Is SPICE Good Enough for Tomorrow's Analog?

Laurence Nagel

Omega Enterprises Consulting
Kensington, CA 94708

Colin McAndrew

Freescale Semiconductor
Tempe, AZ 85284

IEEE BCTM Paper 6.4
October 5, 2010

Overview

- SPICE was first released 40 years ago (next year) !!!
- Virtually every EE student has to learn SPICE to learn how to design integrated circuits (and to graduate)
- SPICE is still around because it has evolved to remain a vital and useful tool in the design process
- What has driven SPICE evolution in the last 40 years?
- Is SPICE Good Enough for Tomorrow's Analog?

Is SPICE Dead?

SPICE is the analog designer's version of Old Faithful — with the emphasis on “old.” Originally designed for simple designs with a handful of transistors, SPICE can't keep up with the demands of today's many-thousand-transistor designs. Companies are going to have to overcome their fears and shift to a top-down design approach if they want to remain competitive. Just working harder isn’t enough anymore.

Ken Kundert, “Why SPICE Won't Cut It For Analog Anymore,” *Computer Design,* April 1999.

IC Technology Changes in the Last 40 Years

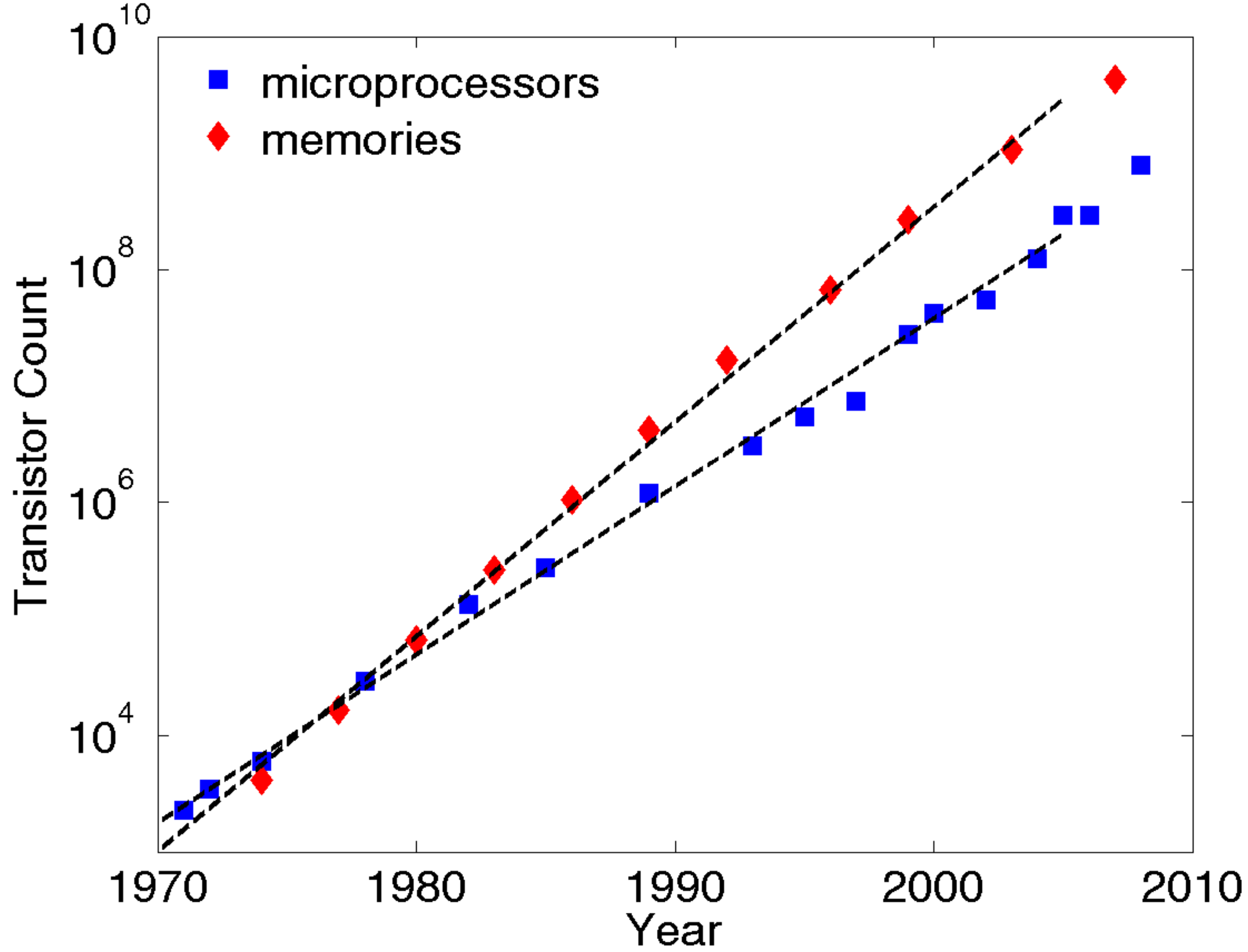
Design Rules in mils (1 mil = 25.4 micron)	Design rules in nanometers (1 micron = 1000 nm)
Masks from rubylith and cameras	Masks from GDS II and e-beam
Chips with a few transistors	Chips with a few billion transistors
Wafer size of one inch	Wafer size of twelve inches
Packages with a dozen pins	Packages with hundreds of pins

