Design-Based “Equivalent Scaling” to the Rescue of Moore’s Law

Andrew B. Kahng
UCSD CSE and ECE Departments

abk@ucsd.edu
http://vlsicad.ucsd.edu
Conclusions

- A new technology node costs billions of dollars in technology development and hundreds of millions of dollars in design enablement.
- Leading-edge companies accept these costs to gain “20%” advantages.
- **Design-based equivalent scaling** offers entire technology nodes of improvements that are essential to the continuation of “Moore’s Law”.
- Recurring theme: “What if we knew…”
  - Bridges between design and manufacturing
  - Bridges between system design and IC implementation
What is “Moore’s Law”? 

- Moore, 1965: “The complexity for minimum component costs has increased at a rate of roughly a factor of two per year”

- Moore’s Law is a law of cost reduction
- Proxy for cost reduction: “scaling of value”
- Proxies for value: “bits”, “hertz”, “density” (= utility, integration)
What Is Scaling?

Intel CPU Trends
(sources: Intel, Wikipedia, K. Olukotun)

# of Transistors
Clock Frequency
Power
Performance/CLK (ILP)

[Sutter09]
Dimension and Transistor Density

• ITRS = International Technology Roadmap for Semiconductors (http://www.itrs.net/)

• Key metric of progress: Metal-1 (M1) half-pitch (F)

• M1 HP scales by 0.7x (note: 0.7 x 0.7 = 0.49 \(\Rightarrow\) density doubles) at each “technology node”

• Rough equivalences:
  • Pitch of M1 \((P_{M1}) = 2F\)
  • Pitch of M2 \((P_{M2}) = 1.25P_{M1}\)
  • Pitch of polysilicon \((P_{poly}) = 1.5P_{M1}\)

  \[\text{Model scaling in both X, Y directions}\]
Basic SRAM, Logic Circuits and Layouts

- Models of SRAM ($U_{SRAM}$) and NAND2 ($U_{NAND2}$) area based on canonical layouts [ISOCC09, ITRS 2009]

\[ U_{SRAM} = 2P_{poly} \times 5P_{M1} = 60F^2 \quad U_{logic} = 3P_{poly} \times 8P_{M2} = 180F^2 \]
Historical Data for MPU Products

\[ U_{\text{SRAM}} = 2P_{\text{poly}} \times 5P_{M1} = 60F^2 \]
\[ U_{\text{logic}} = 3P_{\text{poly}} \times 8P_{M2} = 180F^2 \]

![Graphs showing historical data for MPU products](image)
Frequency

- Figure from 2001 *International Technology Roadmap for Semiconductors* (ITRS) System Drivers Chapter: FO4 INV delays in clock period of Intel microprocessors

Observation: Microarchitecture (pipelining) lever runs out of gas ~2004

Limit: 12-14 FO4 delays
Static power density and “active capacitance” (= dynamic power) density both continue to increase, modulo small resets (high-k, FDSOI, FinFET, …)
ITRS MPU Frequency Roadmap

- **Device speed only**: 41% per year
- **Platform power limit**: 17% per year
- **Device scaling limit**: 8% per year
- **4% per year**

Frequency (GHz)
ITRS MPU Frequency Roadmap

-Danowitz et al., Stanford CPUDB-
Seeing the Future, With 20-20-20 Vision

- **TSMC 28nm → 20nm:** 30% higher speed, 25% less power
- **TSMC 40nm LP → 28nm LP:** 20% higher speed
- **UMC 40nm LP → 28nm LP:** 20% higher speed
- **Samsung 45nm → 32nm:** 30% higher speed, 30% less power
Seeing the Future, With 20-20-20 Vision

Reality: In a new technology node, the best that designers can hope for is 20% less power, 20% more speed, and 20% better density.

Corollary: 10% = half of a technology node that costs many $B

Challenge: How to extract value from new technology ?!?
This Challenge is Due Largely to **Margins**

**Margin** ⇒ **lost benefits of technology**

Design quality (e.g., frequency) vs. Technology Nodes

Guardbands

Lost benefits!

Signoff with larger guardbands

Nominal Scaling
What Can The Semiconductor Industry Do?

• “Surrender”
  • Don’t turn on the transistors: “dark silicon”
“Dark Silicon” Analysis in 2001 ITRS

- Power management gap ⇒ amount of (switched) logic content in an SOC goes to zero
- Unfortunately, chip value also goes to zero

![Graph showing the percentage of area devoted to logic over time with two lines: Constant Power (90W) and Constant Power Density (90W/1.57cm^2). The graph indicates a decline in the percentage of area devoted to logic from 1998 to 2014.]
What Can The Semiconductor Industry Do?

