Power Management from Smartphones to Data Centers

Thomas Wenisch
Morris Wellman Faculty Dev. Asst. Prof. of CSE
University of Michigan

Acknowledgements:
Luiz Barroso, Anuj Chandawalla, Laurel Emurian, Brian Gold, Yixin Luo, Milo Martin, David Meisner, Marios Papaefthymiou, Steven Pelley, Kevin Pipe, Arun Raghavan, Chris Sadler, Lei Shao, Wolf Weber
A Paradigm Shift In Computing

- Transistors (100,000's)
- Power (W)
- Performance (GOPS)
- Efficiency (GOPS/W)

Limits on heat extraction
Limits on energy-efficiency of operations

IEEE Computer—April 2001
T. Mudge
A Paradigm Shift In Computing


- Transistors (100,000's)
- Power (W)
- Performance (GOPS)
- Efficiency (GOPS/W)

Limits on heat extraction
Limits on energy-efficiency of operations
Stagnates performance growth

IEEE Computer—April 2001
T. Mudge

Power: A First-Class Architectural Design Constraint
Four decades of Dennard Scaling

• \( P = C V^2 f \)
• Increase in device count
• Lower supply voltages
→ Constant power/chip

Dennard et. al., 1974

Robert H. Dennard, picture from Wikipedia
Leakage Killed Dennard Scaling

Leakage:
- Exponential in inverse of $V_{th}$
- Exponential in temperature
- Linear in device count

To switch well
- must keep $V_{dd}/V_{th} > 3$

$\Rightarrow V_{dd}$ can’t go down
No more free lunch...

- Need system-level approaches to...
  - ...turn increasing transistor counts into customer value
  - ...without exceeding thermal limits
- Energy efficiency is the new performance

Today’s talk
- Computational Sprinting
  - Improving responsiveness for mobile systems by briefly exceeding thermal limits
- Power mgmt. for Online Data Intensive Services
  - Case study of server power management for Google’s Web Search
Computational Sprinting

Arun Raghavan*, Yixin Luo+, Anuj Chandawalla+, Marios Papaefthymiou+, Kevin P. Pipe#+,
Thomas F. Wenisch+, Milo M. K. Martin*

University of Pennsylvania, Computer and Information Science*
University of Michigan, Electrical Eng. and Computer Science+
University of Michigan, Mechanical Engineering#
Computational Sprinting and Dark Silicon

• A Problem: “Dark Silicon” a.k.a. “The Utilization Wall”
  – Increasing power density; can’t use all transistors all the time
  – Cooling constraints limit mobile systems

• One approach: Use few transistors for long durations
  – Specialized functional units [Accelerators, GreenDroid]
  – Targeted towards sustained compute, e.g. media playback

• Our approach: Use many transistors for short durations
  – Computational Sprinting by activating many “dark cores”
  – Unsustainable power for short, intense bursts of compute
  – Responsiveness for bursty/interactive applications

Is this feasible?
Sprinting Challenges and Opportunities

• Thermal challenges
  – How to extend sprint duration and intensity?
    Latent heat from *phase change material* close to the die

• Electrical challenges
  – How to supply peak currents? **Ultracapacitor/battery hybrid**
  – How to ensure power stability? **Ramped activation (~100μs)**

• Architectural challenges
  – How to control sprints? **Thermal resource management**
  – How do applications benefit from sprinting?

  *6.3x responsiveness for vision workloads on a real Core i7 testbed restricted to a 10W TDP*
Power Density Trends for Sustained Compute

How to meet thermal limit despite power density increase?
Mobile devices limited to passive cooling
Option 2: Decrease Chip Area?

Reduces cost, but sacrifices benefits from Moore’s law
Option 3: Decrease Active Fraction?