Moore's Law

"The complexity for minimum component costs has increased at a rate of roughly a factor of two per year ... Certainly over the short term this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. That means by 1975, the number of components per integrated circuit for minimum cost will be 65,000. I believe that such a large circuit can be built on a single wafer."

Gordon Moore, *Electronics Magazine*, 1965

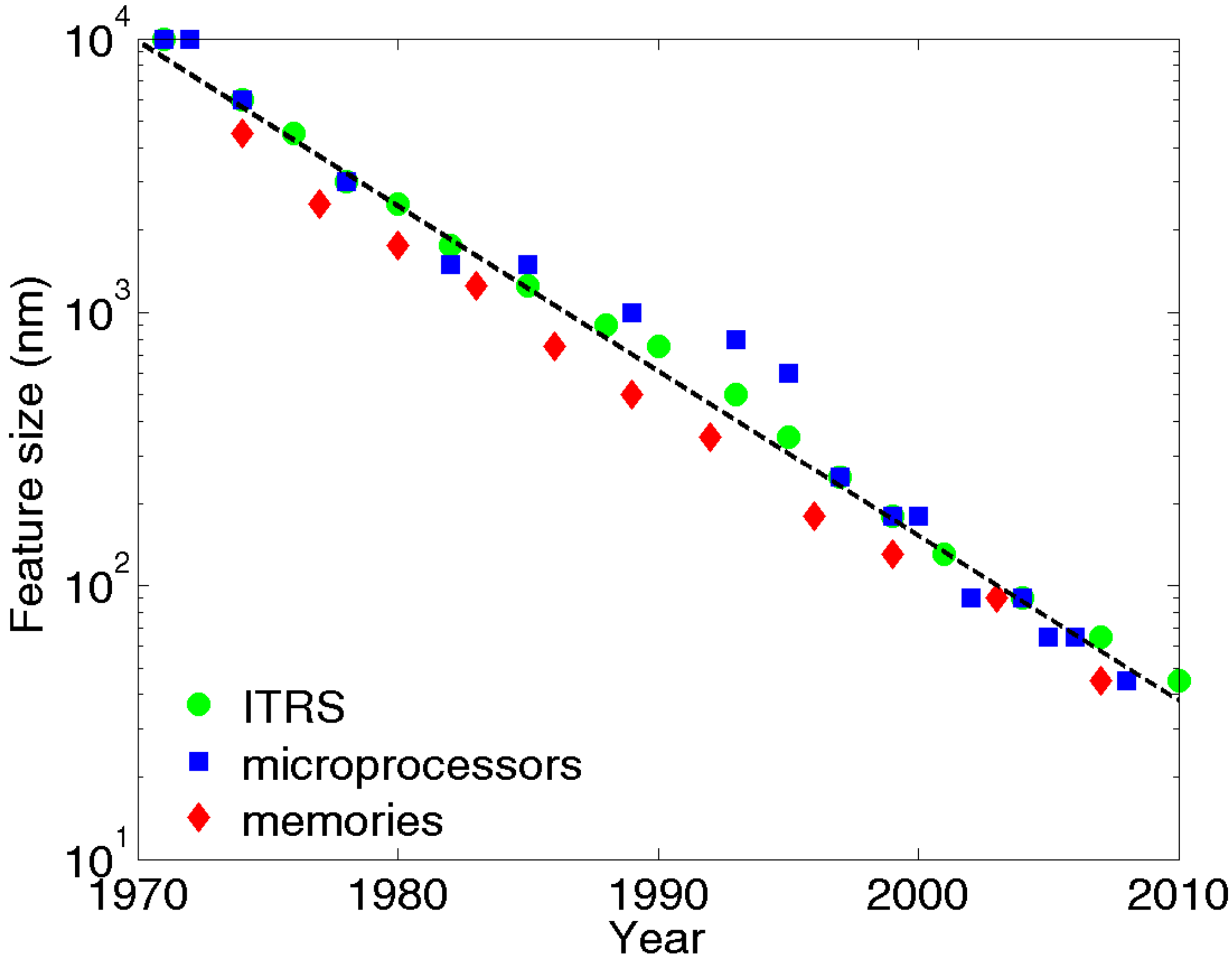
Transistor Count



Moore's Law

- Design rules decrease by a factor of $\sqrt{2}$ every two years (1,000X in 40 years)
- Because of shrinking design rules, the transistor density increases by a factor of 2 every two years (1,000,000X in 40 years)
- Therefore, the per function cost of electronics decreases by a factor of 2 every two years (assuming that the cost per cm^2 doesn't change)