• “Surrender”
  • Don’t turn on the transistors: “dark silicon”
  • Don’t use the transistors as much: less activity
ITRS “Magical” Activity Factor Reduction

• To reduce dynamic power: **Do less work**
• MPU power limit is maintained by assuming a “design-based” reduction of switching activity (-5% per year)

![Graph showing total dynamic power and new total dynamic power with 5% per year reduction of switching activity.](image-url)

**With 5% per year reduction of switching activity**

**Power < 150W**
What Can The Semiconductor Industry Do?

• “Surrender”
  • Don’t turn on the transistors: “dark silicon”
  • Don’t use the transistors as much: less activity

• “Fight”
  • Design-based equivalent scaling!
  • = the rest of this talk

• (There is a third choice)
  • Retire 😊
“Design-Based Equivalent Scaling”

- **Geometric scaling**: Reduction of physical dimensions to improve density (cost per function), performance, reliability, etc.
- **Examples**: scaling of $T_{ox}$, $L_{gate}$, gate pitch
• Equivalent scaling: Non-geometric enhancements of process, devices or materials to improve electrical performance
• Examples: High-K metal gate, FinFET devices
“Design-Based Equivalent Scaling”

• Design-based equivalent scaling: Design technologies that achieve power, performance and cost tradeoffs to rescue Moore’s-Law scaling of value

• Examples: design for variability, low-power design, heterogeneous multi-core architectures, …

… including some research at UCSD …
“Design-Based Equivalent Scaling”

• Design-based equivalent scaling: Design technologies that achieve power, performance and cost tradeoffs to rescue Moore’s-Law scaling of value

• Rest of this talk: 4 vignettes
  • The cost of margins
  • Mitigating “bimodal” variations
  • Adaptivity
  • “What if we knew…”
On the Cost of Margin (a.k.a. Guardband)
Review: Concept of Timing Slack

Basic idea of power optimization: “convert” positive timing slack into power reductions: smaller transistors, area, power, ... *(but this is not easy!)*

\[
\text{Slack} = T_{\text{required}} - T_{\text{arrival}}
\]
Review: Concept of Timing Slack

Basic idea of power optimization: “convert” positive timing slack into power reductions: smaller transistors, area, power, … *(but this is not easy!)*

\[ \text{Slack} = T_{\text{required}} - T_{\text{arrival}} \]

Transistors in positive-slack cells can have smaller \( W_{\text{gate}} \), higher \( V_{\text{th}} \), larger \( L_{\text{gate}} \), more variation, …
Guardband for Variations

• Guardband to cover uncertainties

- Defocus/Dose Variation
- Non-Rectangular Shapes
- Line-End Shortening
- Line Edge Roughness
- Non-Uniform CD
- Imperfect regulators
- IR-drop
- Crosstalk
- Temperature Variation
- Reliability
- Misalignment
- Erosion/Dishing in CMP
- Lens Aberration
- Wafer flatness
- Mask CD Error
- Flare
- Alpha-Particle
- NBTI
- Electromigration
- Hot-Carrier Injection
- Process
- Voltage
- Temp.
- Voltage
- Temperature
- Voltage
- Temperature

<table>
<thead>
<tr>
<th></th>
<th>Process</th>
<th>Voltage</th>
<th>Temp.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FEOL</td>
<td>BEOL</td>
<td></td>
</tr>
<tr>
<td>NMOS</td>
<td>Cap.</td>
<td>Res.</td>
<td></td>
</tr>
<tr>
<td>PMOS</td>
<td>Slow</td>
<td>Slow</td>
<td></td>
</tr>
<tr>
<td>WORST</td>
<td>Low (e.g. 0.9V)</td>
<td>High (e.g. 125°C)</td>
<td></td>
</tr>
<tr>
<td>BEST</td>
<td>Fast</td>
<td>Fast</td>
<td></td>
</tr>
<tr>
<td></td>
<td>High (e.g. 1.1V)</td>
<td>Low (e.g. -40 °C)</td>
<td></td>
</tr>
</tbody>
</table>
Motivating Study: Guardband Reduction [ISQED08]

• What is the true benefit of design/manufacturing optimization techniques?
• 50% guardband reduction

• From delay table analysis:
  • Worst case delay \(\Rightarrow 12.5\%\) reduction
• From capacitance table analysis:
  • Worst case cap. \(\Rightarrow 4\%\) reduction

\[ Y_r = e^{-Ad} \]
\((d: \text{defect density})\)

\[ N_{\text{dies}} = \pi \left( \frac{r^2}{A} - \frac{2r}{\sqrt{2A}} \right) \]
\((r: \text{wafer radius})\)
Design Outcomes from Guardband Reduction

