Introduction

VLSI CAD

Compiled by Oleg Venger

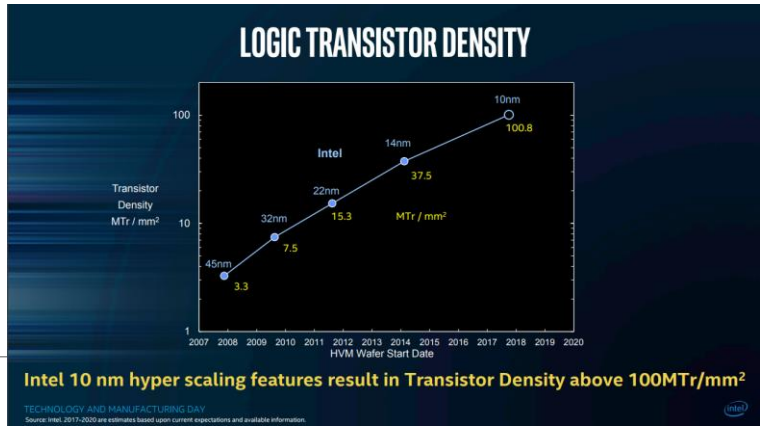
Course Objective

- To get an idea about chip design concepts
- To get an idea about chip design problems that CAD tools help to solve
- To get an idea about emerging design and CAD challenges
- To get an idea about algorithms and data structures used in VLSI CAD tools

