Energy Efficiency and Resilience in Future ICs

Andrew B. Kahng
UCSD VLSI CAD Laboratory
abk@cs.ucsd.edu
http://vlsicad.ucsd.edu

CSE and ECE Departments
University of California, San Diego
Part I: Power Crisis

The Power Gap

Low-Power Design

Beyond Low-Power: Resilience

Conclusion
Value From Semiconductor Scaling

- Value is enabled by integration
  - Greater utility
  - Less cost

- Scaling enables new products
  - Density: more functions per chip
  - Device: better performance

- Challenge: Maintain Scaling of Value
  - Variability $\Rightarrow$ larger design guardband
  - Cost $\Rightarrow$ technology and design risk
  - Leakage $\Rightarrow$ waste of increasingly expensive energy
Scaling of Density

Layout density increase → Capacitance density increase

Capacitance density + 10% / year

\[ P_{\text{dynamic}} \propto CV^2 \]
Scaling of Product

Required performance for multimedia processing (GOPS: Giga Ops/Sec)

2007 ITRS Consumer-Stationary SOC Driver: 220 TFlops on a single chip by 2022
Scaling of Device

To meet the performance scaling...

\[ I_{ds} \propto \mu C_{ox} (V_{dd} - V_{th})^\alpha \]

- Mobility enhancement
- \( V_{dd} \) slowly lowering
- \( V_{th} \) lowering
- \( C_{ox} \propto 1/t_{ox} \) increasing
- \( t_{ox} \) lowering
- Gate leakage
- Subthreshold leakage
Scaling of Device

Graph showing the relationship between Power Consumption and Physical Gate Length from 1990 to 2020. Key points include:
- Sub-Threshold Leakage
- Gate-Oxide Leakage
- Dynamic Power
- Cross-Over
- Possible trajectory for high-k dielectrics
- Technology Node

Power Consumption is measured on a logarithmic scale, while Physical Gate Length is measured on a linear scale.
The Power Gap

- Capacitance
- Functions
- Frequency
- Tox, Vth

- Density
- Product
- Device

- P_{dynamic}
- P_{leakage}

Higher VDD ? → Quadratic \( P_{dynamic} \) increase

Lower VDD ? → Lower Vth
→ Exponential \( P_{leakage} \) increase
Power Limits the Technology Roadmap

High-Performance Device Intrinsic Speed ($1 / \tau$)
($\tau = CV / I$)

Normalized frequency

Normalized to 2001

17% / year
(2001 ITRS)

13% / year
(2009 ITRS)

8% / year
(2011 ITRS)
(tentative / planned)

View ITRS MPU Model
Digression: Concept of Timing Slack

Many power optimizations “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 higher $V_{th}$, larger $L_{\text{gate}}$, more variation, …
Even If We Slow Down Frequency Scaling …

**Cap. scaling:**
\[
\frac{2 \left( \equiv Tr.\ density \right)}{\sqrt{2} \left( \equiv L_{gate} \right)} / 2 \text{ years}
\]

**Freq. scaling:**
4% / year

---

**Roadmap of low-power techniques!**

**Power Gap >> 100x**

**Practical Power Limit**

---

**Clock Gating**

**Multi-Vth**

**Multi-Core Arch**

**L_{gate} Bias***

**Power Gating***

**Adaptive Body Bias**

**Multi-Vdd**

**DVFS***

* Work at UCSD
Clock Gating $\rightarrow$ $P_{\text{dynamic}}$ Reduction

Clock gating

Dynamic Power Reduction

Leakage Power Reduction

UCSD
Andrew B. Kahng 110425
Multi-Vth $\rightarrow P_{\text{leakage}}$ Reduction

Dynamic Power Reduction

Leakage Power Reduction

Critical Timing Path

Clock gating

Multi-Vth

Low-$V_{th}$  High-$V_{th}$

Clock gating

Technology node (nm)

Cumulative efficiency (%)
Multi-Core Architecture $\rightarrow$ $P_{\text{dynamic}}$ Reduction

Dynamic Power Reduction

Clock gating

Multi-Vth

Multi-cores

Figure 1. Multi-core Performance & Power Scenario

Over-clocked (+20%) 1.13x
Max Frequency 1.00x
Dual-core (-20%) 1.02x

M. Domeika, drdobbs.com, Dec. 27, 2008
Gate Length Biasing $\rightarrow P_{\text{leakage}}$ Reduction

**Dynamic Power Reduction**

**Leakage Power Reduction**

- Clock gating
- Multi-Vth
- Multi-cores
- Body biasing
- Power gating
- $L_{\text{gate}}$ biasing

