype / first version of design in FPGA
  - Re-spin as ASIC if volumes motivate

# What about the future?

# Trends

1. Smaller transistors
2. More transistors
3. Higher clock frequencies
4. Higher power

- Smaller, more numerous, faster! (and hotter…)

# 1. Transistor size

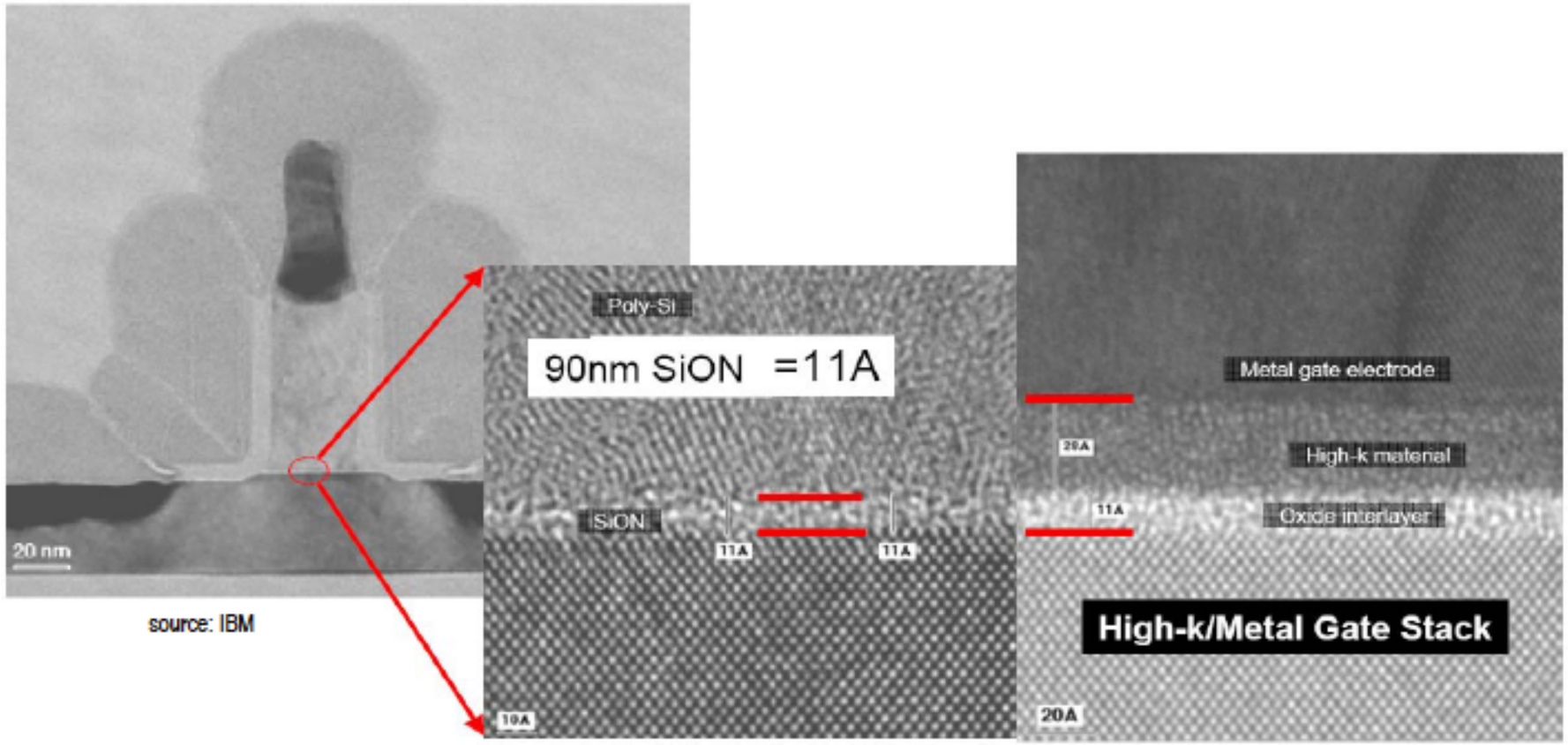
- New “process generation” every few years
- Now: ~20-nm* transistors (channel length)
  - Leakage currents
  - Thin gate oxides (~15 Å)
    - Lower voltages
  - Variability (process, voltage, temperature)
  - Model problems

# 1. Transistor size

- New “process generation” every few years
- Now: ~20-nm* transistors (channel length)
  - Leakage currents
  - Thin gate oxides (~15 Å)
    - Lower voltages
  - Variability (process, voltage, temperature)
  - Model problems