- 40% guardband reduction
  - Area: 13% reduction
  - Dynamic power: 13% reduction
  - Leakage power: 19% reduction
  - Wirelength: 12% reduction
  - SP&R runtime: 28% reduction
  - #Timing viols.: 100% reduction
  - #Good dies (w/ process enhancement): 10% increases
  - #Good dies (w/o process enhancement): 4% increase

Impact of guardband reduction
⇒ insight into costs of guardband
Impact on Yield

- **Guardband reduction in design process** (Actual guardband of fabrication is unchanged)
- Parametric yield will decrease
- Random defect yield will increase

- 20% guardband reduction results in 4% increase in total number of good dies per wafer
On Taming Bimodality
(Double-Patterning Lithography)

TSMC R&D VP Cliff Hou: “At 20nm the challenge is double patterning, …”
– October 24, 2012
CD Bimodality in Double-Patterning Litho

• Two patterning steps $\rightarrow$ Two different CDs
  CD = “Critical Dimension”

- Green lines from 1st patterning
- Blue lines from 2nd patterning

• Two different colorings $\rightarrow$ Two different timings

C12-type cell
- Odd polys in BLUE,
- Even polys in GREEN

C21-type cell
- Odd polys in GREEN,
- Even polys in BLUE
Bimodality Impact on Guardband [SPIE08, ASPDAC09]

- Comparison of design guardband (Min-Max delay)
  - Unimodal representation is too pessimistic
Impact of Bimodality on Path Delay

- Bimodality can help reduce path delay variation
- Reduction of covariance when alternately colored

Variation ($\delta$) is accumulated

Variation ($\delta$) is compensated

SPICE Simulation Results
Impact of Bimodality on Clock Skew

• Different coloring sequences in a clock network

⇒ Clock skew

<table>
<thead>
<tr>
<th>Case</th>
<th>Source to Sink A</th>
<th>Source to Sink B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$C_{12} + C_{12} + C_{12} + \ldots + C_{12}$</td>
<td>$C_{12} + C_{12} + C_{12} + \ldots + C_{12}$</td>
</tr>
<tr>
<td>2</td>
<td>$C_{12} + C_{12} + C_{12} + \ldots + C_{12}$</td>
<td>$C_{21} + C_{21} + C_{21} + \ldots + C_{21}$</td>
</tr>
</tbody>
</table>

- Same color on all clock buffers is better!
Bimodal CD Distribution: 3 Key Facts

1. Design requires **bimodal-aware timing models**
   - Unimodal representation is too pessimistic

2. Data paths benefit from **alternate (mixed) coloring**
   - Exploit existence of two **uncorrelated** CD populations
   - Minimize correlated variations in a given path

3. Clock paths benefit from **uniform** coloring
   - Correlated variation between launch and capture paths
     minimizes bimodality-induced clock skew

⇒ **Principle:** Design can exploit both correlated, uncorrelated variations
DPL Layout-to-Mask Flow

- RTL-to-GDS
- DPL Mask Coloring
- Bimodal-Aware Timing Analysis
- Optimization 1: Maximization of Alternate Coloring (Datapaths)
- Optimization 2: Placement Perturbation for Color Conflict Removal (Clock and Data paths)

Alternate coloring using integer-linear programming

Coloring conflict > Minimum resolution

Placement perturbation using dynamic programming
Overall Timing Improvement

- Bimodal timing model → Reduce pessimism (margin)
- Alternate coloring → Improve timing
- Placement perturbation → Remove conflicts

<table>
<thead>
<tr>
<th>Stage</th>
<th>#Conflict</th>
<th>Timing Metric</th>
<th>Mean CD Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>2nm</td>
</tr>
<tr>
<td>Initial Coloring (Unimodal)</td>
<td>0</td>
<td>WNS (ns)</td>
<td>-1.113</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TNS (ns)</td>
<td>-671.1</td>
</tr>
<tr>
<td>Initial Coloring (Bimodal)</td>
<td>0</td>
<td>WNS (ns)</td>
<td>-0.191</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TNS (ns)</td>
<td>-8.17</td>
</tr>
<tr>
<td>Alternative Coloring</td>
<td>219</td>
<td>WNS (ns)</td>
<td>-0.090</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TNS (ns)</td>
<td>-1.48</td>
</tr>
<tr>
<td>DPL-Corr (+ECO Routing)</td>
<td>0</td>
<td>WNS (ns)</td>
<td>-0.104</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TNS (ns)</td>
<td>-3.43</td>
</tr>
</tbody>
</table>

