SYNTHESIS FOR ADVANCED NODES

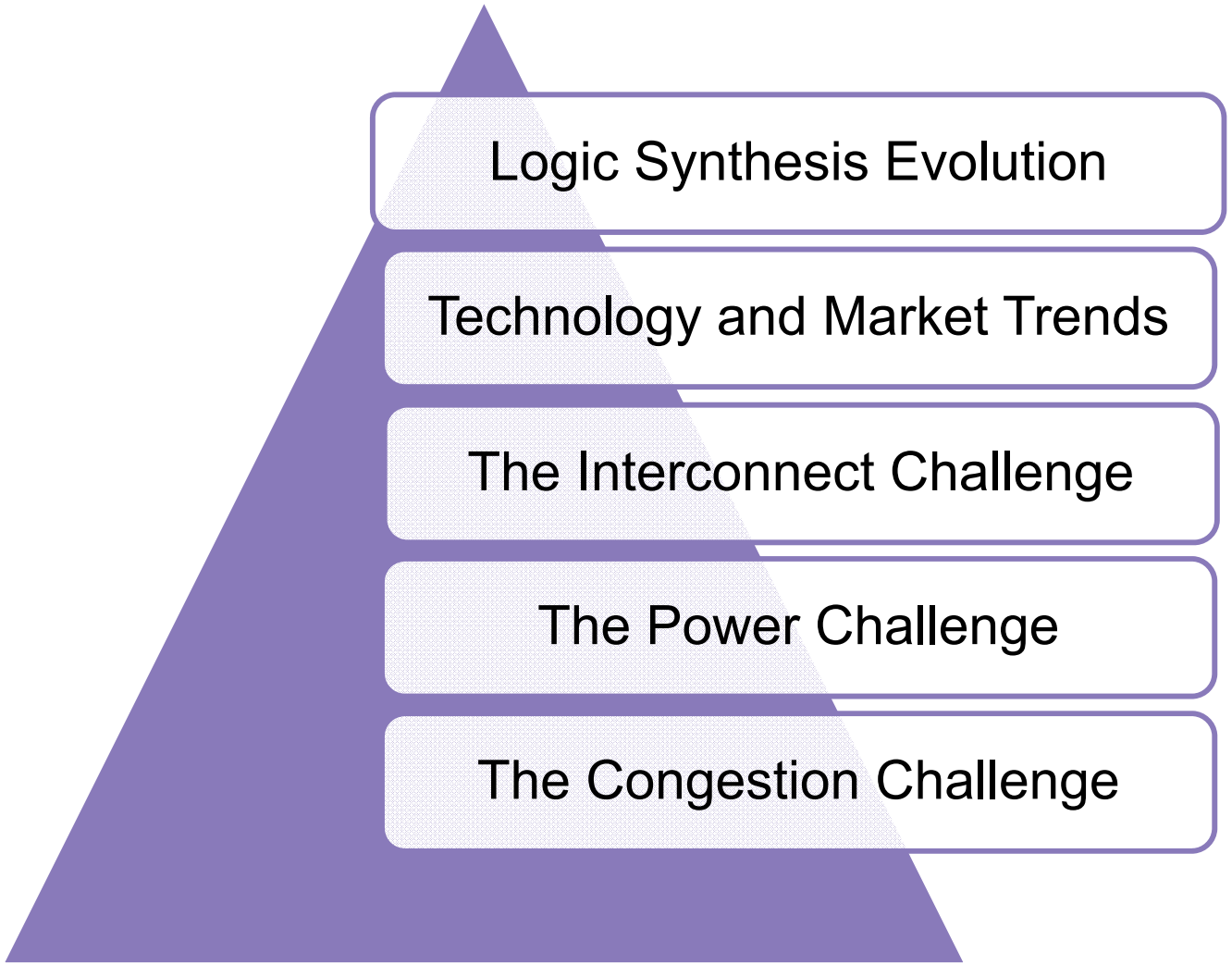
Abhijeet Chakraborty

Janet Olson

SYNOPSYS, INC

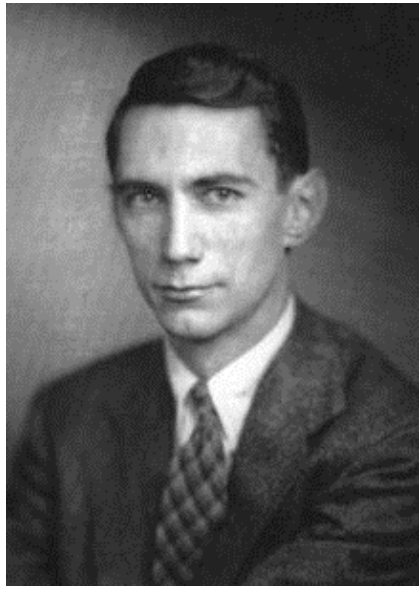
ISPD 2012

Outline



1949: Claude E. Shannon

"The Synthesis of Two-Terminal Switching Circuits,"
Bell System Technical Journal



x \ y	∧ 0	∧ 1
0	0	0
1	0	1

x \ y	∨ 0	∨ 1
0	0	1
1	1	1

x \ y	→ 0	→ 1
0	1	1
1	0	1

x \ y	⊕ 0	⊕ 1
0	0	1
1	1	0

Figure 1. Truth tables

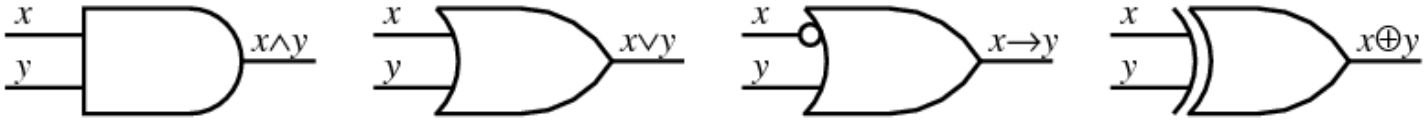


Figure 2. Logic gates

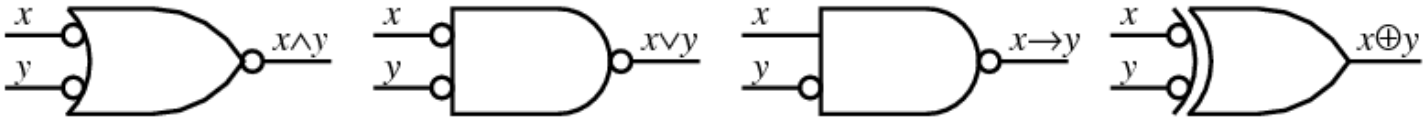
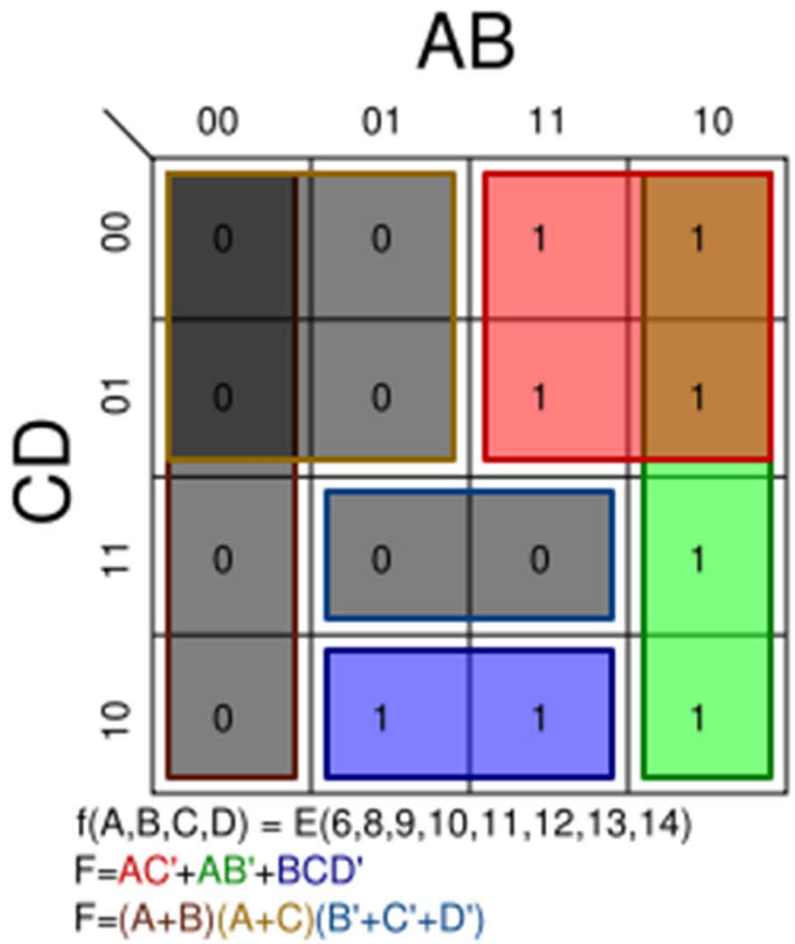