How do we extract application performance from this “dark silicon”?
Design for Responsiveness

• Observation: today, design for sustained performance

• But, consider emerging interactive mobile apps...
  [Clemons+ DAC’11, Hartl+ ECV’11, Girod+ IEEE Signal Processing’11]
  – Intense compute bursts in response to user input, then idle
  – Humans demand sub-second response times
    [Doherty+ IBM TR ‘82, Yan+ DAC’05, Shye+ MICRO’09, Blake+ ISCA’10]

Peak performance during bursts limits what applications can do
Designing for Responsiveness

COMPUTATIONAL SPRINTING
Parallel Computational Sprinting

- Power
- Temperature $T_{ma}$
Parallel Computational Sprinting

Effect of *thermal capacitance*
Parallel Computational Sprinting

Effect of *thermal capacitance*
Parallel Computational Sprinting

Effect of \textit{thermal capacitance}

\begin{align*}
\text{power} & \\
\text{temperature} & = T_{ma}
\end{align*}
State of the art:
Turbo Boost 2.0 exceeds sustainable power with DVFS (~25% for 25s)

Effect of *thermal capacitance*
Hardware Testbed

- Quad-core Core i7 desktop
  - Heat sink removed
  - Fan tuned for 10W TDP
- Power profile
  - Idle: 4.5 W
  - Sustainable (1 core 1.6GHz): 9.5 W
  - Efficient sprint (4 core 1.6GHz): ~20 W
  - Max sprint (4 core 3.2GHz): ~50 W
- Temperature profile
  - Idle: ~45°C
  - Max safe: 78°C
- 20g copper heat spreader on package
  - Can absorb ~225J heat over 30°C temperature rise

Models a system capable of 5x max sprint intensity
Max sprint: 3s @ 3.2GHz, 19s @ 1.6GHz
Responsiveness & Energy Impact

Sprint for responsiveness (3.2GHz): 6.3x speedup
Race-to-idle (1.6GHz): 7% energy savings (!!!)
Extending Sprint Intensity & Duration: Role of Thermal Capacitance

- Current systems designed for thermal *conductivity*
  - Limited capacitance close to die

- To explicitly design for sprinting, add thermal *capacitance* near die
  - Exploit latent heat from phase change material (PCM)
PCM Heat Sink Prototype

• Aluminum foam mesh filled with Paraffin wax
  – Relatively form-stable; melting point near 55°C
  – Working on a fully-sealed prototype w/ thermocouples
Demo of PCM melting

Nickel-plated Copper fins; paraffin wax
Impact of PCM prototype

PCM extends max sprint duration by almost 3x
Power Management of Online Data-Intensive Services

David Meisner†*, Christopher M. Sadler‡, Luiz A. Barroso‡, Wolf-Dietrich Weber‡, Thomas F. Wenisch†

†The University of Michigan  *Facebook  ‡Google, Inc.
Power: A first-class data center constraint

- Installed base grows 11%/yr.
- 2.5% of US energy
  - $7.4 billion/yr.

Annual data center CO₂:
- 17 million households

Lifetime Cost of a Data Center
- Peak power determines data center capital costs

Improving energy & capital efficiency is a critical challenge
Online Data-Intensive Services [ISCA 2011]

• Challenging workload class for power management
  – Process TBs of data with $O(\text{ms})$ request latency
  – Tail latencies critical (e.g., 95th, 99th-percentile latency)
  – Provisioned by data set size and latency not throughput
  – Examples: web search, machine translation, online-ads

• Case Study: Google Web Search
  – First study on power management for OLDI services
  – Goal: Identify which power modes are useful
The need for energy-proportionality

How to achieve energy-proportionality at each QPS level?
Two-part study of Web Search

• Part 1: Cluster-scale throughput study
  – Web Search on O(1,000) node cluster
  – Measured per-component activity at leaf level
  – Use to derive upper-bounds on power savings
  – Determine power modes of interest/non-interest

• Part 2: Single-node latency-constrained study
  – Evaluate power-latency tradeoffs of power modes
  – Can we achieve energy-proportionality with SLA “slack”

Need coordinated, full-system active low-power modes
Background: Low-power mode taxonomy