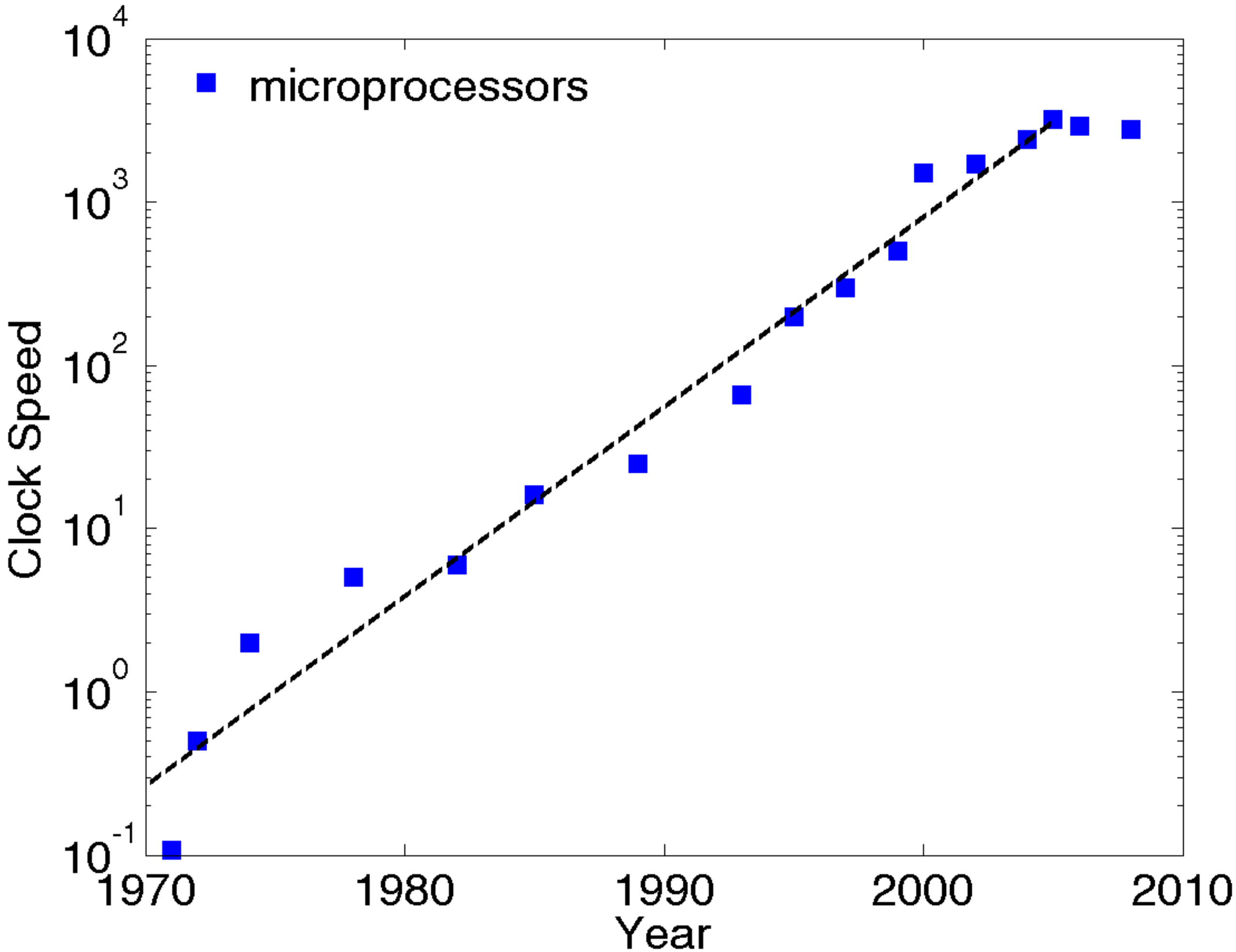
Minimum Feature Size



Corollaries to Moore's Law

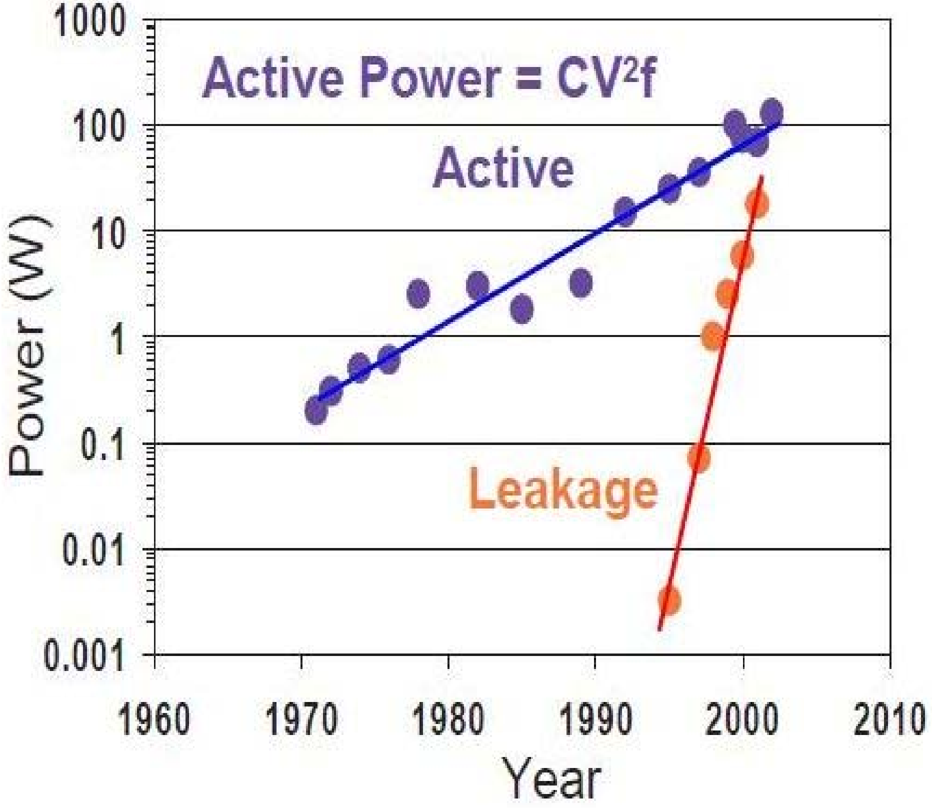
- The switching speed of a transistor decreases by a factor of $\sqrt{2}$ every two years. The maximum frequency of operation increases by the same factor.
- Because device capacitance and parasitic capacitance decreases by a factor of $\sqrt{2}$ every two years, and the clock rate (can) increase by a factor of $\sqrt{2}$ every two years, the per function power ($CV^2 f$) was supposed to stay constant. Unfortunately, this neglected leakage current!

Clock Speed



microprocessors
Clock Speed
10^4
10^3
10^2
10^1
10^0
10^{-1}
1970
1980
1990
2000
2010
Year

Leakage Current



Paul Packan (Intel), CICC 2008

Pre-SPICE Milestones

- 1947 - Point Contact Transistor Invented
- 1959 - Planar Integrated Circuit Process Invented
- 1960 - MOS Transistor Invented
- 1963 - Complementary MOS Invented
- 1966 - ECAP Simulation Program Published
- 1971 - SPICE First Released

SPICE (Simulation Program with Integrated Circuit Emphasis)

- DC operating point analysis, small-signal AC analysis and transient analysis in one package
- Built-in models for diodes and bipolar transistors
- Modified Newton-Raphson iteration with heuristics that worked well with bipolar circuits
- Implicit integration techniques reduced problems with the widely spread time constants of an IC
- Utilized sparse matrix techniques, so it could run circuits with hundreds of nodes

Computing in the '70s

- The UC Berkeley computer in the '70s was a CDC 6400
- The input to the computer was punched cards
- The output of the computer was a line printer
- The MIPS rate was comparable to an Intel 286 (1 MIPS)
- The maximum available memory was 100,000 octal 60 bit words daytime and 140,000 octal at night (comparable to 256 KByte)

The SPICE Era

- 1973 - SPICE1
- 1975 - SPICE2
- 1981 - HSPICE
- 1984 - PSPICE
- 1984 - Eldo
- 1986 - SPECTRE
- 1989 - SPICE3

The SPICE Era

- SPICE Applications were
  - Analog circuits (small)
  - Critical paths in digital circuits
  - Memories
- SPICE Algorithms were tuned to go faster but not work smarter
- Model development was driven by technology evolution and digital circuits
- Emergence of “funny” circuits (such as switched C) which required special simulators (such as SWITCAP)