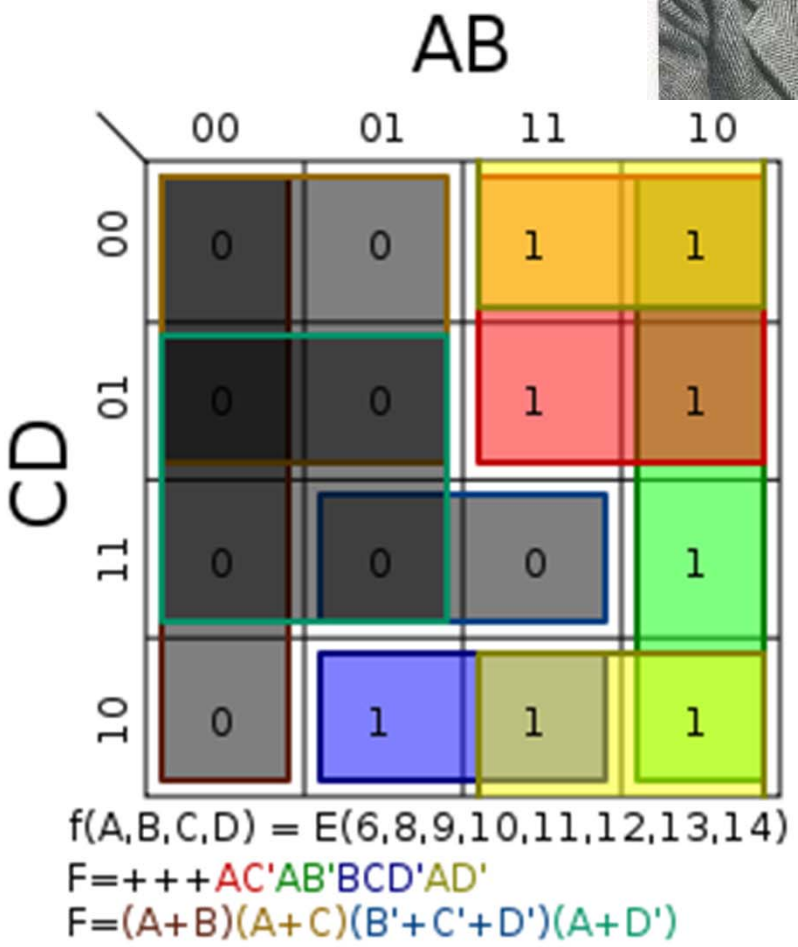
Figure 3. De Morgan equivalents

Figure 4. Venn diagrams

1953: Maurice Karnaugh

"The Map Method for Synthesis of Combinational Logic Circuits," Bell Labs

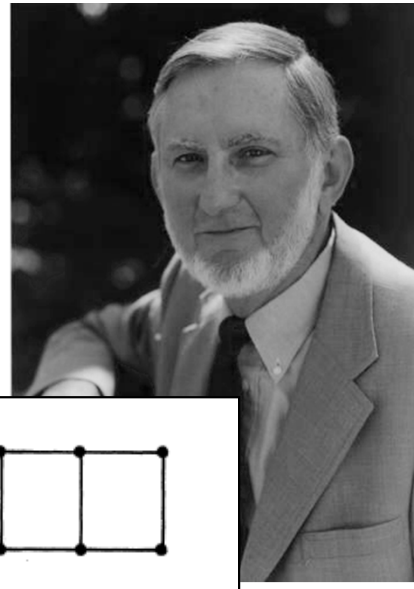




SYNOPSYS® 25 Years

1956: Edward J. McCluskey

"Algebraic Minimization and the Design of Two-Terminal Contact Networks," Ph.D. Thesis, MIT



Minimization of Boolean Functions using Prime Implicants

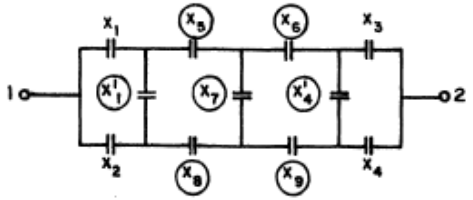
	x_4	x_3	x_2	x_1	
5	0	1	0	1	✓
6	0	1	1	0	✓
(ϕ) 9	1	0	0	1	✓
13	1	1	0	1	✓
(ϕ)14	1	1	1	0	✓

	x_4	x_3	x_2	x_1
5, 13	-	1	0	1
6, 14	-	1	1	0
9, 13	1	-	0	1

(a) Determination of Prime Implicants

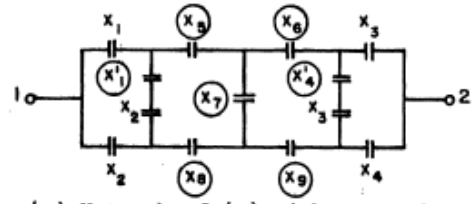
	5	6	13
*	x		x
*		x	
			x

(b) Prime Implicant Table

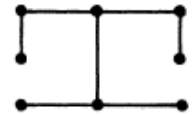


(a) A Contact Network to which a Path Having Transmission $x_9x_8x_7x_6x_5x_4'x_3'x_2'x_1'$ is to be Added

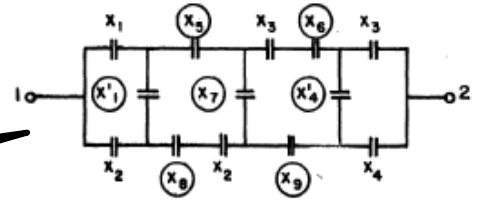
(b) Graph of Desired-Contact Subnetwork of (a)



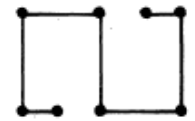
(c) Network of (a) with x_3 and x_2 Added to Break Desired-Contact Loops



(d) Graph of Desired-Contact Subnetwork of (c)



(e) Network of (a) with x_2 and x_3 Contacts Added so that Only One Desired-Contact Path Remains



(f) Graph of Desired-Contact Subnetwork of (e)

Design of Two-Terminal Networks

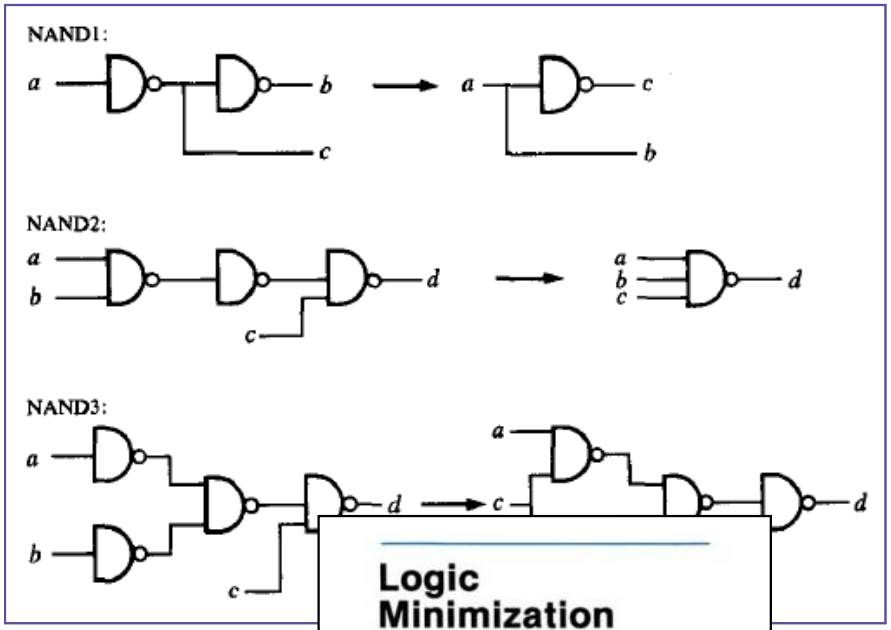
1981: Multi-Level Minimization

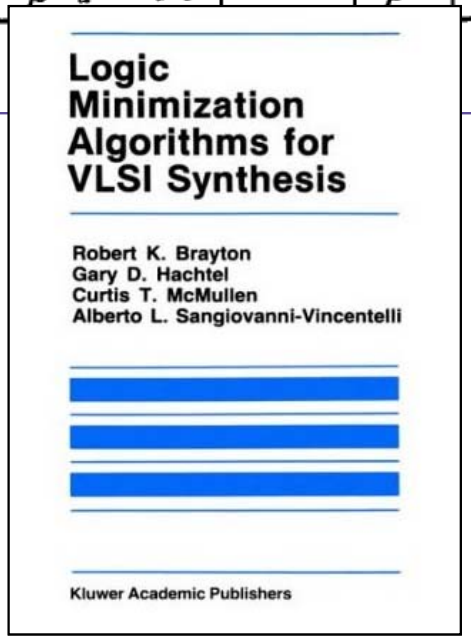
IBM, UCB, and University of Colorado at Boulder

***John Darringer, William H. Joyner, and Louise H. Trevilyan**, e.g. "Logic Synthesis Through Local Transformations," IBM, 1981*

***Robert K. Brayton, Gary D. Hachtel, A. Richard Newton, and Alberto L. Sangiovanni-Vincentelli**, e.g. "Logic Minimization Algorithms for VLSI Synthesis," UCB, 1984*

Espresso – Using heuristic and algorithms to reduce logic complexity

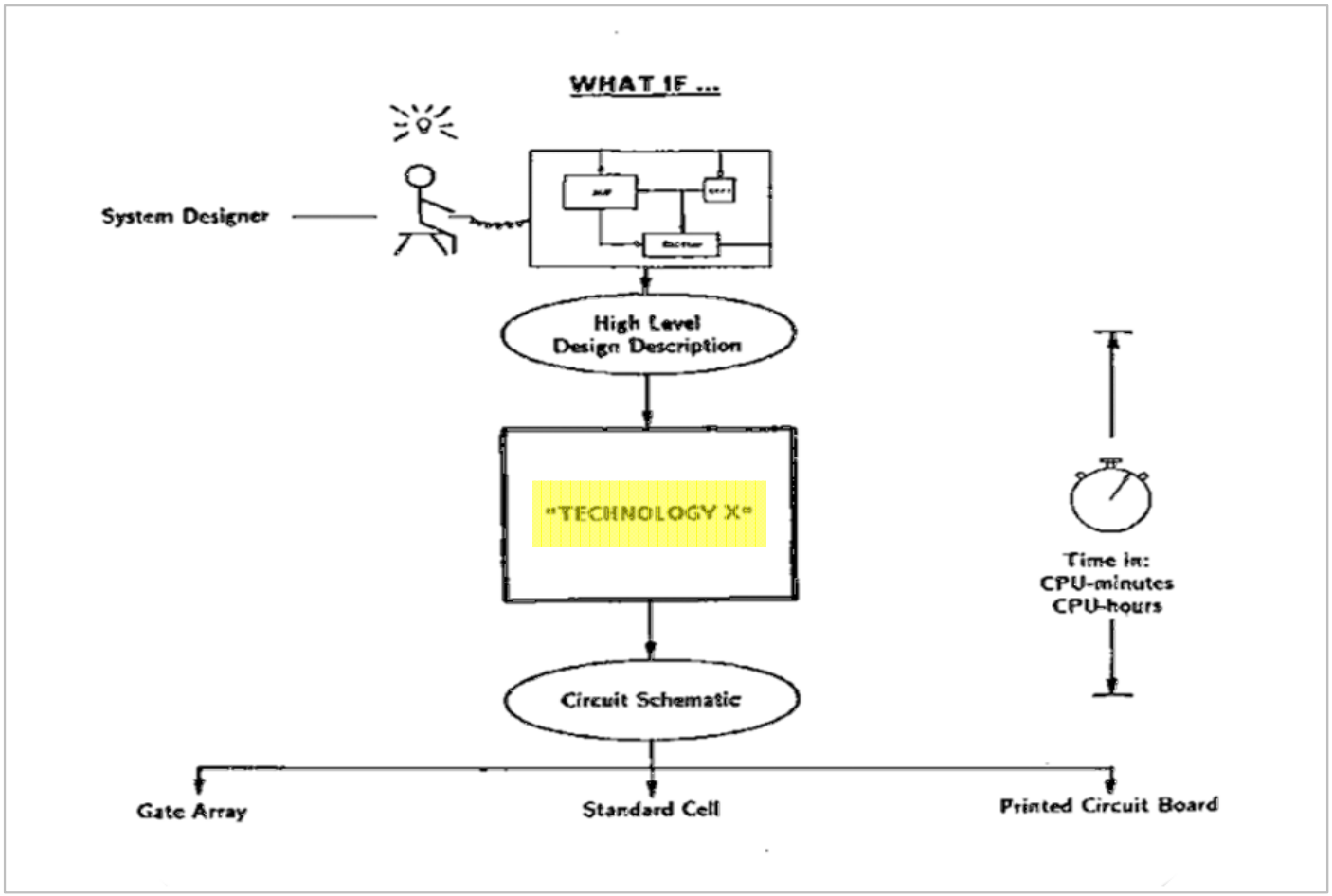




1986: Logic Compiler

Optimal Solutions, Inc. (aka Synopsys, Inc.)