*) “7-nm process” does not refer to transistor dimensions…

# Thin oxides

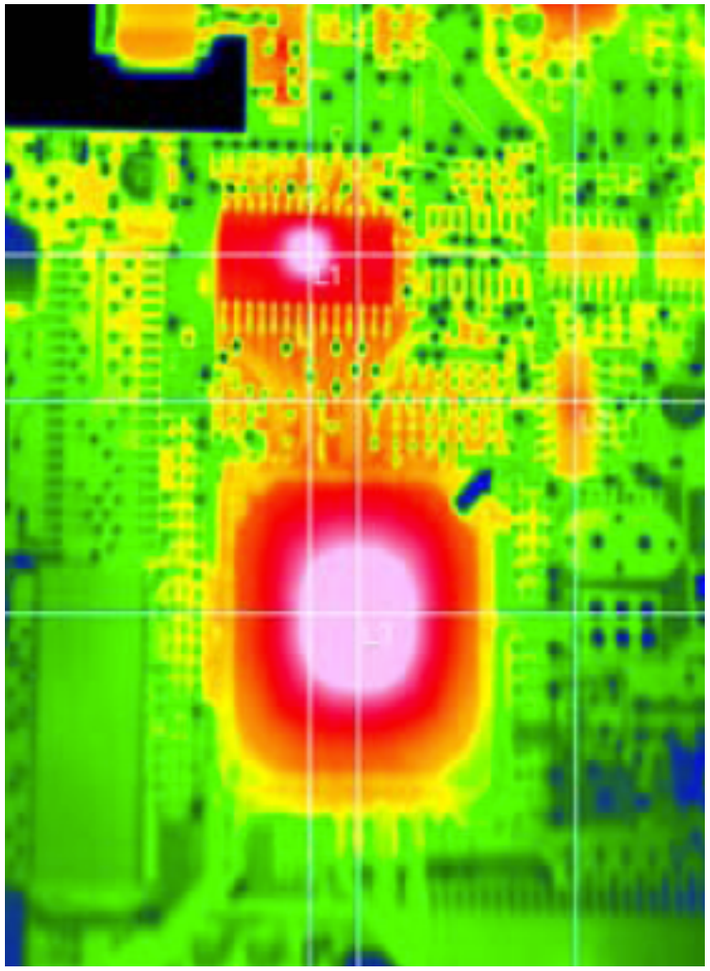


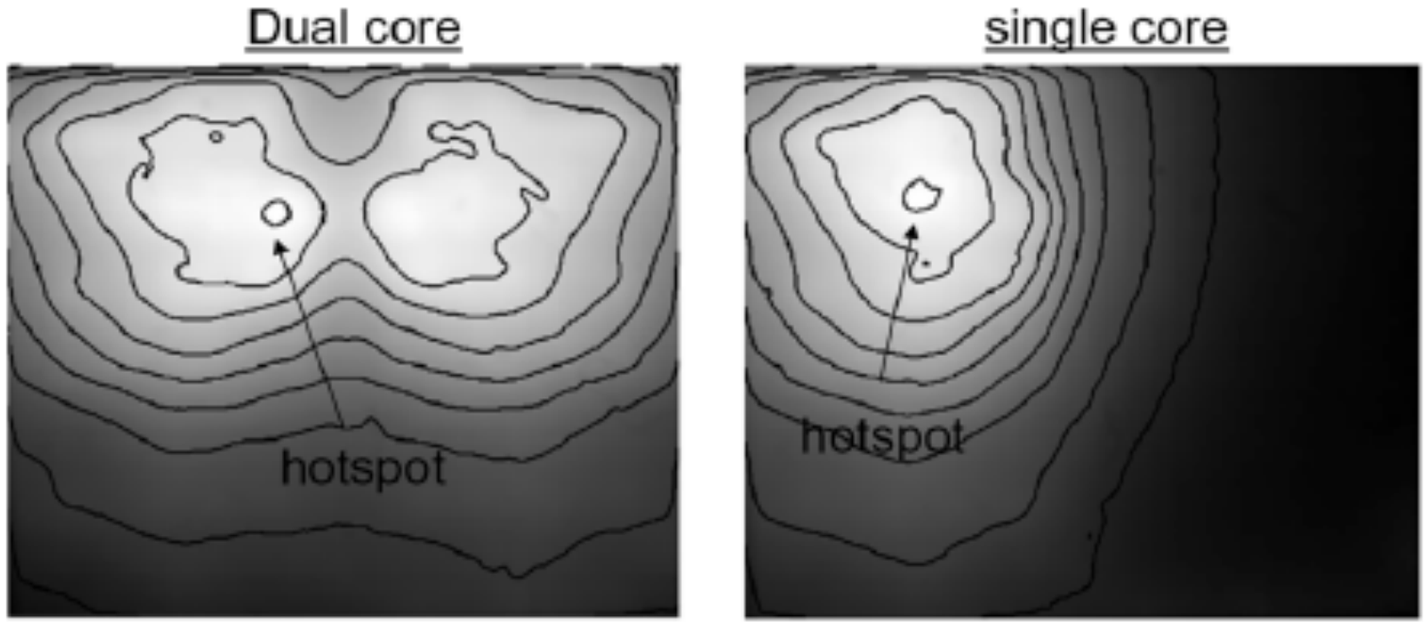

**90nm Gate Dielectric:**
**Tinv = 19A**
**ToxGL = 11A**

**High-k/Metal Gate Stack:**
**Tinv = 15.5A**
**ToxGL = 20A**

- Few layers of atoms!