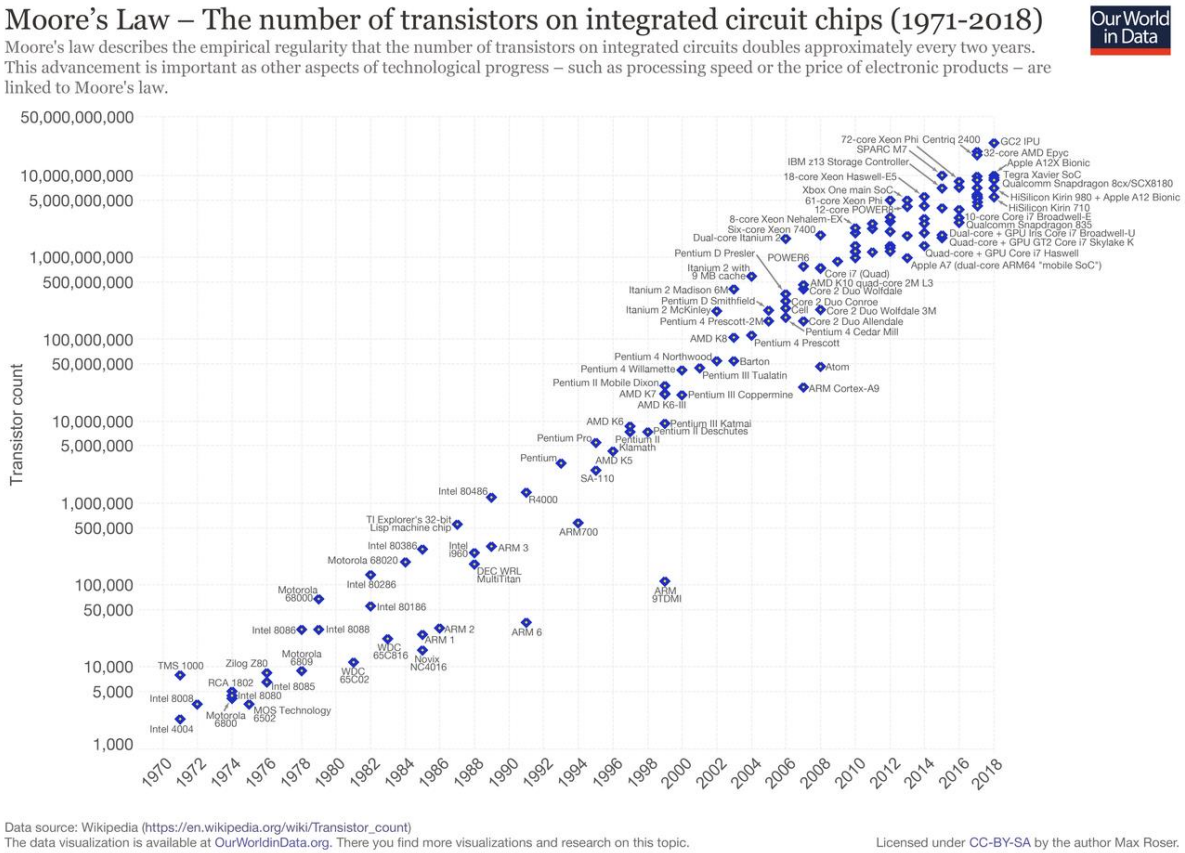
Course outline

1. Introduction
2. Basic algorithms on graphs
3. Static timing analysis
4. Routing
5. Placement
6. Buffer insertion
7. Computational Boolean algebra
8. BDDs
9. Two-level logic synthesis
10. Multi-level logic synthesis
11. SAT
12. Technology mapping

Trends in microelectronics



- Improvements in device technology
  - Smaller circuits
  - Higher performance
  - More devices on a chip
- Higher degree of integration
  - More complex systems
  - Higher reliability challenges
  - Lower cost of computation
- Design considerations:
  - Use most recent technologies: to be competitive in performance
  - Reduce design cost: to be competitive in price
  - Time-to-market is critical: speedup design time

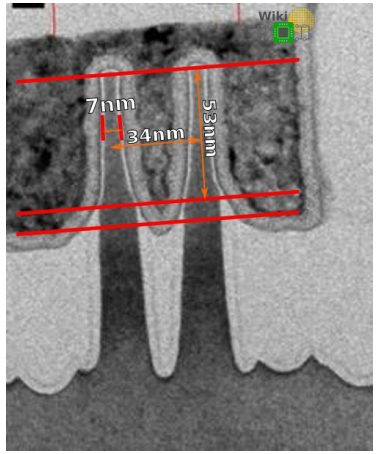


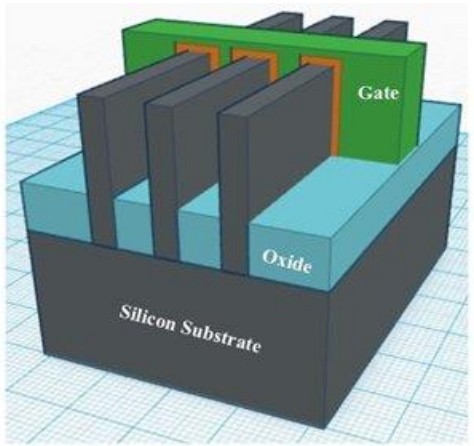
Feature size scaling. Intel history

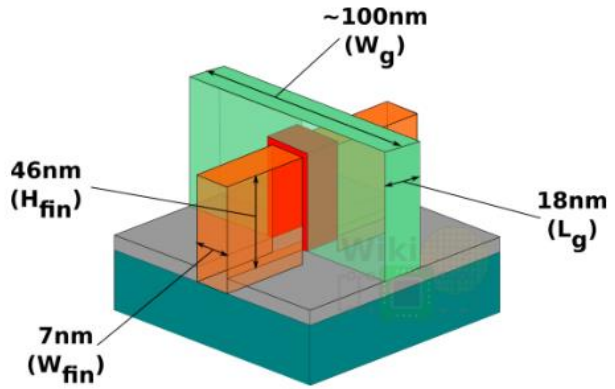
- There is physical limit to minimal printable critical dimension: $CD = k_1 \frac{\lambda}{NA}$. This imposes complex challenges for photolithography for nodes beyond 90nm
- Small devices also have physical problems: switching speed, leakage power, etc.
- Thin wires suffer from high resistivity and electromigration
- Over years, the layout has become more regular, and lot of materials research has been done to keep Moor's law alive

2003	90nm	Strained Si
2005	65nm	
2007	45nm	High-k and Metal Gate Transistor
2009	32nm	Wet lithography
2011	22nm	FinFET transistors
2014	14nm	Non-copper critical metal layers
2019	10nm	
2021	7nm	EUV