Bimodality impact can be effectively mitigated!
On Adaptivity
Adaptive Voltage Scaling Approaches

Power

Open Loop AVS
- Pre-characterize LUT
- Post-silicon characterization

Closed-Loop AVS
- Generic monitor
- Design dependent replica
- In-situ monitor

Error Detection System
- In-situ performance monitor
- Measure actual critical paths
- Error detection and correction system
- $V_{dd}$ scaling until error occurs

Application Driven AVS
- Loading-aware AVS (software technique)
  - Application-driven $V_{dd}$ and frequency scaling

Process-aware AVS
- Post-silicon characterization

Process and temperature-aware AVS
- Generic on-chip monitor
- Design-dependent monitor

Application-driven $V_{dd}$ and frequency scaling

[Martin02, Tschanz03, Burd00, Elgebaly07, Drake08, Chan12, Hartman06, Fick10, Das06, Tschanz10, Lin09]
Design-Dependent RO

- Timing variability is design-specific → why use generic monitor?
- Idea: Select gates to form DDROs with similar delay sensitivity to variations (Lgate, Vth, V, T, …) as actual critical paths
- Benefits: low area overhead, automated flow, standard cells only
- Can cluster critical paths having similar sensitivities to reduce #ROs
For each cluster, synthesize a DDRO using integer linear program

Off-line or on-chip delay estimation

Gate sensitivities

Critical path sensitivities

Cluster critical paths

Delay sensitivity – temp. (%)

Delay sensitivity – Vdd (%)

Sum of delay sensitivities error (%)

X: cluster centroids

Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Average

DDRO

45nm SOI test chip

ARM Cortex M3

DDRO

[ISQED12]
Design-Dependent RO vs. Generic RO

- **SPICE Monte Carlo Simulation**
  - 30 samples

- **45nm test chip measurement**
  - Each monitor have 3 copies per chip, 19 chips (no wafer split)

---

**Estimation error**

- **DDRO**
  - Estimation error = $-0.5\% \sim 3.7\%$

- **hvt+rvt Inv RO**
  - Estimation error = $-1.7\% \sim 5.1\%$

---

**Std. Deviation of $\Delta F_{\text{max}}$**

- DDRO
- Critical path replica
- hvt+rvt INV RO

**$F_{\text{max}}$ Correlation Coefficient**

- DDRO
- Critical path replica hvt+rvt INV RO
Process-aware Voltage Scaling (PVS) [ICCAD-2012]

• Monitor design considerations
  • Critical path maybe difficult to be identified (IP from 3rd party)
  • Multiple modes/voltages $F_{\text{max}}$ calibration takes long test time

• Proposal: tunable monitor
  • Design monitor to guardband for arbitrary circuit (overdesign)
  • Tune monitor based on $F_{\text{max}}$ of sample chips to recover design margin (calibrate only once)

• Abstract voltage scaling property instead of matching critical path
  • Enable analysis of worst-case voltage scaling
Voltage Scaling Properties

- $V_{\text{min}} = \text{Minimum } V_{dd}$ to meet timing constraints = process distance/scaling rate
- Process distance: process-induced frequency shift relative to target frequency
- Scaling rate: frequency shift for a unit voltage difference

$$f_{\text{target}} \leq \frac{\Delta f}{\Delta V}$$
PVS Monitor Design Concept

- RO is used as a reference for voltage scaling
- Design ROs with the worst case voltage scaling properties → guardband for arbitrary circuits
- A circuit meets its timing when

\[ \max_{i=1}^{m} (V_{\text{min}_\text{ro}}(i,k)) > \max_{j=1}^{n} (V_{\text{min}_\text{path}}(j,k)) \]

Maximum of m ROs Maximum of n paths

- Design challenges
  - \( V_{\text{min}_\text{ro}} > V_{\text{min}} \) of any data path across all process conditions
**$V_{\text{min}}$ Analysis**

- **Key observation:** $V_{\text{min}}$ is bounded by NMOS or PMOS dominated cells (e.g., NOR3 at FS corner)
  - Use NAND, NOR type ROs

<table>
<thead>
<tr>
<th>Cell type</th>
<th>SS</th>
<th>TT</th>
<th>FF</th>
<th>SF</th>
<th>FS</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVX0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAND2X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAND3X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAND4X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOR2X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOR3X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOR4X0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Graph showing $V_{\text{min}}$ analysis](image_url)
Design RO with Tunable $V_{\text{min}}$