<table>
<thead>
<tr>
<th>Technology node (nm)</th>
<th>Dynamic Power Reduction</th>
<th>Leakage Power Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>350</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>300</td>
<td>270%</td>
<td>270%</td>
</tr>
<tr>
<td>250</td>
<td>300%</td>
<td>300%</td>
</tr>
<tr>
<td>200</td>
<td>270%</td>
<td>300%</td>
</tr>
<tr>
<td>150</td>
<td>525%</td>
<td>9000%</td>
</tr>
<tr>
<td>100</td>
<td>8500%</td>
<td>16200%</td>
</tr>
<tr>
<td>50</td>
<td>4000%</td>
<td>7000%</td>
</tr>
<tr>
<td>0</td>
<td>5000%</td>
<td>35000%</td>
</tr>
</tbody>
</table>
Transistor Gate-Length Biasing  

- Bias Impact
  - Exponential
    - \( (I_{\text{sub}}) \) Leakage reduction
    - Variability reduction
  - Linear
    - Performance reduction
- \( \Rightarrow \) Apply very small biases (+2nm, +4nm, etc.) just before tapeout
Transistor Gate-Length Biasing  

⇒ Challenging global optimization over millions of gates, with complex timing constraints

(Spent two years developing a leading industry tool...)
**Transistor Gate-Length Biasing**

Andrew B. Kahng

- **Bias Impact**
  - Exponential
    - \( I_{\text{sub}} \) Leakage reduction
  - Variability reduction
  - Linear
    - Performance reduction
- \( \Rightarrow \) Apply very small biases (+2nm, +4nm, etc.) just before tapeout
  - Chip-scale optimization: trade timing slack for leakage power everywhere possible
  - UCSD-patented flow currently offered in TSMC’s Green “Power Trim” service
  - Energy savings for just AMD/ATI Radeon GPUs: >> 10^9 watt-hours
Power Gating $\rightarrow P_{\text{leakage}}$ Reduction

Dynamic Power Reduction

Leakage Power Reduction

Clock gating
Multi-Vth
Multi-cores

Gating
Multi-Vth
Multi-cores

Body biasing

Power gating

Andrew B. Kahng 110425
Power Gating

- Typical operation modes: active, idle

- Power gating: cut off leakage path during predefined idle modes

Typical operation with power gating:

Without power gating

With power gating

More than 10x leakage saving during idle mode
New: ‘Runtime’ Power Gating

• Clock gating and power gating address dynamic and leakage power during idle mode
• But in active mode, cycles are still wasted
  • E.g., execution time spent waiting for memory access

• What If…. Circuit design (wakeup logic) to enable faster, more flexible wakeup of one or more cores?

⇒ shorter power gating intervals

⇒ “runtime” power gating
Token-Based Power Gating

- Architectural power gating: Send tokens to control core power gating

- About tokens:
  - Sent by cache or memory controller
  - Received by core
  - Stamped with system cycle in which it was generated
  - Has estimated request wait latency
  - Sets minimum core wakeup latency
  - Managed by token controller

Joint work with Tajana Rosing and Rick Strong, UCSD
Token-Based Power Gating

• About token controller:
  • Manages token properties
  • Queries cores for performance information
  • Maintains peak current constraint by managing core wakeup latencies
  • Maximizes energy savings, e.g., by balancing wakeup latencies for parallel apps, or increasing wakeup latencies for underutilized cores

Joint work with Tajana Rosing and Rick Strong, UCSD
System Model and Tool Flow

- Multi-core assumptions
  - 4 cores
  - Private L1 (32KB-2way-0.5ns), L2 caches (2MB-2way-9ns)
  - MESI cache coherence protocol
  - Memory:
    - Size = 2GB
    - Latency = 40ns
  - Core
    - Type: In-order EV4
    - ISA: ALPHA64
    - Frequency: 2GHZ
    - Width: 2
  - OS: Vanilla-Linux-2.6.27

- Tools
  - M5 Full-System Simulator (Spec2006, Parsec2.0,Splash2.0 benchmarks)
  - McPAT used to generate power numbers
Wakeup Latencies vs. Energy Saving

- Different memory hierarchy levels have different latencies
  - L1 hit latency = 0.5ns // L2 hit latency = 9ns // Memory Latency = 40ns
  - Lowering the core wakeup latency (10ns → 5ns → 2ns) can make power gating for smaller idle periods more attractive

Joint work with Tajana Rosing and Rick Strong, UCSD
And More: ABB / Multi-VDD / DVFS / NTC / ...