SPICE Version of Moore's Law

SPICE CPU = Timepoints

* (Newton Iterations / Timepoint)

* (CPU / Newton Iteration / Transistor)

* (Transistors)

SPICE Version of Moore's Law

- Timepoints increase by at least $\sqrt{2}$ every two years
- Newton Iterations / Timepoint is constant
- CPU / Newton Iteration / Transistor is simply the CPU required to evaluate a device model. This has been relatively constant.
- Transistors increase by at least $\sqrt{2}$ every two years
- This is at least an N^2 Process!!!

SPICE Version of Moore's Law

- Fortunately, computer CPUs get faster $\sqrt{2}$ every two years

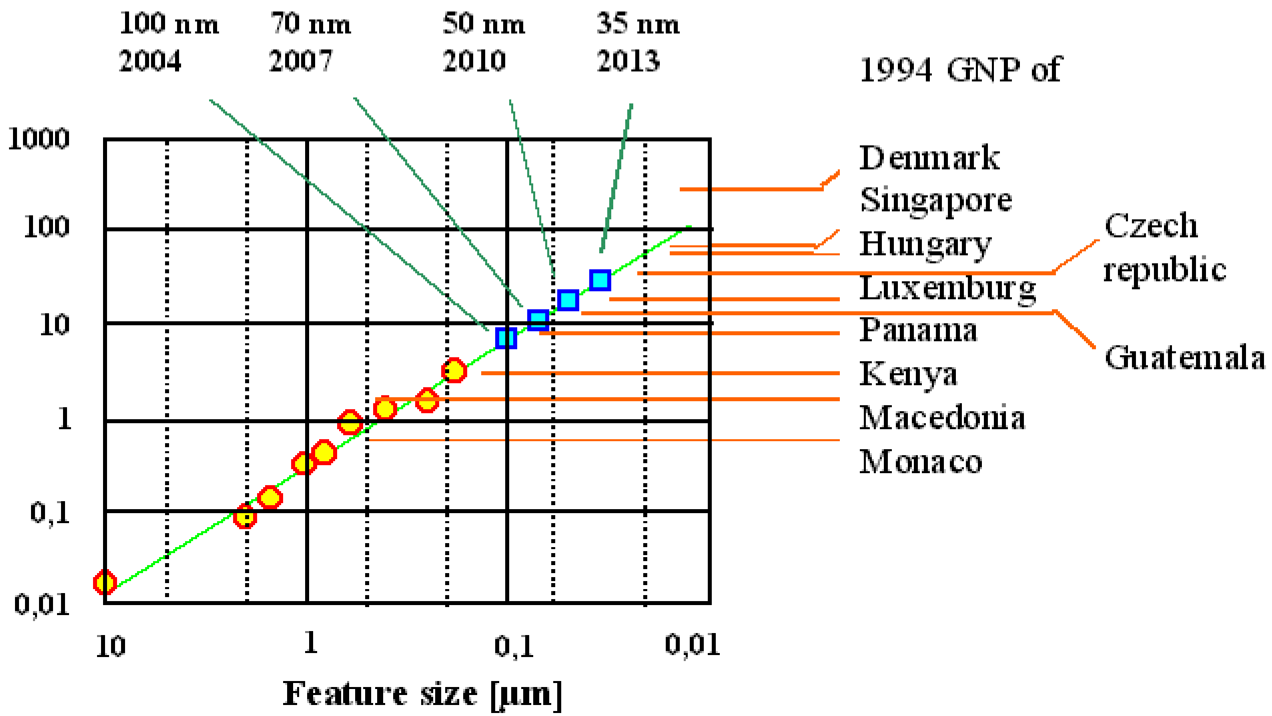
SPICE Version of Moore's Law

- Fortunately, computer CPUs get faster √2 every two years

Still ...

- SPICE CPU consumption doubles every four years!!!