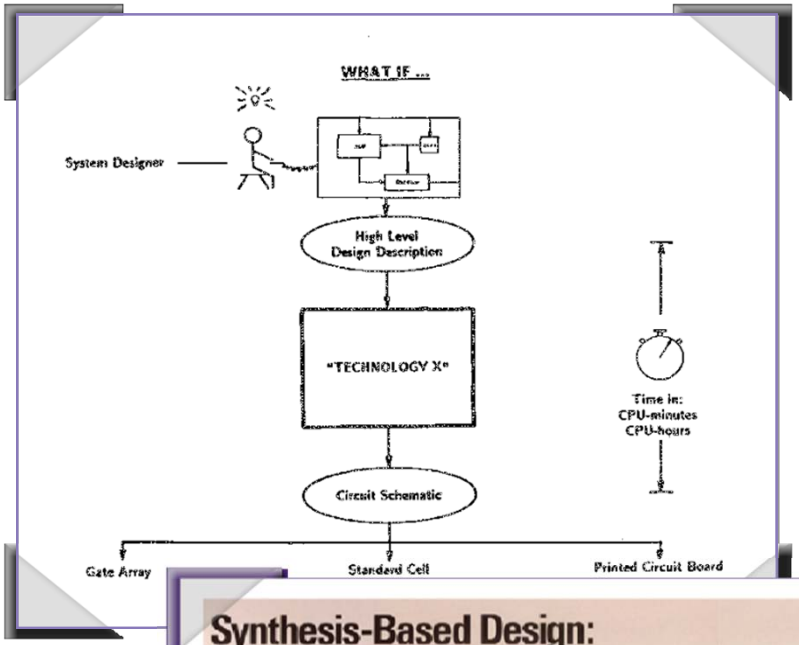
Technology X – Provide automation and increase productivity for gate level designers

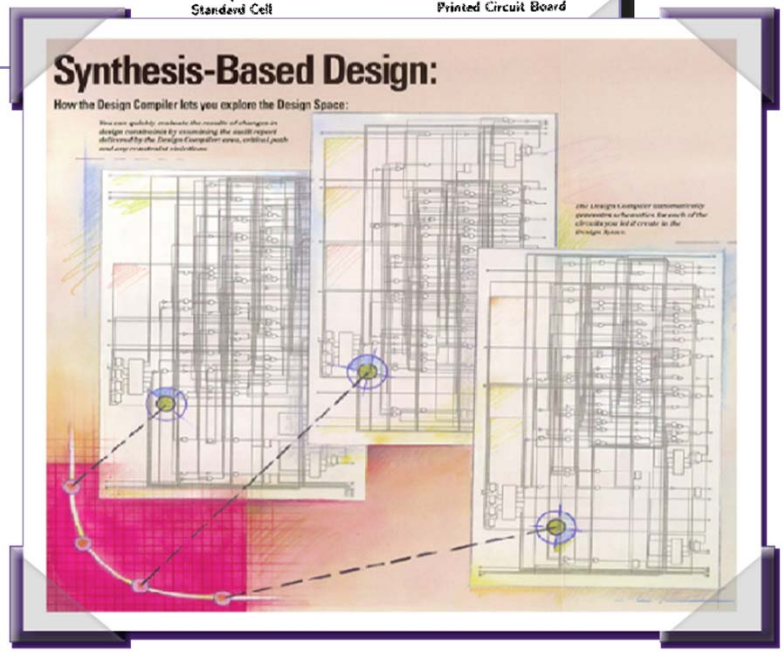


1988: ‘Technology X’=Design Compiler

Synopsys Founders Bill Krieger, Aart de Geus, and Dave Gregory

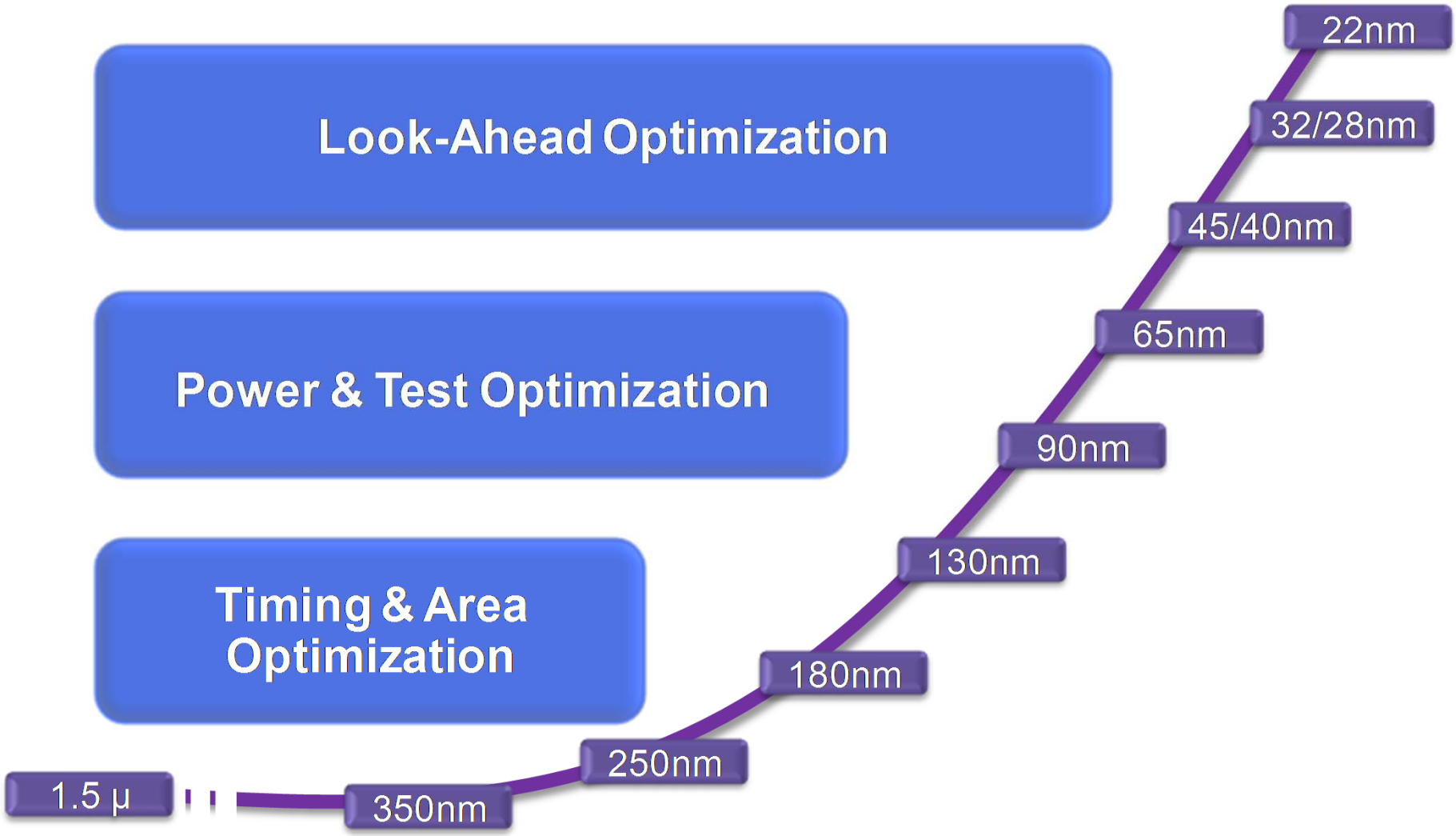
- Launched the era of HDL-based design
- Massive productivity gains
- From days to hours and minutes