Dynamic Power Reduction

Leakage Power Reduction

Multi-VDD, AVS
DVFS, AVS
RTL-Opt
Data gating
DC-DC efficiency
Near-threshold computing

Multi-Vth
Multi-cores
Clock gating
Power gating
L_{gate} biasing
Body biasing

Technology node (nm)

Cumulative efficiency (%)
DVFS: Dynamic Voltage/Frequency Scaling

- DVFS enables
  - Operation at multiple power-performance points
  - Adaptation to different operating conditions or modes
- Observation 1: DVFS changes only voltage and frequency, not the design itself \( \Rightarrow is \ a \ fixed \ design \ always \ optimal? \)
- Observation 2: Lifetime energy changes with scenario \((R \times X)\) \( \Rightarrow is \ scenario-oblivious \ design \ always \ optimal? \)

![Diagram showing different duty cycle (R) and different frequency scaling (X)]

e.g., talk mode
- Low Performance \((1 - R)\)
- High Performance \((R)\)
e.g., standby mode

Joint work with Rakesh Kumar and John Sartori, UIUC
DVFS Suboptimality #1

- No single design can work well in all modes: “jack of all trades, master of none”

- **What If: selective replication**
  - Replication benefits are different in each module
  - Optimal use of replication = knapsack formulation

Multi-mode design

- Example: CTL module has 12% energy savings through replication
**DVFS Suboptimality #1**

- No single design can work well in all modes: “jack of all trades, master of none”

**What If: selective replication**
- Replication benefits are different in each module
  - Optimal use of replication = knapsack formulation
  
  $\Rightarrow$ avg 9% energy savings
DVFS Suboptimality #2

- Lifetime energy is not optimal

Design A saves energy in high-perf mode.
Design B saves energy in low-perf mode.
Neither is optimal for lifetime energy.

\[ E = P_{hi} \times R + P_{lo} \times (1-R) \]

- Why this happens

<table>
<thead>
<tr>
<th>operating point</th>
<th>Timing slack (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freq. voltage</td>
<td>Path A</td>
</tr>
<tr>
<td>1.0GHz 0.95V</td>
<td>0.047 -0.109</td>
</tr>
<tr>
<td>200MHz 0.60V</td>
<td>-2.429 0.108</td>
</tr>
</tbody>
</table>

How to optimize for multiple performance modes?
Context-Driven Multi-Mode Design

- Goal: Find a minimum lifetime energy design
- Conventional design flow sets constraints (frequency, voltage) before implementation (but the min-energy constraints are unknown!)
- What If: context-driven multi-mode design: design to the scenario rather than to constraints \( \Rightarrow \) avg 8% energy savings

Joint work with Rakesh Kumar and John Sartori, UIUC
The Power Gap (roadmapped by UCSD since 2001)

Low-Power Design (Lgate biasing, runtime power gating, scenario-aware and replication-based DVFS)

Beyond Low-Power: Resilience

Conclusion
New Mindset

- Better-than-worst-case (typical case) design
- Dynamic reliability (error) management
- Living with variations

Further Energy Reductions

Resilient Design

- Limit of Worst-Case design
- Typical-Case + error-tolerance

Voltage scaling (↓)

Energy / Instruction
# Types of Resilience

<table>
<thead>
<tr>
<th>Error Acceptance</th>
<th>Error Tolerance</th>
<th>Error Avoidance</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Allow errors</td>
<td>• Detect and correct errors dynamically</td>
<td>• No error allowed</td>
</tr>
<tr>
<td>• Approximate computation for accuracy insensitive applications</td>
<td>• Error detection FF + architectural correction schemes</td>
<td>• DVFS + canary circuits</td>
</tr>
</tbody>
</table>

## Key Ideas
- Allow errors
- Approximate computation for accuracy insensitive applications
- Detect and correct errors dynamically
- Error detection FF + architectural correction schemes
- No error allowed
- DVFS + canary circuits

## UCSD Works
- Approximate arithmetic design
- Recovery-driven design
- Design-Dependent Ring Oscillator
The Power Gap

Low-Power Design

Beyond Low-Power: Resilience

- Error Tolerance
  - Recovery-Driven Design

- Error Acceptance
  - Approximate Arithmetic Logic

- Error Avoidance
  - Design-Dependent Ring Oscillator

Conclusion
Recovery-Driven Design

- Motivation #1: If the design uses an error-tolerance mechanism, then the design process should be modified accordingly

- Motivation #2: “Error rate” demands the use of functional information
Recovery-Driven Design

• Low-power methodology for error-tolerant designs
  • Minimize power for a target error rate
  • Slack redistribution with functional information

Voltage Scaling $\rightarrow$ Path Optimization $\rightarrow$ Power Reduction

reduce voltage until the error rate exceeds a target
optimize frequently exercised, negative slack paths
reducing power w/o affecting error rate