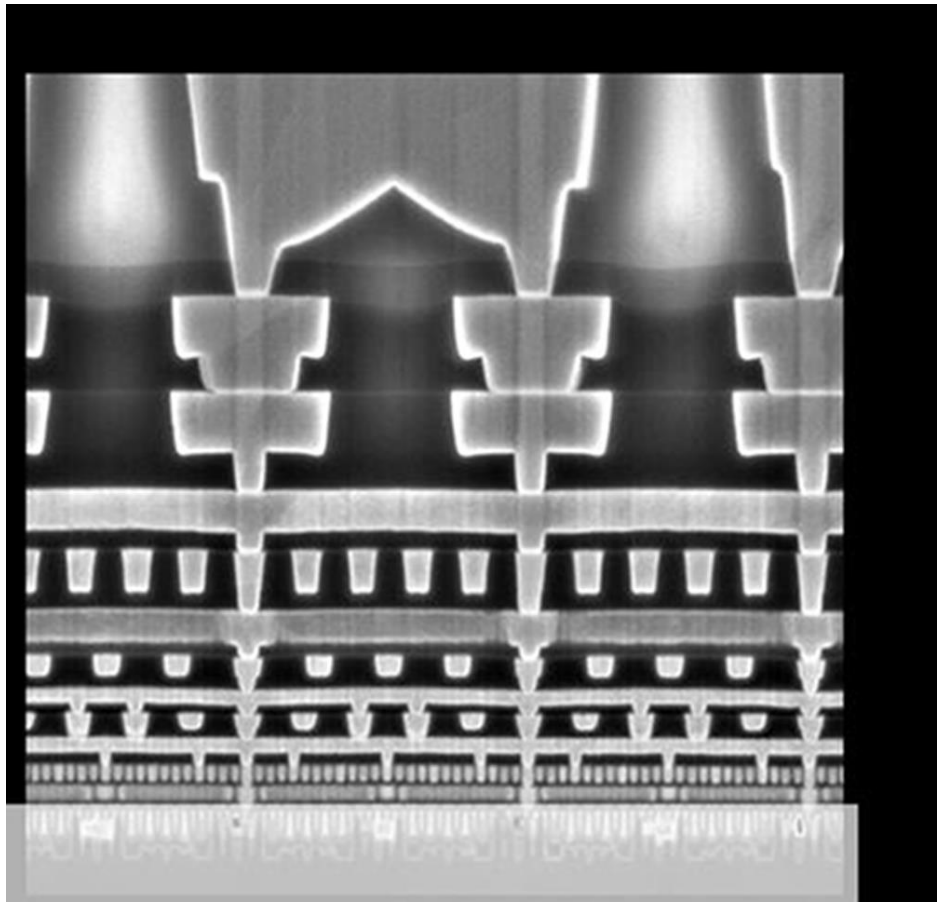
Intel's 10 nm process technology







Feature	Pitch, nm
Fin	34
Gate	54
M0	40
M1	36
M2-M4	44
M5	52
M6	84
M7-M8	112
M9-M10	160
TM0	1080
TM1	11K



Microelectronics design problems

- Intrinsic physical scaling limits
- Capital investment for fabrication
- Large-scale design management: use CAD design tools

Modeling abstractions

Architectural:

…

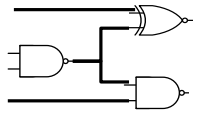
PC = PC+1;

FETCH (PC)

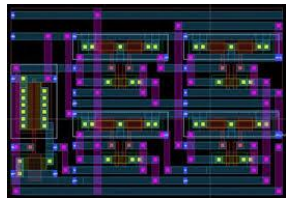
DECODE(INSTR)

…

Logic:

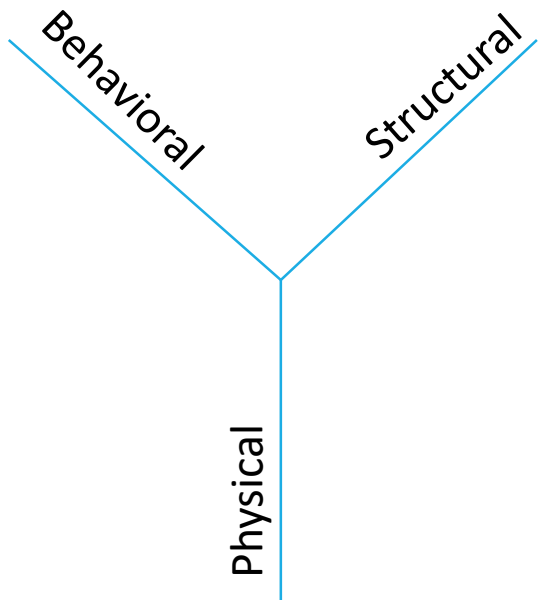


Physical:



- Architectural level:
  - Operations implemented by resources
- Logic level:
  - Logic functions implemented by gates
- Physical level:
  - Physics of devices modeled
  - Devices are geometrical objects

Modeling views



- Behavioral view:
  - Abstract function
- Structural view:
  - Interconnection of parts
- Physical view:
  - Physical objects with sizes and positions

Circuit optimization

- Performance
  - Delay and cycle time
  - Latency
  - Throughput
- Power consumption
- Area (yield and packaging cost)
- Manufacturing (yield)
- Testability

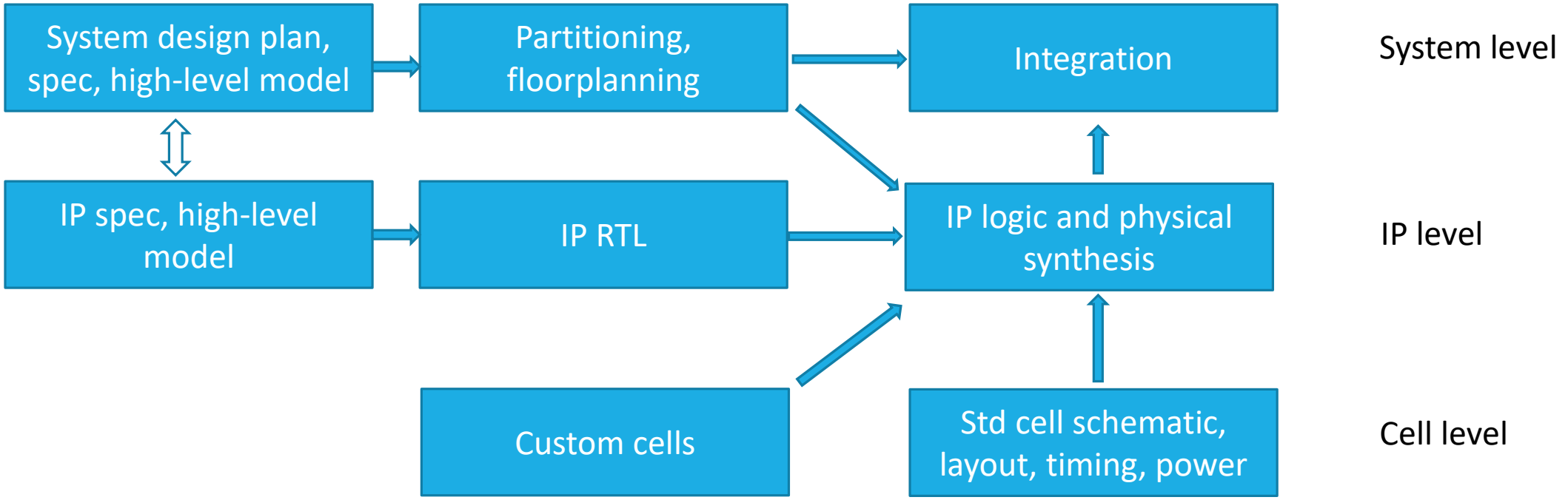
VLSI CAD

- CAD (Computer-Aided Design) tools: software that help designer in design process (wide design types)
- EDA (Electronic Design Automation) : software tools for designing electronic systems, e.g. integrated circuits)
- The tools work together in a design flow that designers use to make the design
- Can be seen as a compiler from algorithm to mask patterns

Designer roles

- Architects develop specifications
- Microarchitects define functional blocks
- Logic designers write detailed description of functional blocks (RTL)
- Circuit designers implement logic in transistors
- Mask designers draw layers to create transistors
- Integration team assembles components to a product
- Validation Engineers check for errors at every step
- Design Automation provides CAD tool support at every step