IC Development Cost



- **SPICE is one contributor to this**

H. Föll, 2004

Challenges of the '90s

- By the end of the 1980's, at around the 1µm technology node, it was clear that smarter techniques were necessary
- There were numerous problems brought on by shrinking design rules
- As transistors became faster, it became possible to integrate RF circuits and the wireless explosion was on.
- This necessitated an entirely new line of algorithms and simulators

RF Simulation

- 1988 - Microwave Design System (MDS)
- 1991 - Libra
- 1994 - ADS
- 1996 - SPECTRE RF
- 1998 - Eldo RF
- 2004 - HSPICE RF

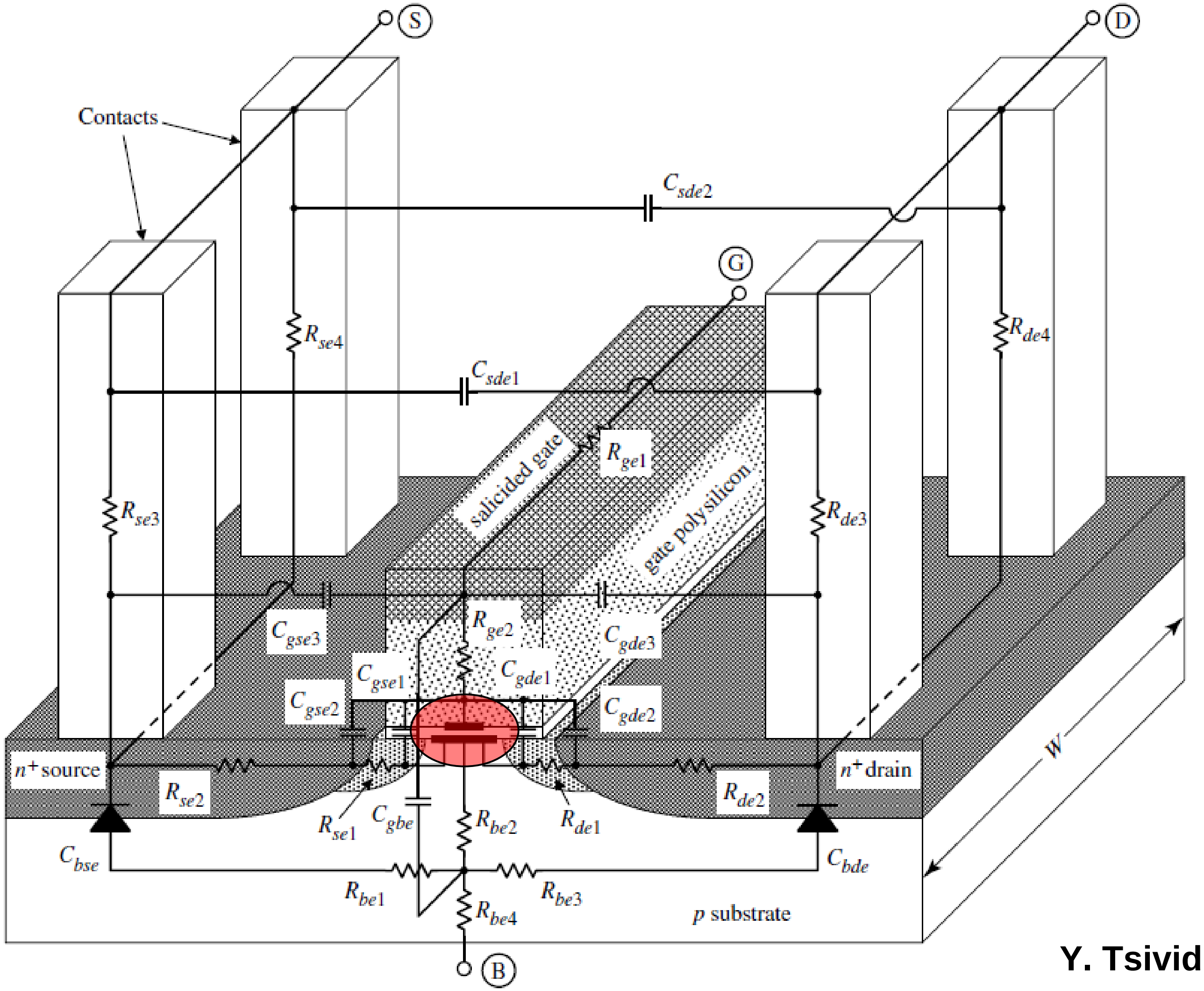
SPICE Development Drivers

- Smaller Dimension
- Higher Frequencies
- More Complexity

Consequences of Smaller Dimensions

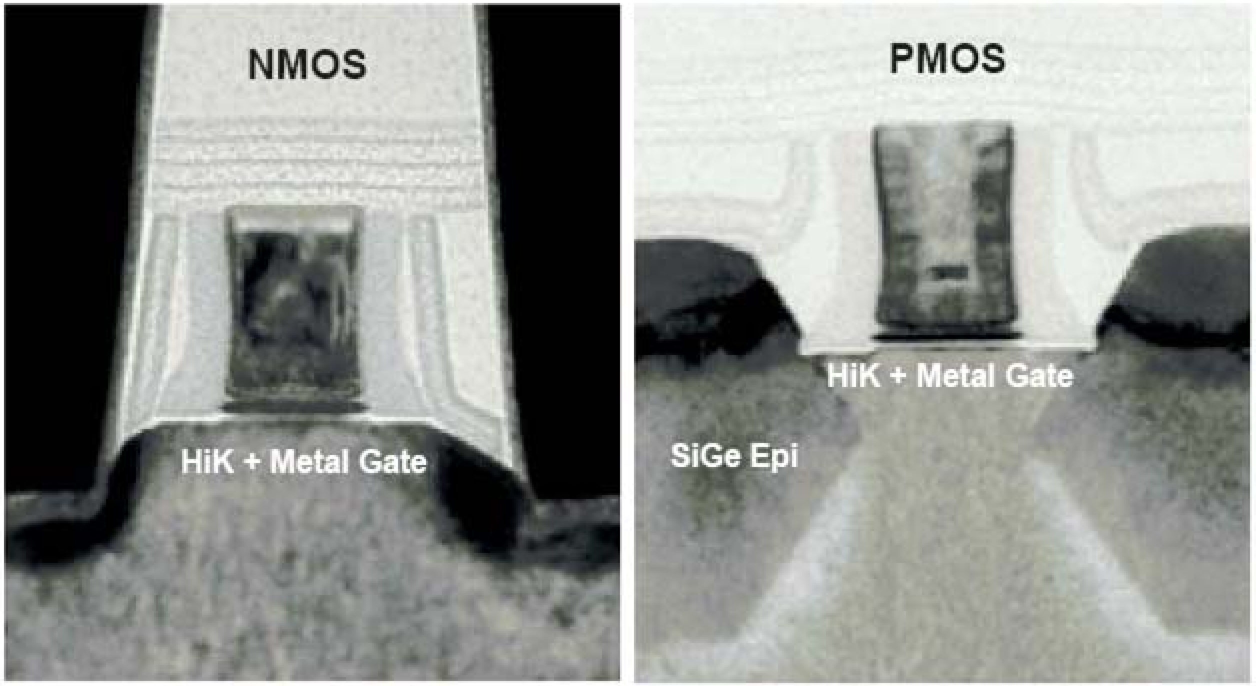
- Device variability and mismatch becomes increasingly important; using usual ± 3σ files misses important effects of mismatch
- Devices are placed closer to each other, increasing unintended coupling between devices
- Reduced design rules increase proximity effects, from wells and stress
- Reduced design rules increase the importance of parasitics, and result in incredible complexity of "accurate" extracted netlists