# Variability (T)






- Leakage, power, heat, higher leakage, …
  - Positive feedback :-(

# 2. Transistor count

- Now: billions of transistors on single chip
  - Still increasing* (x2 per process generation; cf. Moore's Law)
    - Replicated blocks (memories, cores)
      - Re-use of designer effort
    - Networks-on-chip for communication
    - Redundancy

*) End in sight though!

# 3. Clock frequency

- Rising until ~10 years ago
- Now: 3–5 GHz
  - No significant further increase expected
    - Transistors still get faster with scaling, wires don’t
- Larger chips, difficult synchronization
  - On-chip wavelength @ 5 GHz: ~30mm?

# 4. Power

- Now: < 150 W per chip (at reasonable cost)
  - Packaging technology breakthrough required for significant increase
- Limitations:
  - Supply currents (1 V means ~150 A!)
  - Cooling (35 W / $cm^2$)
- Control and regulate voltage and frequency

# Future?

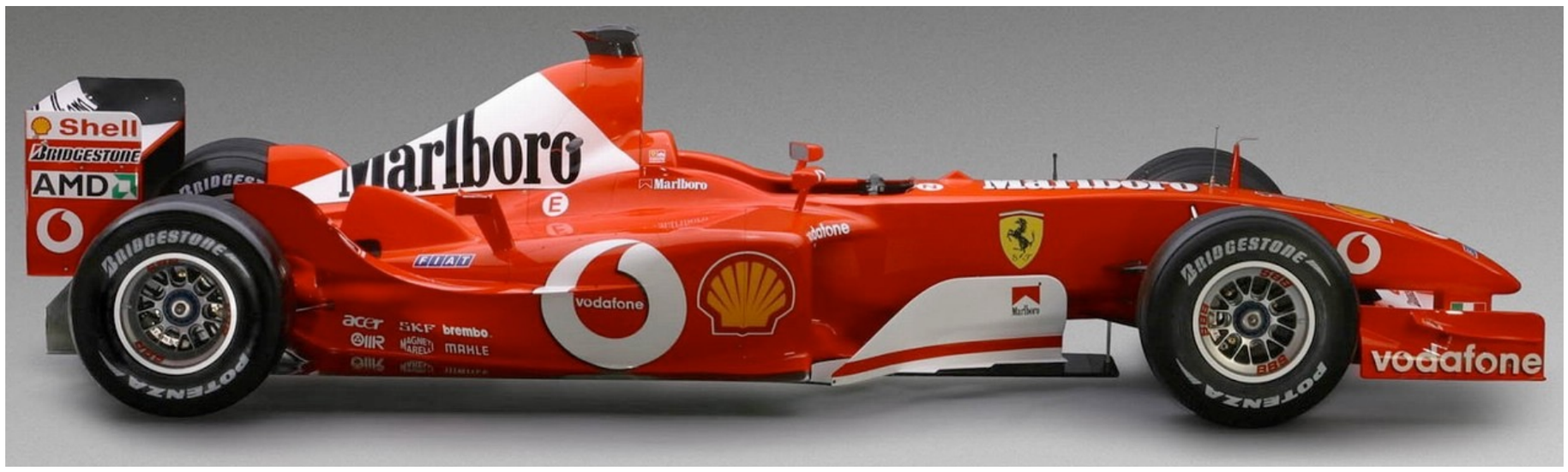
- Transistor size limit
  - 20 nm? 10 nm?
- Bigger design blocks
  - Gates, adders/memories, processor cores, … ?
  - Software aspects add to complexity
- Specialization
  - Several cost/performance points

# Summary

- ASIC design: a bigger challenge!
  - More design freedom
  - Higher performance / lower power
  - Higher cost
  - More chances for expensive mistakes

# Summary

- ASIC design: a bigger challenge!
  - More design freedom
  - Higher performance / lower power
  - Higher cost
  - More chances for expensive mistakes



larssv@chalmers.se

# Room 4477

larssv@chalmers.se

# Room 4477

## Embedded Electronics System Design (MPEES)