Generalized design flow

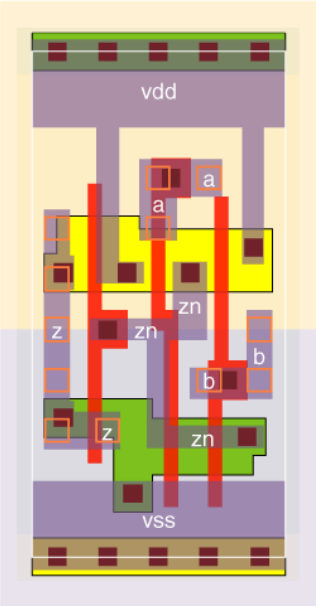


- Multiple iterations on each stage with transitions to previous stages
- Continuous validation (performance and functional)

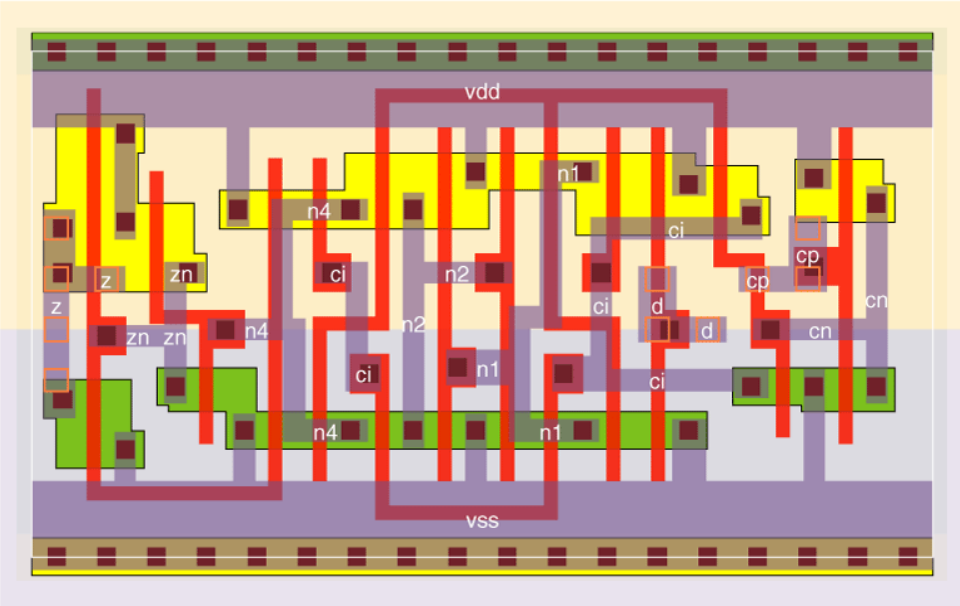
Standard Cell Library

- Standard cell libraries contain models for basic building blocks for digital design
- Library build process is called characterization. Includes comprehensive SPICE simulations, manual or automated schematic/layout creation
- Different CAD tools use different cell models (or views)
  - Logical view – for logic synthesis
  - Physical layout view – for physical synthesis
  - Timing, power, noise, parasitics – for performance validation

Cell layout examples



and2 layout



flipflop layout

Source: http://www.vlsitechnology.org

Cell timing example

```
cell (INVX1) {
  cell_footprint : inv;
  area : 129.6;
  cell_leakage_power : 0.0310651;
  pin(A)  {
    direction : input;
    capacitance : 0.0159685;
    rise_capacitance : 0.0159573;
    fall_capacitance : 0.0159685; }
  pin(Y)  {
    direction : output;
    capacitance : 0;
    rise_capacitance : 0;
    fall_capacitance : 0;
    max_capacitance : 0.394734;
    function : "(!A)";
    timing() {
      related_pin : "A";
      timing_sense : negative_unate;
      cell_rise(delay_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( \
          "0.147955, 0.218038, 0.359898, 0.922746, 1.76604", \
          "0.224384, 0.292903, 0.430394, 0.991288, 1.83116", \
          "0.365378, 0.448722, 0.584275, 1.13597, 1.97017", \
          "0.462096, 0.551586, 0.70164, 1.24437, 2.08131", \
          "0.756459, 0.874246, 1.05713, 1.62898, 2.44989"); }
      rise_transition(delay_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( ... ); }
      cell_fall(delay_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( ... ); }
      fall_transition(delay_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( ... ); }
    } /* end timing */
    internal_power() {
      related_pin : "A";
      rise_power(energy_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( ... ); }
      fall_power(energy_template_5x5) {
        index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
        index_2 ("0.025, 0.05, 0.1, 0.3, 0.6");
        values ( ... ); }
    } /* end internal_power */
  }   /* end Pin Y */
}     /* end INVX1 */
```