Parasitics are Critical



Y. Tsividis and C. McAndrew, 2010

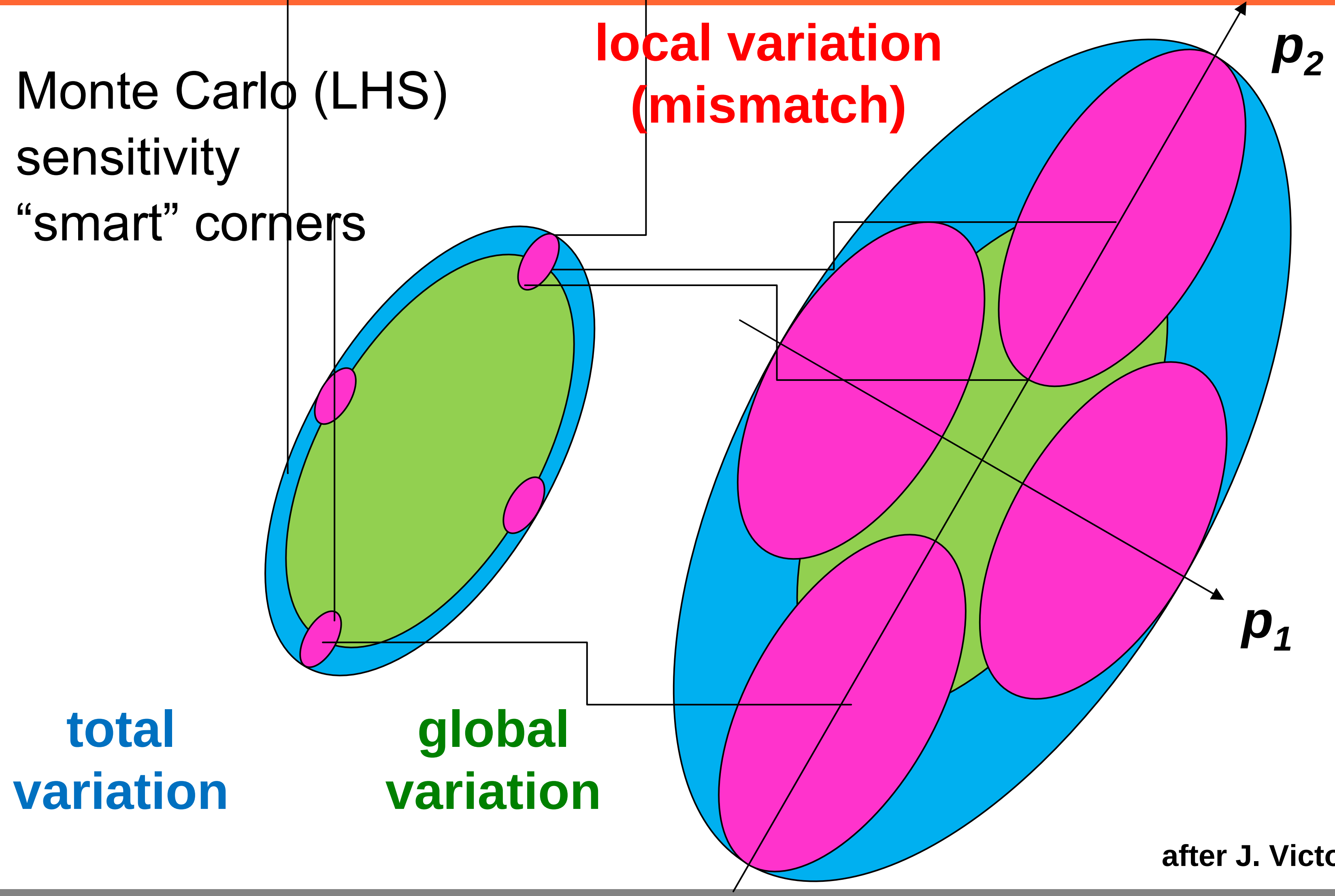
“Scaling” is Now Exotic Materials, Profiles, Strain



C. Auth, Intel Tech. Journal, 12, 2008

Correlation Structure of Variability Changes

- Monte Carlo (LHS)
- sensitivity
- “smart” corners



after J. Victory, 2010

Consequences of Higher Frequencies

- Reduced device dimensions results in devices that can operate at much higher frequencies, which requires models that are accurate at higher frequencies
- RF operation requires a special class of simulator and algorithms that can simulate RF performance
- RF performance (distortion, phase noise) is much more sensitive to second-order device effects and derivatives of charge and current

Consequences of More Complexity

- Increased device density enables the integration of complex systems that include digital, RF, and analog circuitry on the same chip
- More complex chips require more ancillary circuitry for performing self testing, self calibration, and many different modes of operation
- Complex circuits require advanced simulation algorithms (especially in RF) and specialty analyses in MATLAB or Python Scripts

Back To The Future

- SPICE is still around because it is a fairly general purpose tool that fills an important need
- SPICE is based on the simple but realistic assumption that an electronic circuit can be described by a set of nonlinear ordinary differential equations
- The electrical properties of the circuit then can be determined by solving the set of equations for various specialized cases such as DC, AC, transient, and periodic steady-state.

How SPICE Needs To Evolve

- SPICE needs a more general circuit description language like Verilog-A or VHDL-A
- SPICE needs to become more efficient
- New platforms are required that will allow multiple copies of SPICE to simulate different portions of a large circuit simultaneously

High Level SPICE Circuit Description Language

- In the future, electronic circuits will be built with radically different elements
- SPICE will need to accommodate new elements with greater ease than is now possible
- SPICE needs to understand a circuit description language such as Verilog-A or VHDL-A
- This would give circuit designers and model developers the same flexibility that software engineers have in a programming language

A More Efficient SPICE

- Exploit the “obvious parallelism” in model evaluation using Graphics Processing Units
- Exploit the “obvious latency” in large circuits
- Exploit behavioral modeling and simulation techniques to simplify complex problems

Concurrent SPICE Simulations

- Large systems need multiple copies of SPICE simulating different portions of the circuit simultaneously
- This is an important application of parallel computing that requires effective interprocess communication
- Behavioral modeling and simulation techniques will facilitate concurrent simulations

Is SPICE Good Enough for Tomorrow's Analog?

Yes !!!

- Because SPICE is fairly general purpose, it will play an important role in IC design for a long time
- SPICE will not simulate billion transistor circuits, but it will be used for the device model and the low-level blocks
- Perhaps a million copies of SPICE will simulate a billion transistor circuit!