- Identified two circuit knobs to tune $V_{\text{min}}$
  - Series resistance
  - Cell types (INV, NAND, NOR)
- Example circuit strategy
  - Allow tuning of series resistance of each stage to high or low
  - Different cell types cover different process corners
PVS Experiment Result

- Default setting: low resistance in all stages
  \[ V_{\text{min\_est}} - V_{\text{min\_chip}} = 13\text{mV} \text{ on average (guardband for worst-case)} \]

- With \( F_{\text{max}} \) information per die, can tune RO configuration to drive \( V_{\text{min\_est}} - V_{\text{min\_chip}} \rightarrow 0 \)

- Better on-chip sensing and adaptation \( \Rightarrow \) more reduction of runtime power overheads (\( V_{\text{dd}} \))

Monte Carlo SPICE simulation

65nm, OpenSPARC T1 module
On “What if We Knew …”
What If We Knew...(switching activity from workload)

Error-Tolerant Design

CPU, heal thyself ...

Errors are detected and corrected with redundancy technique

Problem:
- Many paths have near-critical slack → wall of (critical) slack
- Scaling beyond the critical operating point causes massive errors that cannot be corrected

Reshape slack distribution for gracefully increasing error rate

Frequently-exercised paths:
: upsize cells

Rarely-exercised paths:
: downsize cells

Scale voltage further
Recovery-Driven for Error-Tolerant Designs

- Minimize power for a target error rate
- Slack redistribution based on functional information

Voltage Scaling
- reduce voltage until the error rate exceeds a target

Path Optimization
- optimize frequently exercised, negative slack paths

Power Reduction
- reduce power without affecting error rate
What If We Knew … (scenarios, duty cycles)

Dynamic Voltage Freq. Scaling

- DVFS allows adaptation to workloads & operating conditions
- Multi-Mode (or DVFS) design operates at multiple power/performance points with different lifetimes

1.0V, 1GHz (e.g., talk mode)
0.7V, 100MHz (e.g., standby mode)

- Conventional EDA tool: require constraints (freq., voltage) before implementation (which constraints will provide minimum energy?)
- Replication: Create replicas that target each performance mode (Replication incurs a large area overhead)

Use scenario/duty cycle information for multi-mode optimization [TCAD12]
DVFS Design Implementations

- Context-aware design shows up to 19.5%, 7.6% (avg.) energy reduction over conventional multi-mode design
- Replication-based design shows up to 25.4%, 9.1% (avg.) energy reduction over conventional multi-mode design
- **Selective-replication design**

![Graph showing energy reduction vs. allowable area overhead]

- **Layout results (OpenSPARC/FFU)**
  - 16% power reduction with 10% area overhead (R=1%)
  - FFU module has 12% energy savings through selective-replication
What If We Knew … (accuracy requirements)

Problem:
• Accuracy requirement can change during runtime → benefits of approximation could be reduced

Approximate Design
What is the square root of 10?
“a little more than three”
“3.162278…”
Approximation could be faster and more powerful

Adapt to changing requirements with runtime accuracy configuration

[DAC 2012]
“accuracy-configurable approximate adder”

lower power consumption
higher accuracy
Accuracy-Configurable Adder

- Accuracy configuration with pipelined adder

- Power reduction when accuracy requirement varying

\[ \text{Accuracy} = \text{Avg.} \left( 1 - \frac{|\text{result} - \text{reference}|}{\text{reference}} \right) \]

Average 30% power savings vs. no accuracy configuration
What If We Knew … (Lifetime (MTTF) Reqts)

Direct relation; if A increases then B increases

Inverse relation; if A increases then B decreases

Tunable at design or runtime

Tunable at design
Example: Electromigration MTTF vs. $F_{\text{max}}$

- $F_{\text{max}}$ increases with relaxing MTTF$_{\text{require}}$
- Up to +60% of $F_{\text{max}}$ for -30% of MTTF$_{\text{require}}$

$F_{\text{max}}$ improvement is determined by:
- Mix of cell sizes
- Length and timing constraints of critical paths

- 65nm technology
- Fixed area

![Graph showing MTTF vs. $F_{\text{max}}$ for different applications (DMA, AES, JPEG)]
Conclusions

• A new technology node costs billions of dollars in technology development and hundreds of millions of dollars in design enablement

• Leading-edge companies accept these costs to gain “20%” advantages

• *Design-based equivalent scaling* offers entire technology nodes of improvements that are essential to the continuation of “Moore’s Law”

• Recurring theme: “What if we knew…”
  • ⇒ Bridges between design and manufacturing
  • ⇒ Bridges between system design and IC implementation
THANK YOU