Source: www.cs.utah.edu/~elb/cadbook

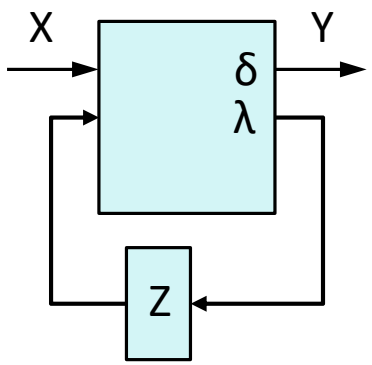
A good cell library features

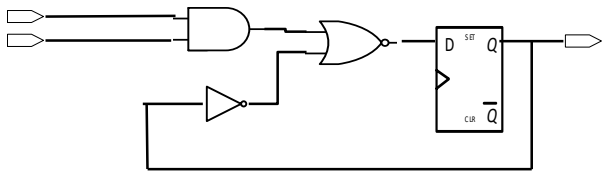
- Inverters, buffers, delay cells
- Simple logic functions (e.g. NAND2) and more complex (e.g. AOI, OAI)
- A variety of drive strengths for all cells
- Larger variety of drive strengths for inverters and buffers
- Cells with balanced rise and fall delays (clock gating cells, clock buffers)
- Flip-flops, latches both positive and negative edge, level sensitive; with optional set/reset pins
- Scan flops, latches
- DFM cells
- Multiple Process, Voltage and Temperature (PVT) Characterization Corners

CAD problems in standard library design

- Shorten input sequence for SPICE simulation
- Transistor placement automation
- Transistor routing automation
- Design for Manufacturing (DFM) (e.g. metal fill)
- Design rules check efficiency

Logic synthesis





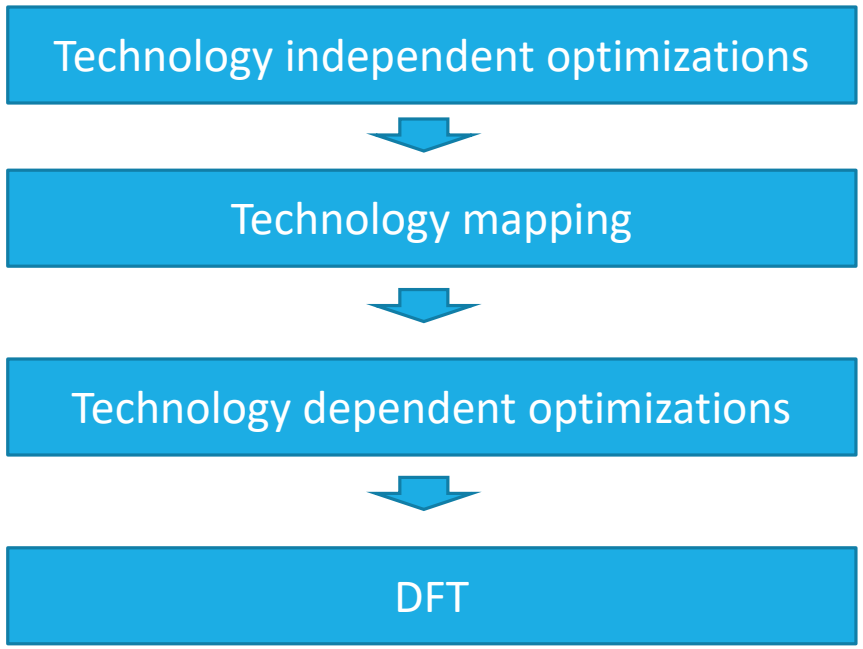
Given: Finite-State Machine F(X,Y,Z,λ,δ) where:

- X: input alphabet
- Y: output alphabet
- Z: set of internal states
- λ : X x Z -> Z - next state function
- δ : X x Z -> Y – output function

Produce: circuit C(G,W) where:

- G: set of circuit components (cells)
- W: set of wires connecting G

Logic Synthesis flow



Logic minimization and restructuring

Cover Boolean network by gates from target cell library

Timing optimization; physically driven optimizations

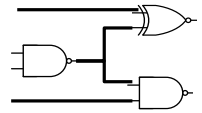
Test logic insertion

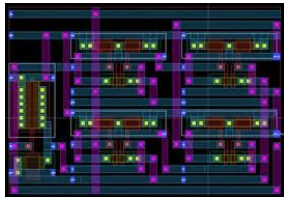
CAD problems in logic synthesis

- Minimization of Boolean network
- Optimization of FSM models
  - State minimization, encoding
- Synthesis of sequential multiple-level logic networks: optimization of area, delay, power, testability
- Equivalence verification
- Technology mapping: optimal selection of library cells
- Automatic test pattern generation (ATPG)

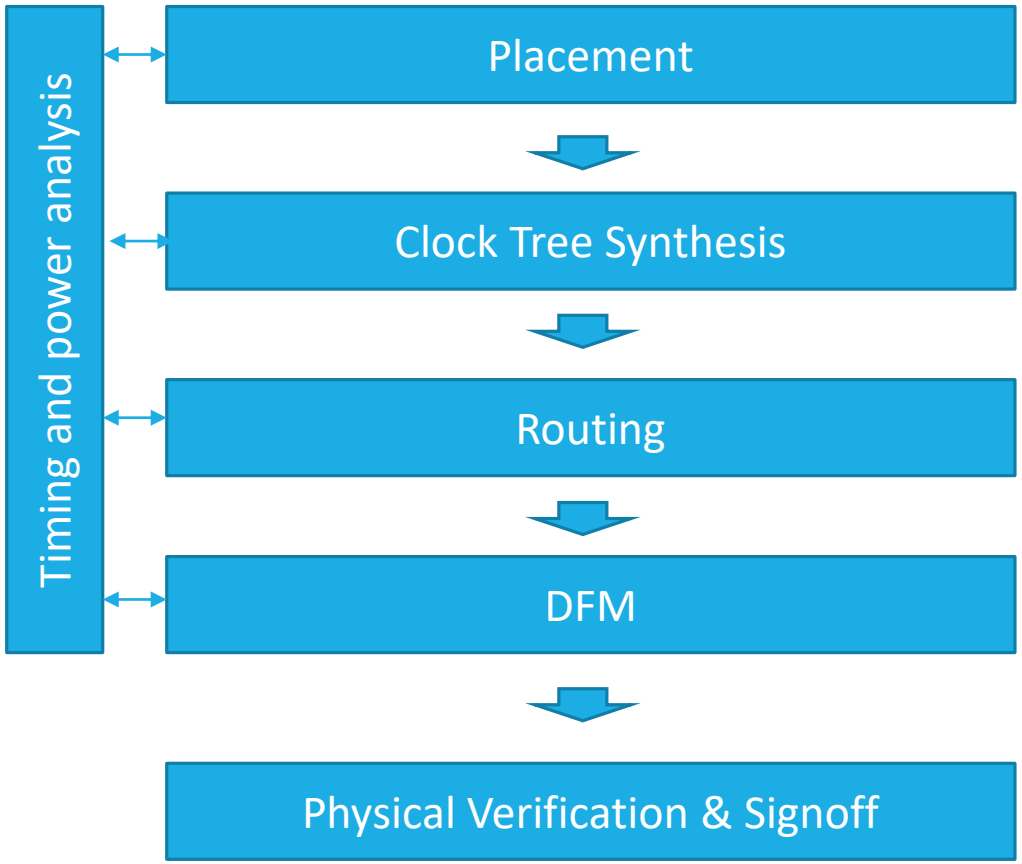
Physical synthesis

- Given:
  - Gate level netlist
  - Physical layout area
  - Timing constraints
- Produce legal layout:
  - Assign cell coordinates
  - Route wires
  - Satisfy design rules and constraints (electrical, DRC, DFM)





Physical synthesis flow



Decide cell locations; High fanout nets synthesis; post-placement optimization

Clock tree topology; buffers relocation and sizing

Make wires. Timing and congestion driven

Metal fill

LVS verification; DRC; signoff timing analysis

CAD problems in Physical Synthesis

- Timing analysis: timing graph processing to determine timing violations
- Buffering: high-fanout net synthesis
- Placement: assign netlist nodes location under complex optimization objective and constraints
- Routing: generate correct interconnect layout
- Gate sizing: select optimal drive strength for circuit cells
- Layout resilience to RTL changes
- Layout geometry rules check and violation detection
- … and more ….