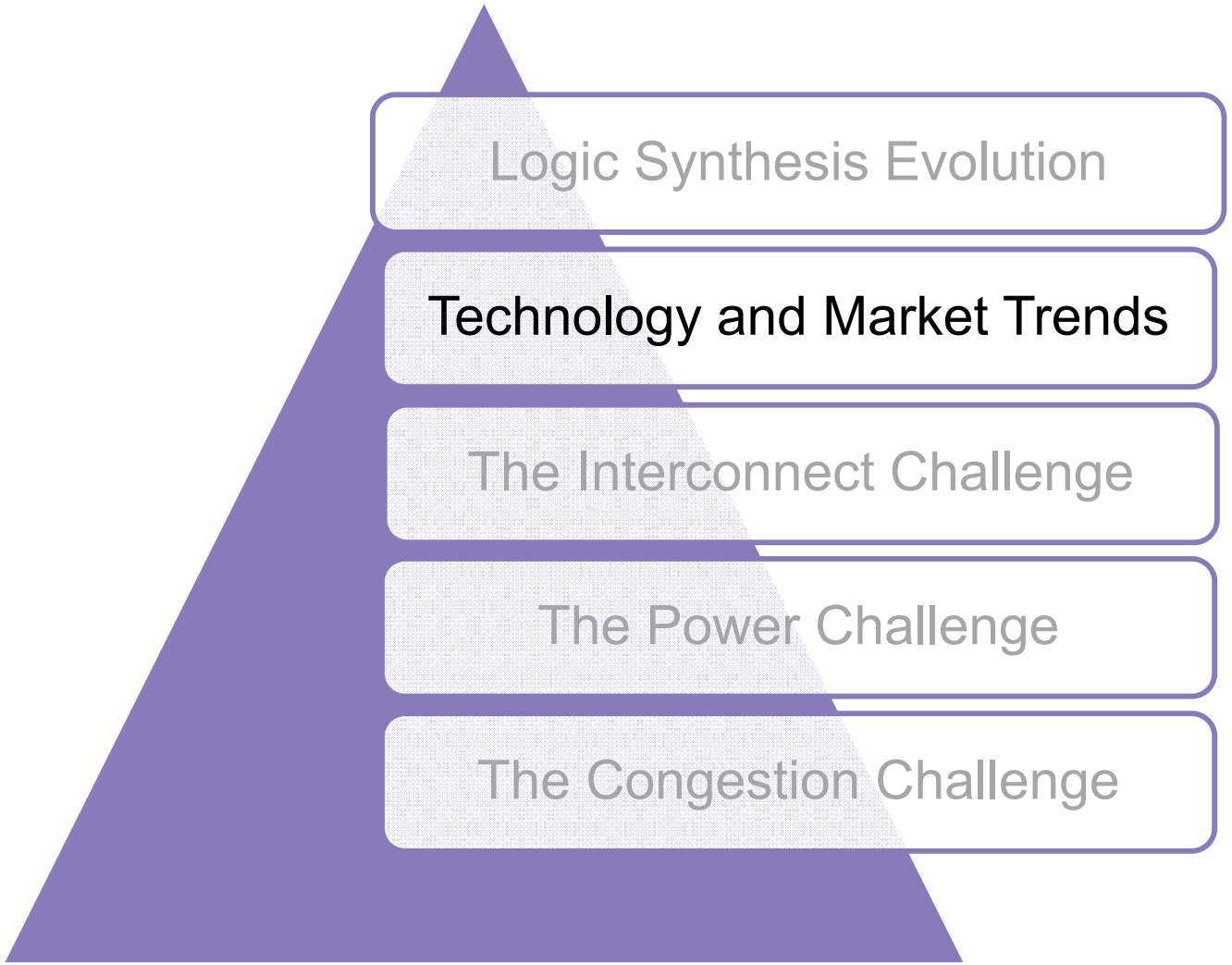


SYNOPSYS® 25 years

Design Compiler: Keeping Up With the Technology Curve in EDA



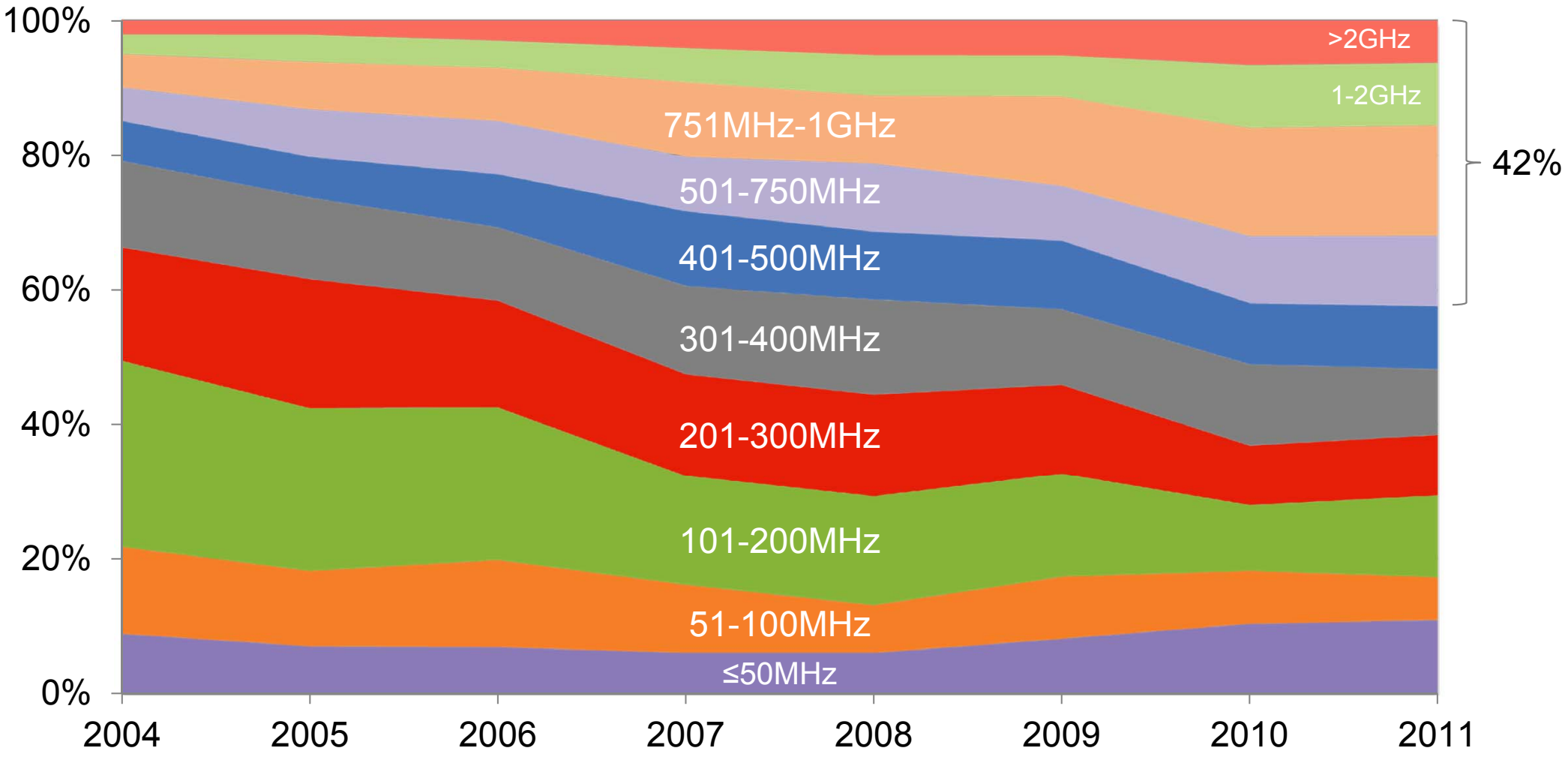
Outline



Market Challenge: Mobile Devices

- A single device that does it all!
  - Cell Phone
  - Digital Media
  - Full Internet
  - GPS
  - Computer
  - Interactive Gaming
  - HDTV with DVR
- All powered by a 900mAh battery
- Market Drivers (ITRS)
  - Size/Weight Ratio
  - Battery Life
  - Function: 2X Every 2 Years
  - Time to Market: ASAP