- **Active Modes** – Reduce speed of component
  - Continue to do work (e.g., DVFS)
- **Idle Modes** – Put component into sleep mode
  - No useful processing (e.g., ACPI C-states)

### Spatial Granularity

- **DVFS**
- **ACPI C-states**
- **MemScale [ASPLOS’11]**
- **Self-refresh**
- **Dual-speed disks**
- **Spin-down**

### Balanced scaling (open problem)

- Consolidation & shutdown

- **PowerNap [ASPLOS’09, TOCS’11]**
Background: Web Search operation

Query: “Ice Cream”

What if we turn off a fraction of the cluster?
Web Search operation

Google

Ice cream

About 177,000,000 results (0.24 seconds)

Billionaire Boys Club / Ice Cream
In 2005 Pharrell Williams, one half of the Grammy winning production team The Neptunes, along with his manager Robert Walker, partnered with Japanese...

Ice cream - Wikipedia, the free encyclopedia
Ice cream is a frozen dessert usually made from dairy products, such as milk and cream, and often combined with fruits or other ingredients and flavours. History - Production - Commercial delivery - Dietary

Images for ice cream

Ben & Jerry's Homemade Ice Cream
Official site for Ben & Jerry's super premium ice cream, light ice cream, sorbet, frozen yogurt and novelties. Find flavor listings and nutrition facts for...

Places for ice cream near Ann Arbor, MI

A Washenaw Dairy - 88 reviews - Place page
B Stucchi's Ice Cream - 76 reviews - Place page
C Ben & Jerry's Ice Cream - 7 reviews - Place page
D Stucchi's Ice Cream - 26 reviews - Place page
E Dairy Queen - 6 reviews - Place page
F Cold Stone Creamery - 9 reviews - Place page

What if we turn off a fraction of the cluster?
Web Search operation

Cluster-level techniques cause data unavailability
Study #1: Cluster-scale throughput

- Web Search experimental setup
  - O(1,000) server system
  - Operated at 20%, 50%, 75% of peak QPS
  - Traces of CPU util., memory bandwidth, disk util.

- Characterization
  - Goal: find periods to use low-power modes
  - Understand *intensity* and *time-scale* of utilization
  - Analyze using *activity graphs*
CPU utilization

How often can we use the mode?

What is the intensity of activity?

We can use a mode with a 2x slowdown 45% of the time on time scales of 1ms or less.
CPU utilization

50% of Max QPS

1 ms granularity sufficient

Very little Idleness > 10ms
CPU utilization

CPU active/idle mode opportunity from a bandwidth perspective

Percent of Time

CPU utilization

50% of Max QPS

Very little Idleness > 10ms

1 ms granularity sufficient

Time Scale

100 μs 1ms 10ms 100ms 1s 10s

© 2013 Thomas Wenisch
Memory bandwidth utilization

50% of Max QPS

Percent of Time

Time Scale

100 ms 1s 10s 100s 1000s

50% 30% 10% Idle

Significant Periods of Under-utilization

No Significant Idleness
Memory bandwidth utilization

- CPU
- Mem
- Disk
- System
- Cluster

Insufficient idleness for memory idle low power mode

Disk transition times too slow
See paper for details

Irrelevant similarities for system
Irrelevant for details
Study #2: Leaf node latency

- Goal: Understand latency effect of power modes

- Leaf node testbed
  - Faithfully replicate production queries at leaf node
  - Arrival time distribution critical for accurate modeling
  - Up to 50% error from Naïve loadtester

- Validated power-performance model
  - Characterize power-latency tradeoff on real HW
  - Evaluate power modes using Stochastic Queuing Simulation (SQS) [EXERT ’10]
• Scarce full-system idleness in 16-core systems
• PowerNap [ASPLOS ’09] with batching [Elnozahy et al ‘03]
Full-system coordinated idle modes

- Scarce full-system idleness for multicore

- PowerNap [ASPLOS ’09] with batching [Elnozahy et al ‘03]

**Batching + full-system idle modes ineffective**
• We assume CPU and memory scaling with $P \sim f^{2.4}$
• Optimal Mix requires coordination of CPU/memory modes
Full-system coordinated active scaling

- We assume CPU and memory scaling with $P \sim f^2.4$.
- Optimal Mix requires coordination of CPU/memory modes.