Joint work with Rakesh Kumar and John Sartori, UIUC

[HPCA10] [DAC10]
Recovery-Driven Design: Experimental Results

- Path toggling extraction and error rate estimation

- Power comparison across design techniques

- Fast (20X)

- Accurate

- 25% power savings w/ 2% error

- 22% power savings w/ Razor flip-flop
Resilient Overhead Reduction (Ongoing Work)

• Resilient overhead: For the resilience, design overheads are required, i.e., additional circuit and operations (pipeline flush)

• New tradeoffs in resilient design
  
  \[ \text{Pros.} \quad \text{tradeoff} \quad \text{Cons.} \]
  
  \[ \text{ET register} \uparrow \]
  
  avoid over-design, voltage scaling further
  
  cost for ET registers, recovery overhead

Goal:
Minimize the cost function (power) using the tradeoffs

Approach:
Find optimal assignment of registers (error-tolerant or normal)
The Power Gap

Low-Power Design

Beyond Low-Power: Resilience

- Error Tolerance
  - Recovery-Driven Design

- Error Acceptance
  - Approximate Arithmetic Logic

- Error Avoidance
  - Design-Dependent Ring Oscillator

Conclusion
Error Avoidance

• Adjust Vdd, Freq. according to delay margin
  • No error → recovery mechanism not required

Conventional approaches

• Inverter-based RO:
  • Critical paths have different sensitivity to process variations
• Critical path RO:
  • Replicating critical path with long interconnect costs area
Monitor for Resilience: DDRO

- **Problem:** Measure real-time performance variation in an adaptive system

- **Approach:** Select gates to form design-dependent ring oscillators (DDROs) with similar delay sensitivity to variations \((L_{\text{gate}}, V_{\text{th}}, \text{Tox}, V, T, \ldots)\) as actual critical paths

- **Potential Benefits:**
  - Specific to path’s rising or falling transition
  - Can cluster critical paths having similar sensitivities to reduce number of RO
  - Low area overhead
  - Automated design flow, standard cells only
For each cluster, synthesize a DDRO using integer linear program

45nm SOI test chip

45nm SOI test chip
Monte Carlo Simulation Results (30 samples)

Without within-die variation modeling

Estimated delay (ns) vs Actual delay (ns) for DDRO, Critical path RO, and Inv. RO.

- DDRO: Estimation error: -1.4% ~ 3.7%
- Critical path RO: Estimation error: -2.0% ~ 4.1%
- Inv. RO: Estimation error: -4.3% ~ 7.1%

With within-die variation modeling

Estimated delay (ns) vs Actual delay (ns) for DDRO, Critical path RO, and Inv. RO.

- DDRO: Estimation error: -0.5% ~ 3.7%
- Critical path RO: Estimation error: -1.3% ~ 3.6%
- Inv. RO: Estimation error: -1.7% ~ 5.1%
The Power Gap (roadmapped by UCSD since 2001)

Low-Power Design (Lgate biasing, runtime power gating, scenario-aware and replication-based DVFS)

Beyond Low-Power: Resilience (recovery-driven design, approximate arithmetic, design-dependent RO)

Conclusion
The Elephant

- Big picture for power and resilience spans
  - Software and applications
  - Architectures
  - Interconnects
  - Memories
  - Circuits and devices
  - Technology
  - Fundamental limits

- Different “animal” from recent talk topics: Design for Manufacturability, Technology Roadmap, 22nm Chip Implementation, 3D PDN Pathfinding, …!
What I Spend My Time On  http://vlsicad.ucsd.edu/~abk/

- **Connecting and building:** dots, bridges, big pictures, ...
- The IC Design-Manufacturing Interface
- The ITRS roadmap
- IC physical design (clustering, placement, interconnect design, …)
- NOC modeling and optimization (ORION2.0, trace-driven optimizations, …)
- Utilities for teaching (math drill generator, “automatic editor”, …)
- Current projects
  - MARCO Gigascale Systems Research Center: “Physical Architecture Components: Models, Roadmaps and Integrations” = system-level impacts of 3D, new memories, new design optimizations
  - UC Discovery: “Integrated Modeling, Process, and Computation for Technology (IMPACT)” Center (Design-Manufacturing Interface)
  - **SRC (with UIUC):** “New Directions in Architecture and Design of Scalable Energy Constrained SoCs”
  - SRC (with UCLA): “New Directions in Design-Aware Manufacturing”
  - NSF (with UCLA): “Research on Benchmarking and Robustness of VLSI Sizing Optimizations”
  - Qualcomm: “Power Delivery Pathfinding for 3D Through-Silicon Stacking”
  - STMicroelectronics: “Across-Field Variation Mapping From Silicon Measurements, and Design-Driven DoseMap Flow”
Thank You!