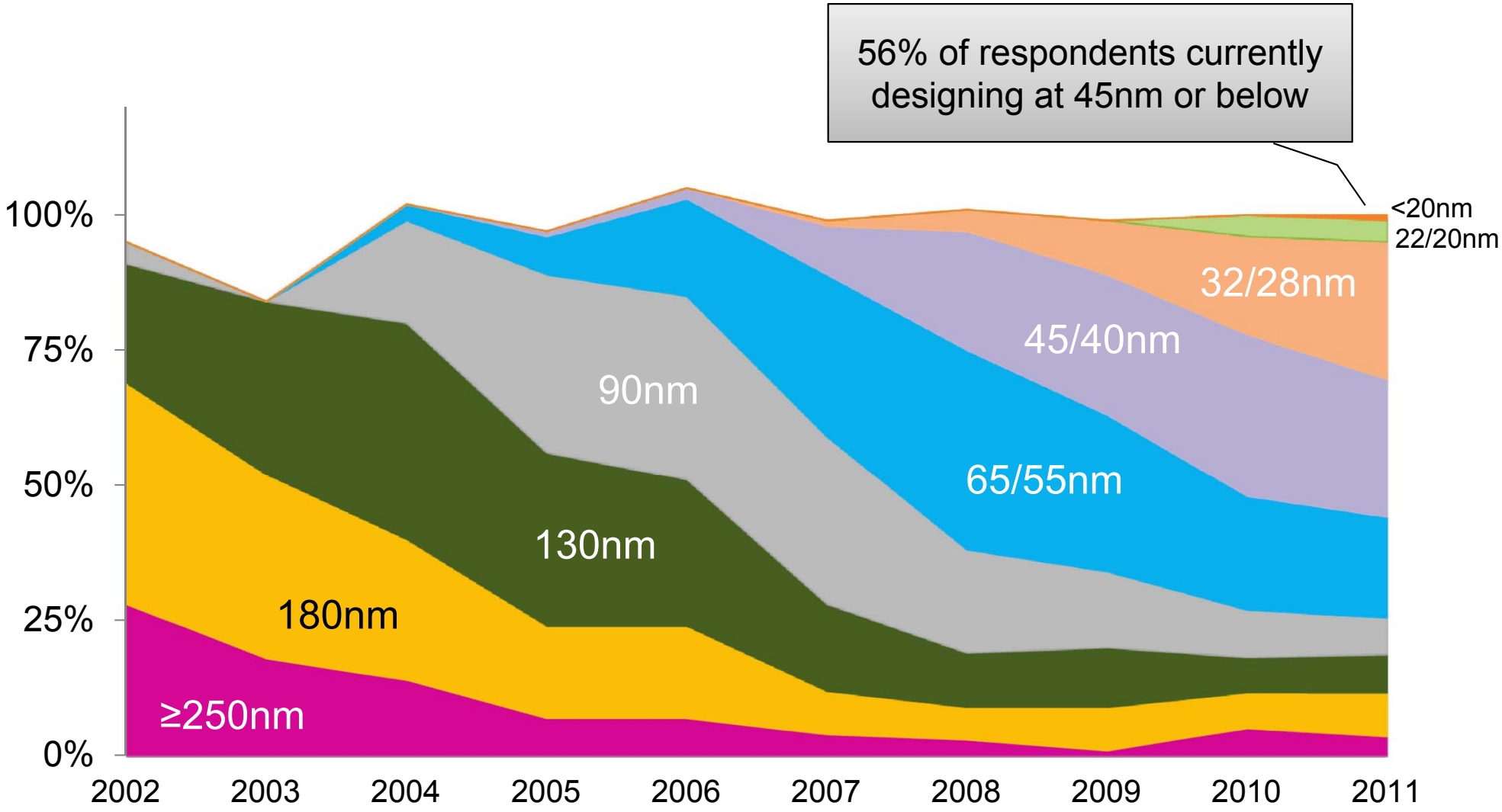
Market Trends: Clock Frequency



N = 962

Source: 2011 Synopsys Global User Survey

Market Trends: Advanced Design

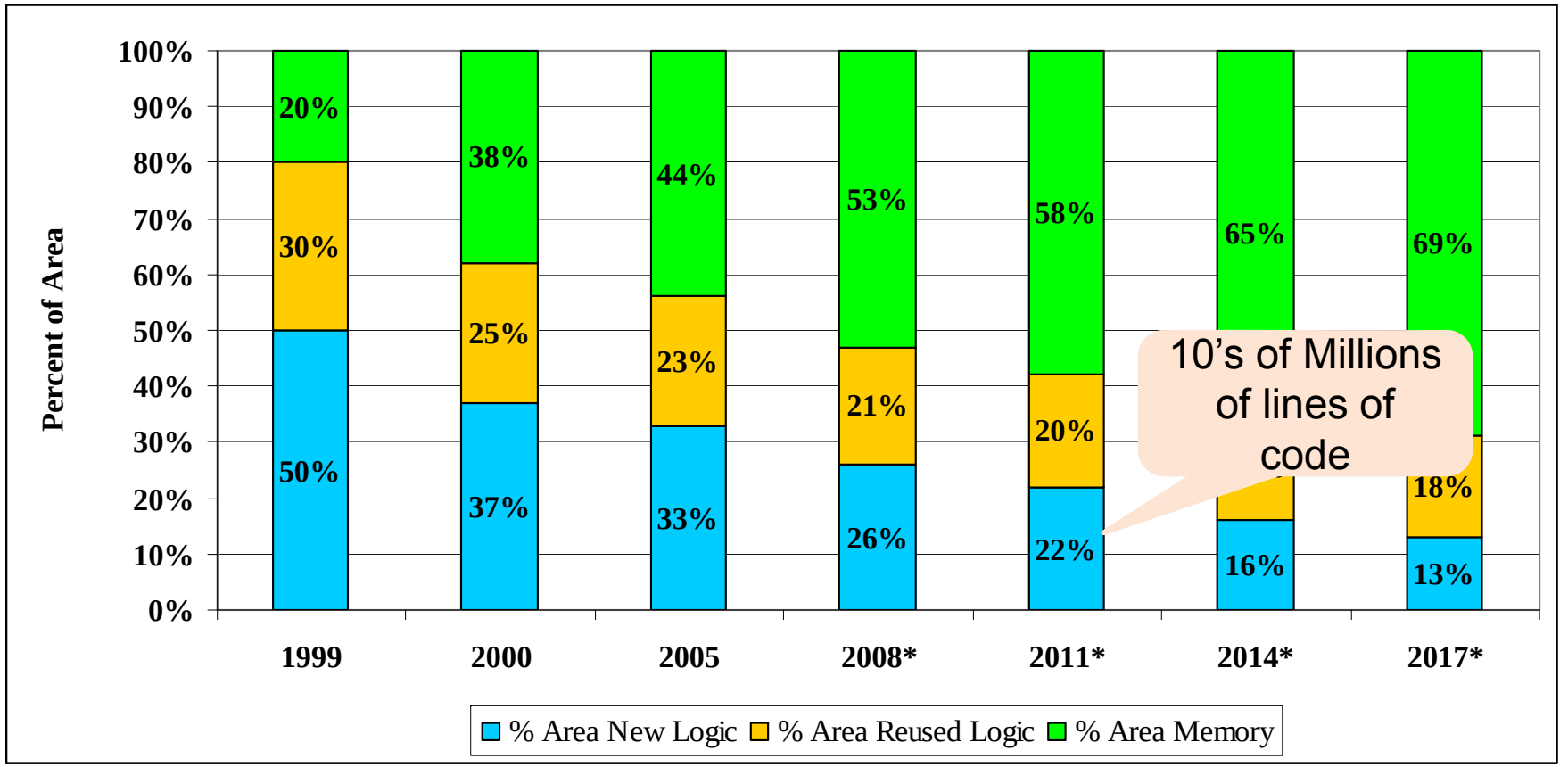


N = 1277

Source: 2011 Synopsys Global User Survey

Looking Into The Next Decade

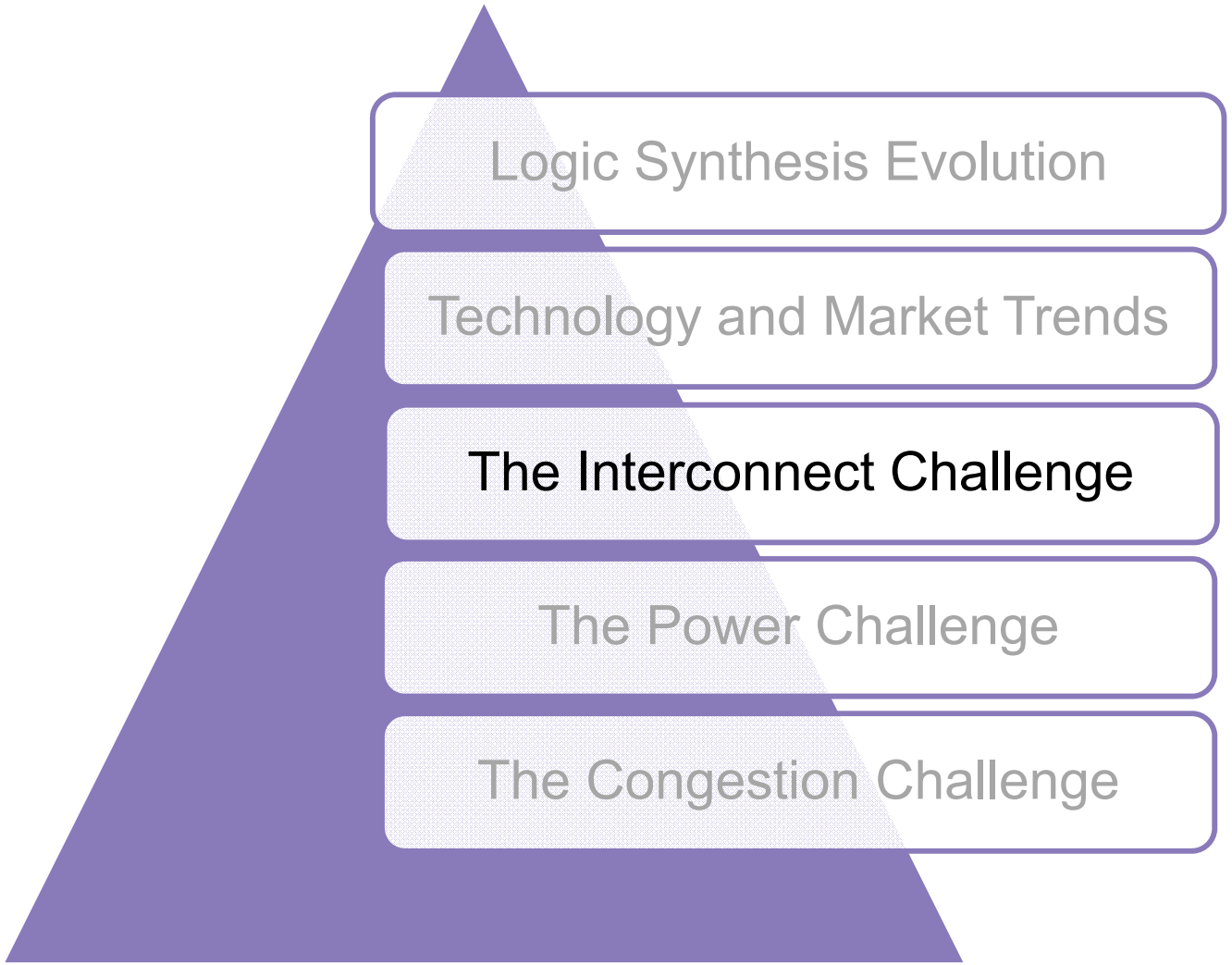
Making the Transition to High-Level Design… Again!



New logic allows design of chips that are differentiated in Silicon

Source: Semico Research Corp.

Outline



The Interconnect Challenge

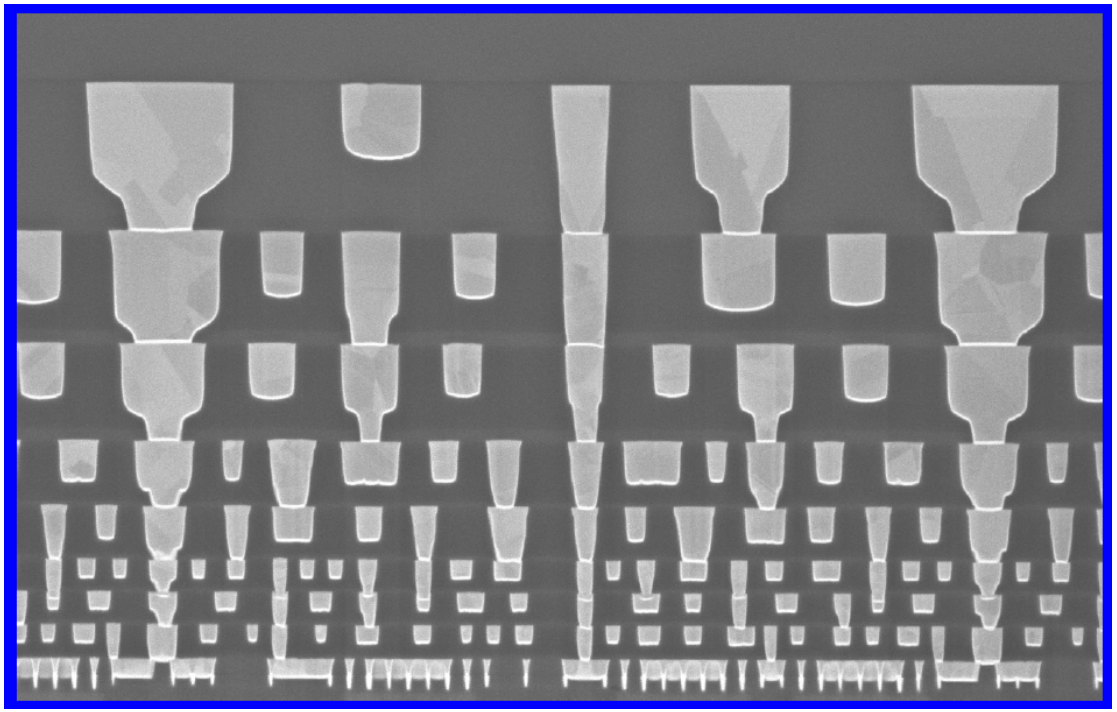
	2005	2010	2012
Process	130 to 90nm	65nm, 45nm, <32nm	28nm, 20nm, 14nm
Wire length (m/cm^2)	1019	2222	3143
Important new Effects	Route Topology	Layer Awareness Coupling Capacitance	Resistive shielding Much less resistance on higher metal layers

- Impact of interconnect has to be considered in Logic Synthesis
  - Helps faster convergence – tight correlation with the backend
  - Produces more efficient designs – lower area, power
  - Design flow becomes more predictable
  - Improves performance – higher frequency

Metal Layers Can Affect Timing

Pre-Route Vs. Post-Route Correlation

- In some cases, pre-route estimates can be overly pessimistic/optimistic
  - Increased RC variation at advanced nodes
- Layer-awareness can provide more accuracy
  - All metal layers do not have the same RC values
  - Account for detoured nets due to congestion or obstruction

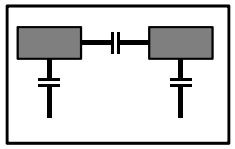
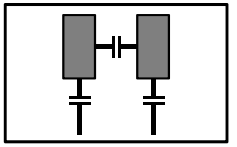
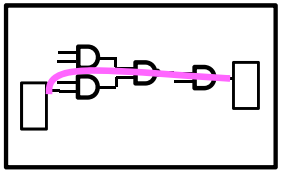
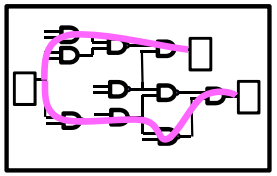
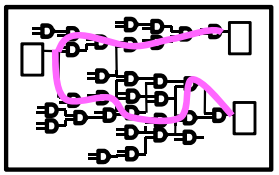


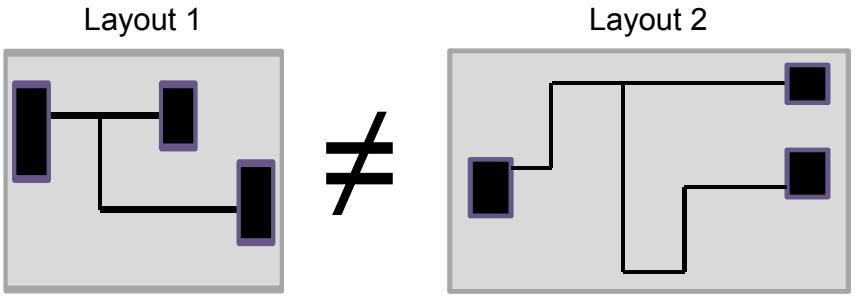
✓Metal 1-3 pitches match transistor pitch
✓Graduated upper level pitches optimize density and performance