95th-Percentile Latency Increase

Power (Percent of peak)

1x 1.2x 1.4x 1.6x 1.8x 2x

- Single-component active low-power modes insufficient.
Comparing power modes

- Allow SLA “slack” – deviation from 95th-percentile latency
Comparing power modes

- Allow SLA "slack" – deviation from 95th percentile latency

Core-level power modes provide negligible power savings

Only coordinated active modes achieve proportionality
OLDI power management summary

- OLDI workloads challenging for power management

- Cluster-scale study
  - Current CPU power modes sufficient
  - Massive opportunity for active modes for memory
  - Need faster idle and active modes for disk

- Latency-constrained study
  - Individual idle/active power modes do not achieve proportionality
  - PowerNap + batching provides poor latency-power tradeoffs

Need coordinated, full-system active low-power modes to achieve energy proportionality for OLDI workloads
Sponsors

For more information

http://www.eecs.umich.edu/~twenisch
Backup Slides
Typical data center utilization

Low utilization (≤20%) is endemic

- Provisioning for peak load
- Performance isolation
- Redundancy

*But, historically, vendors optimize & report peak power*

Source: Barroso & Hölzle, Google 2007
Idle periods are short

Most idle periods are < 1 Sec in length

[Meisner '09]
Background: PowerNap [ASPLOS’09, TOCS’11]

Full System Idle Low-Power Mode

- Full-system nap during <1s idle periods
  - OS detects idleness, triggers transition
  - Transparent to user software
  - Exploits deepest component sleep modes

Energy proportional if transition << avg. work
Average power

DVFS saves rapidly, but limited by $Pwr_{cpu}$

PowerNap becomes energy-proportional as $T_t \to 0$
Response time

DVFS response time penalty capped by $f_{min}$

PowerNap penalty negligible for $T_t \leq 1\text{ms}$
Nap power is ~10W, but PSU uses additional 25W
PSU also limiting factor for transition
PowerNap & multicore scaling

- Request-level parallelism & core scaling thwart PowerNap
  - Full-system idleness vanishes...
  - ... even at low utilization
  - *Per-core idleness unaligned*

- “Parking” (Package C6) saves little
  - Automatic per-core C1E on HLT already very good

*Need to create PowerNap opportunity via scheduling*
Background: MemScale [ASPLOS’11]  
Active Low-Power Mode for Main Memory

• Goal: Dynamically scale memory frequency to conserve energy

• Hardware mechanism:
  – Frequency scaling (DFS) of the channels, DIMMs, DRAM devices
  – Voltage & frequency scaling (DVFS) of the memory controller

• Key challenge:
  – Conserving significant energy while meeting performance constraints

• Approach:
  – Online profiling to estimate performance and bandwidth demand
  – Epoch-based modeling and control to meet performance constraints

System energy savings of 18%  
with average performance loss of 4%
MemScale frequency and slack management

Estimate performance/energy via models

CPU

MC, Bus + DRAM

Epoch 1
Epoch 2
Epoch 3
Epoch 4

Time

High Freq.
Low Freq.

Pos. Slack
Neg. Slack
Pos. Slack

Actual
Profiling
Target

Calculate slack vs. target

Estimate performance/energy via models
In Brief: Computational Sprinting [HPCA 2012]

- Many interactive mobile apps are **bursty**
- Power density trend leading to “**dark silicon**” (esp. mobile)
  - Today, we design for sustained performance
  - Our goal: *design for responsiveness*

- Computational Sprinting
  - Intensely, but briefly exceed Thermal Design Power (TDP)
  - Buffer heat via **thermal capacitance** using phase change material

10.2x responsiveness via 16-core sprints within 1W TDP