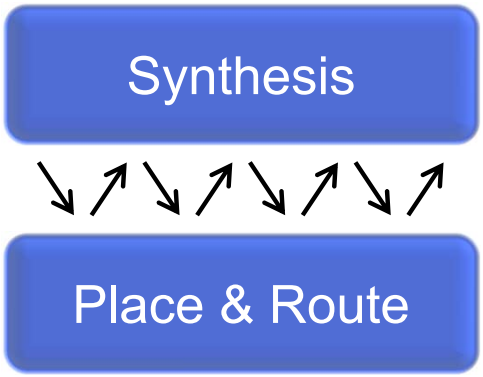
Source: "Intel: A 32nm Logic Technology Featuring 2nd-Generation High-k + Metal-Gate Transistors, Enhanced Channel Strain and 0.171um2 SRAM Cell Size in a 291Mb Array," S. Natarajan, et al., IEDM, 2008

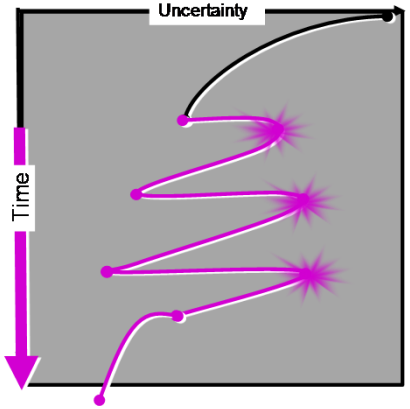
Brief History of Interconnect Modeling

Physically-Aware Timing Correlation

	1986	2005	2010
Process	>1 um to 90nm	130 to 90nm	65nm, 45nm, <32nm
Modeling	WLM	Physical Aware Net Models	Consider more effects – Coupling Cap Density
Coupling Capacitance	Fanout-based Capacitance	 Wider wires More spacing	 Taller wires Less spacing
Density			



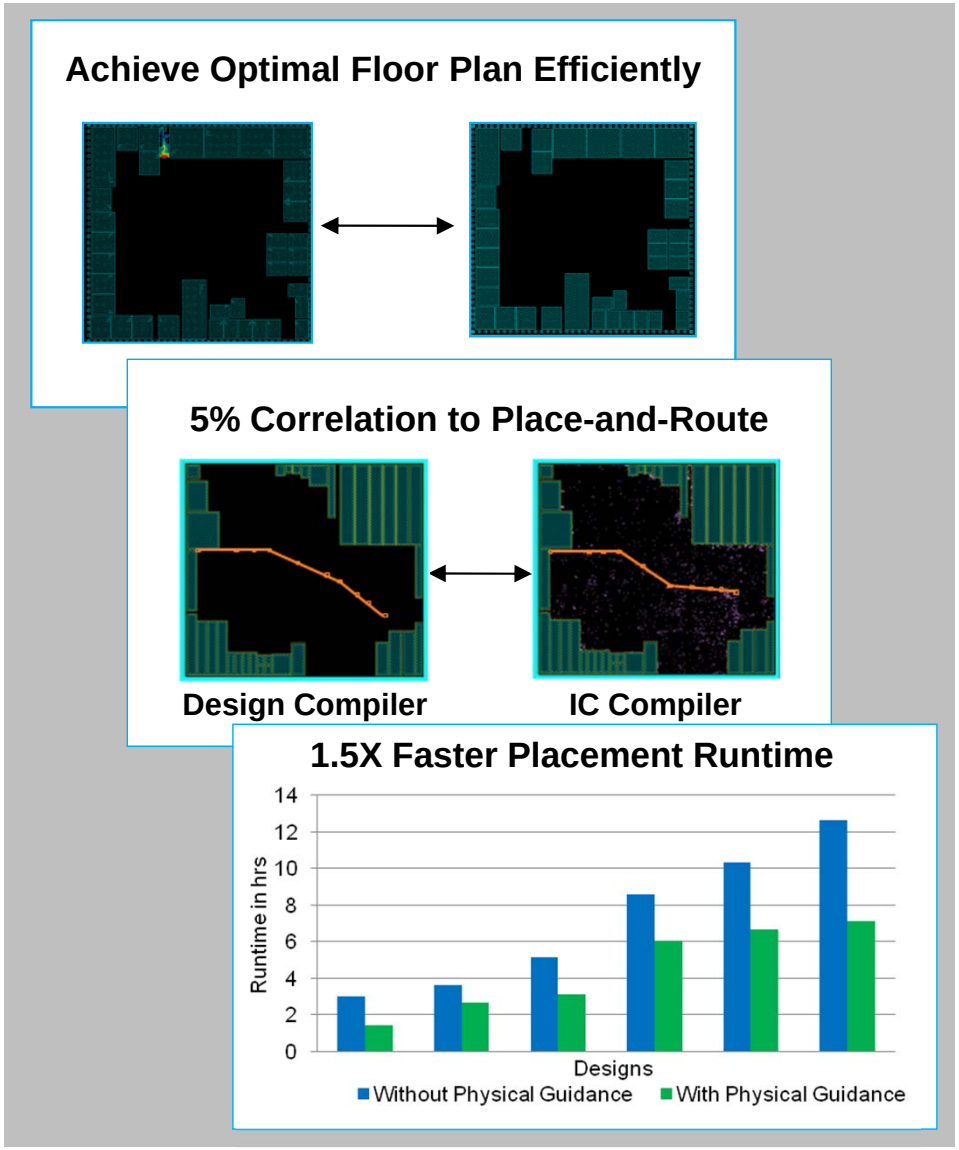




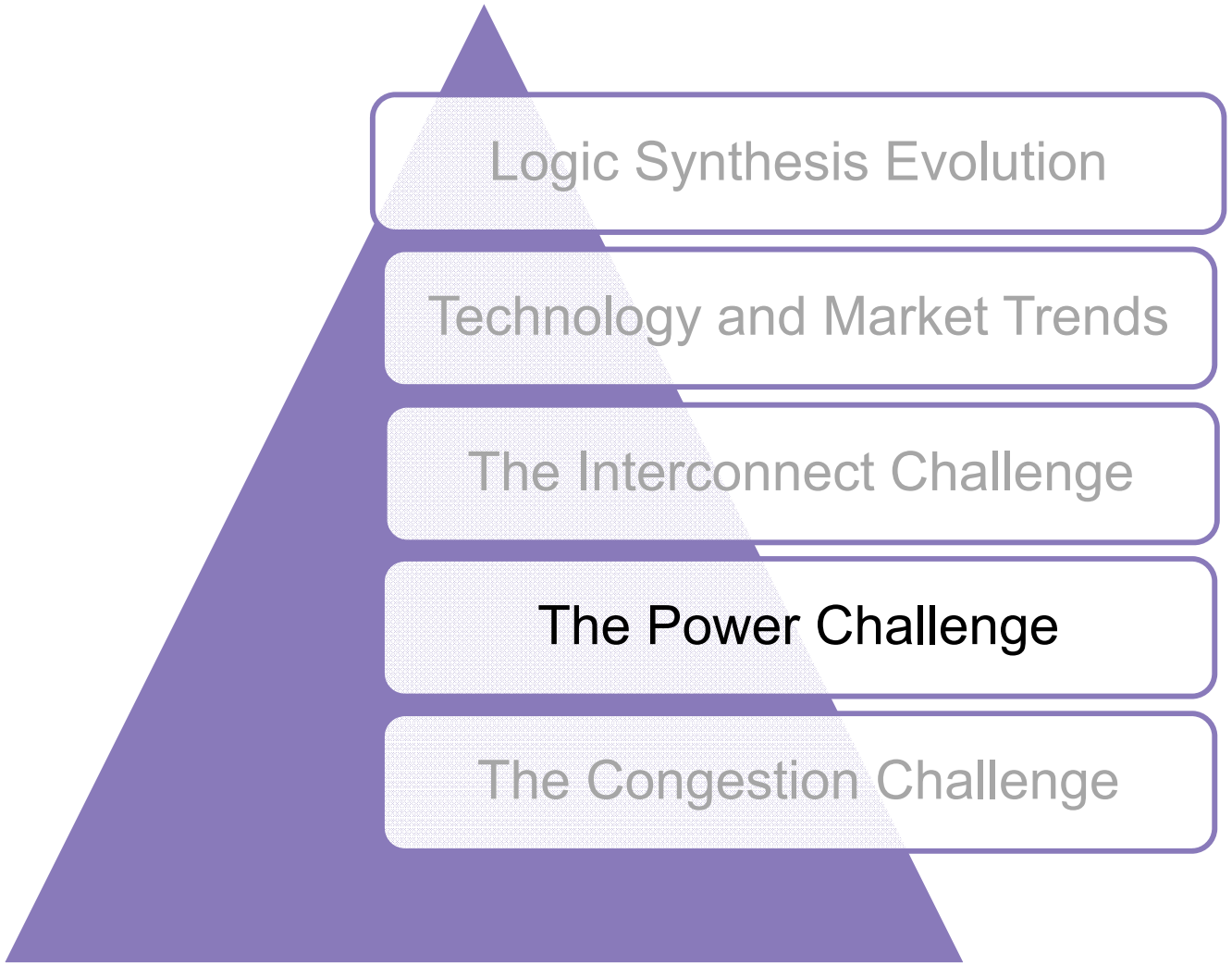
SYNOPSYS 25 years

Solving the Interconnect Challenge

- Modeling of resistance and capacitance with good fidelity
- Look-ahead optimizations
- Floorplan exploration
- Physical guidance to Place and Route tool
  - Faster convergence
  - Tight timing correlation to layout



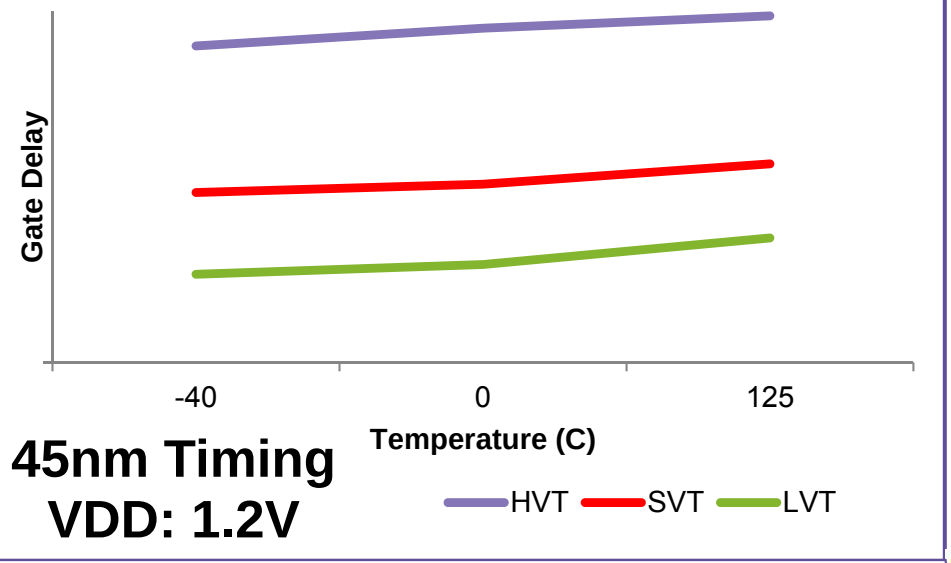
Outline



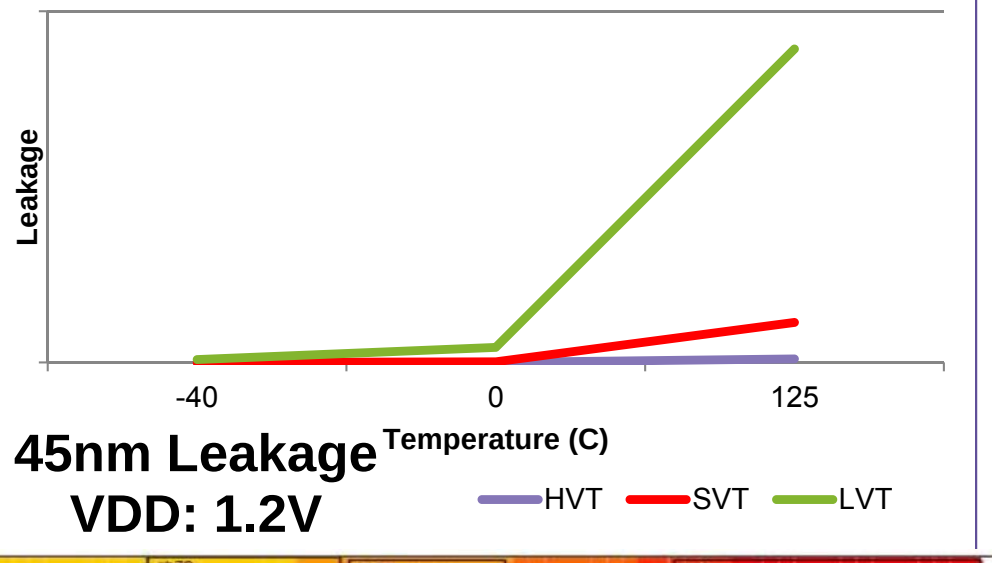
The Static Power Challenge

- We have reaped the rewards of high-K and metal gates – static power is still increasing
- Static power increases 3X per technology node AND ~ 2X every 10°C
  - Higher static power increases temperature, AND higher temperature increases static power
- Time-variable temperature gradients across the IC surface
  - Increasingly complex power management schemes exacerbate this problem
- Temperature Inversion

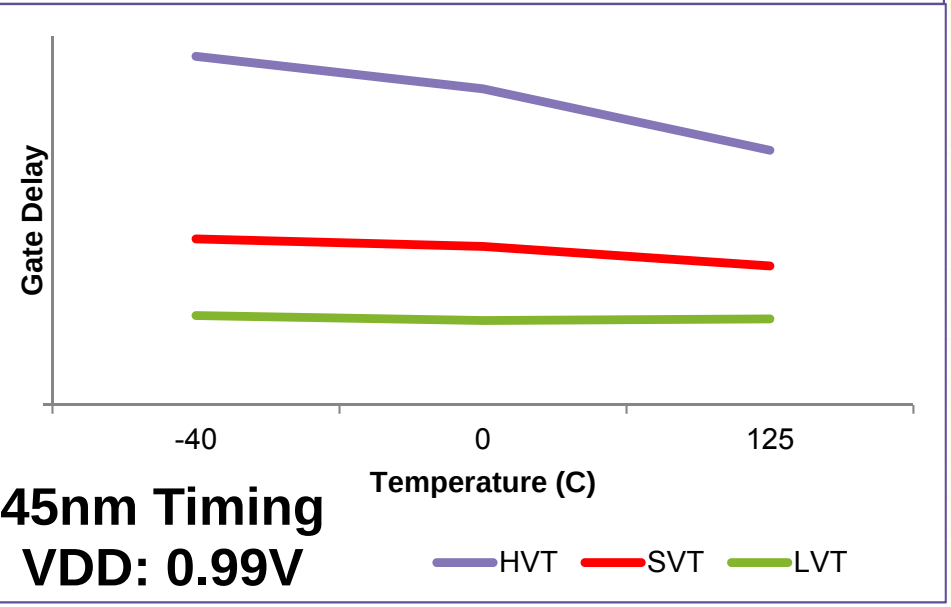
Synthesis Needs to Optimize Across Many PVT Corners, Concurrently



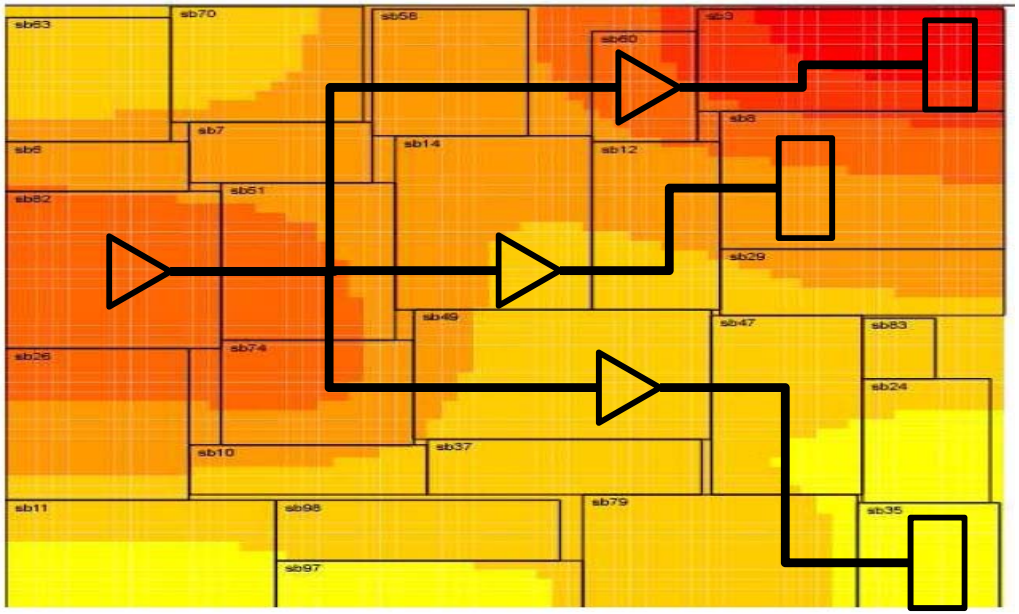
45nm Timing
VDD: 1.2V



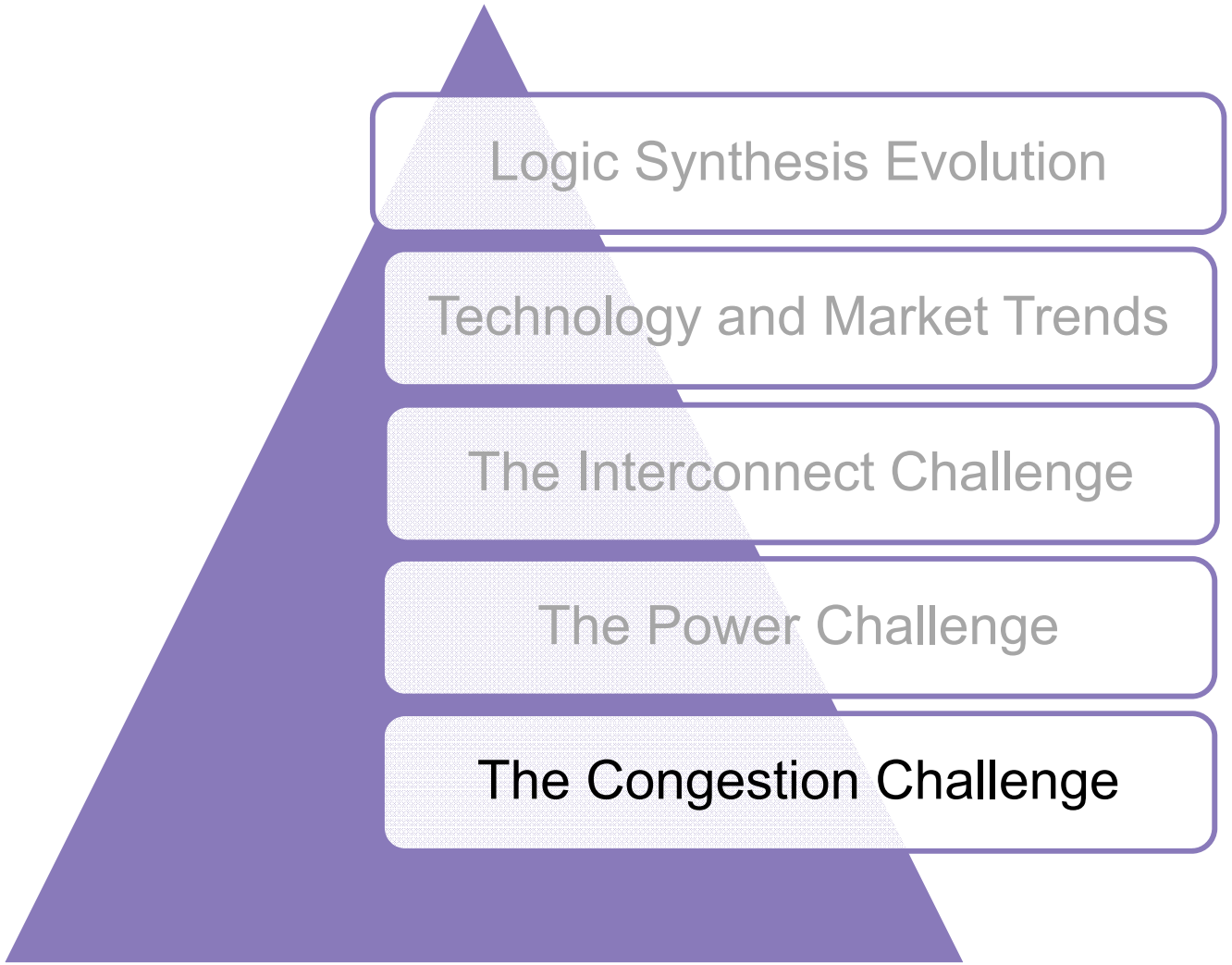
45nm Leakage
VDD: 1.2V



45nm Timing
VDD: 0.99V



Outline



The Congestion Challenge

- Until recently, improving congestion could only be addressed in the back-end, during **placement** and **routing**.
- Congestion was seen as an exclusively back-end problem.
- The placer solution is to spread out the cells in the congested areas.
- The router solution is to do many search-and-repair loops.

The Congestion Challenge (Cont...)

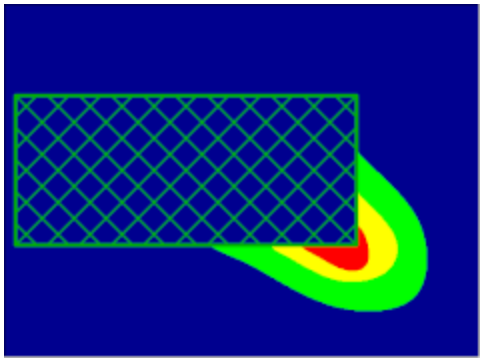
- Solving congestion with placement doesn't always work.
  - Sometimes, congestion persists even after spreading the cells.
  - Even when it does work, it comes at a cost to delay and area.

- Solving congestion in routing can also be problematic
  - Many search-and-repair loops may be needed.
  - It takes a lot of CPU time (and calendar time!).
  - It's risky.

Types of Congestion

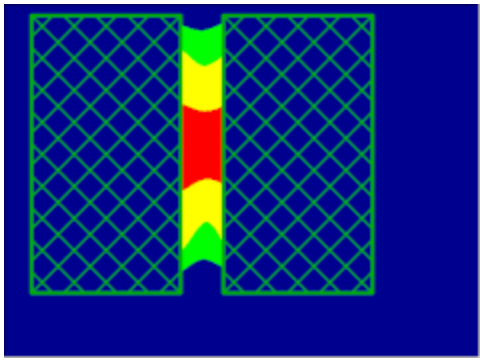
Floorplan congestion

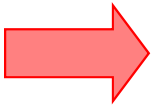
- Caused my macro or port placement.
- Can probably only be effectively addressed by changing the floorplan.

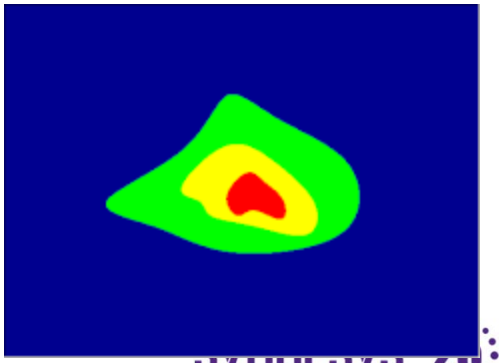


Cell congestion

- Shows up as a blob of congestion out in the open.
- Probably due to the nature of the netlist.







Logic Synthesis and Congestion

- Routing congestion is often caused by the **topology** of the netlist.
- The netlist topology is the result of many arbitrary decisions made during logic synthesis.
- During logic synthesis, we have a great deal of control over the netlist topology.
- By making some wise decisions during logic synthesis, we can reduce congestion considerably.

27

Congestion: What can be done?

- *In general, once the logic is mapped and placed, it is too late.*

- The topology of the network
  - Many nets (graph edges)
  - Highly connected (a many-to-many mapping)

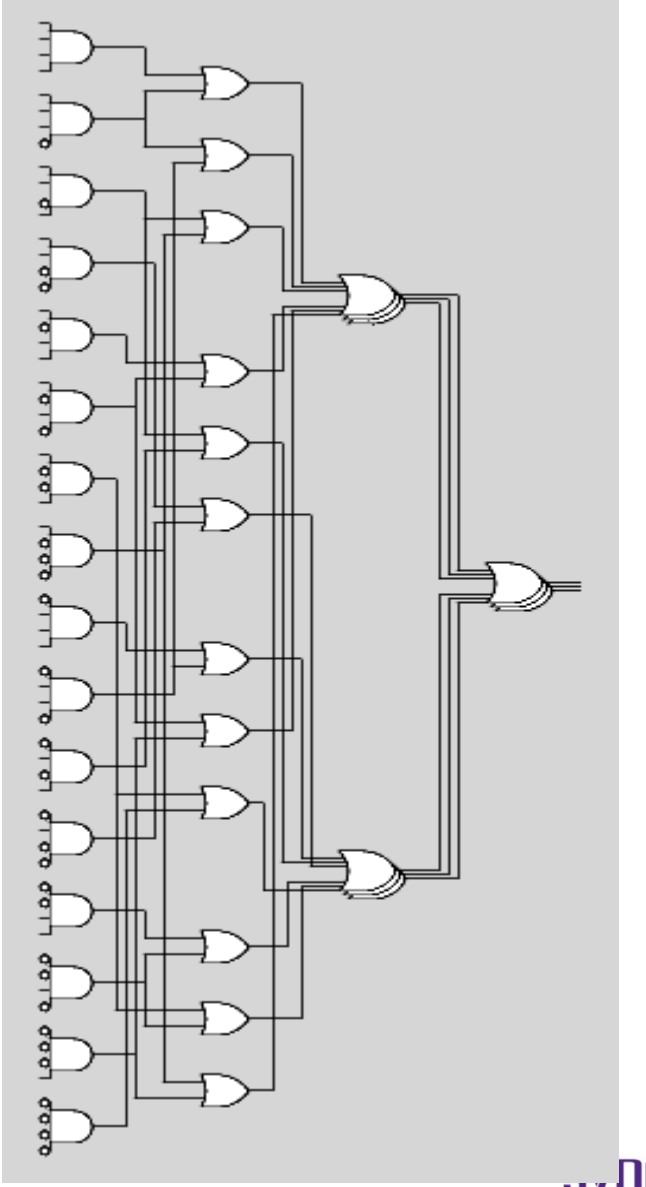
- Examples:
  - Large sums of products
  - Parallel high-fanout nets
  - Aggressive or widespread logic sharing
  - Scan logic

28

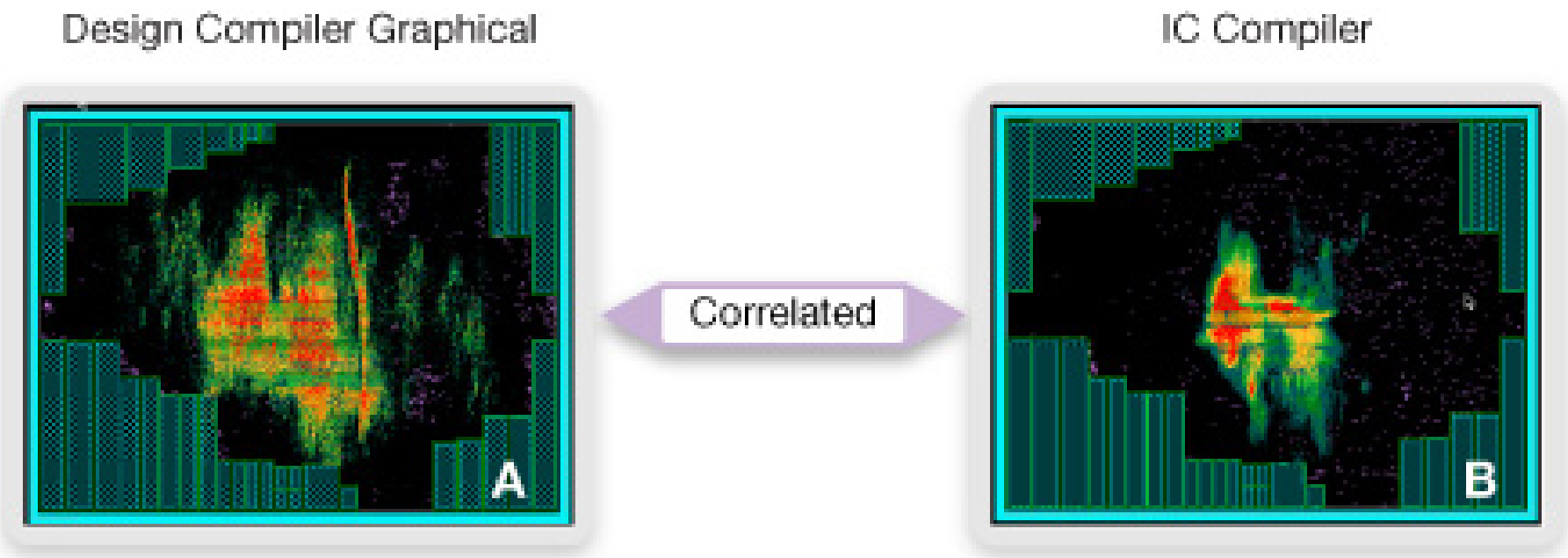
Large Sums of Products

- Large sums of products can create congestion if the OR trees combining the minterms (or cubes) differ in structure.

- When this happens, no placement of the cells can solve the congestion.

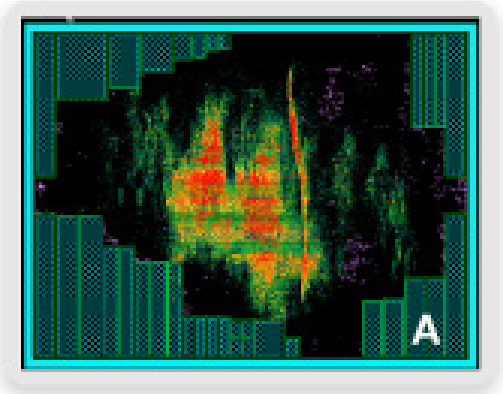


Congestion: Prediction



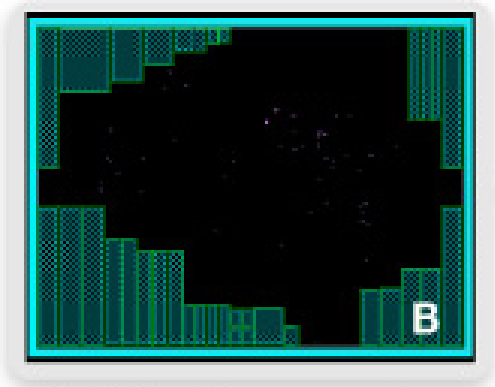
Congestion: Optimization

Design Compiler Graphical



Congestion prediction

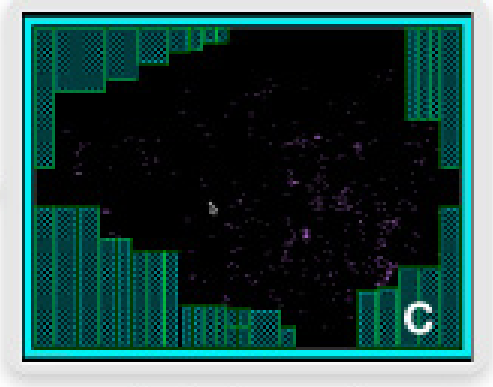
Design Compiler Graphical



Optimized for congestion

Correlated

IC Compiler



Post placement

Looking Into the Next Decade

There is a Great Deal of New Technology Ahead!