Extreme Scale Computer Architecture: Energy Efficiency from the Ground Up

Josep Torrellas
Department of Computer Science
University of Illinois at Urbana-Champaign
http://iacoma.cs.uiuc.edu

Design, Automation & Test in Europe (DATE)
March 2014
Wanted: Energy-Efficient Computing

• **State of the Art:**

  ![Blue Waters Supercomputer](image)

  Performance: 11 PF
  Power: 6-11 MW (idle to loaded)
  10MW = $10M per year electricity

  University of Illinois Blue Waters Supercomputer

• **Extreme Scale computing:** 100-1000x more capable for the same power consumption and physical footprint
  • Exascale \(10^{18}\) ops/cycle datacenter: 20MW
  • Petascale \(10^{15}\) ops/cycle departmental server: 20KW
  • Terascale \(10^{12}\) ops/cycle portable device: 20W
Recap: How Did We Get Here?

- **Ideal Scaling (or Dennard Scaling):** Every semiconductor generation:
  - Dimension: 0.7
  - Area of transistor: $0.7 \times 0.7 = 0.49$
  - Supply Voltage $V_{dd}$, C: 0.7
  - Frequency: $1/0.7 = 1.4$
  - Constant dynamic power density

- **Real Scaling:** $V_{dd}$ does not decrease much.
  - If too close to threshold voltage ($V_{th}$) $\rightarrow$ slow transistor
  - Dynamic power density increases with smaller tech
  - Additionally: There is the static power
  - Power density increases rapidly
Design for E Efficiency from the Ground Up

• New designs for chips with 1K cores:
  – Efficient support for high concurrency
  – Data transfer minimization

• New technologies:
  – Low supply voltage ($V_{dd}$) operation
  – Efficient on-chip voltage regulation
  – 3D die stacking
  – Resistive memory
  – Photonic interconnects
Thrifty Multiprocessor

1,000 core chip

Stacked DRAM

CPU module

Boad

Cabinet

- Funded by DOE, DARPA, NSF, Intel
- Similar to Runnemedede project funded by DARPA UHPC [HPCA2013]
Low Voltage Operation

- $V_{dd}$ reduction is the best lever for energy efficiency:
  - Big reduction in dynamic power; also reduction in static power
- Reduce $V_{dd}$ to bit higher than $V_{th}$ (Near Threshold Voltage--NTV)
  - Corresponds to $V_{dd}$ of about 0.5-0.55V rather than current 1V

- Advantages:
  - Potentially reduces power consumption by more than 40x
- Drawbacks as of now:
  - Lower speed (1/10)
  - Higher variation in gate delay and power consumption
Basics of Parameter Variation

- Deviation of device parameters from nominal values: e.g., $V_{th}$, $L_{eff}$
Variation in the Thrifty Manycore

- Larger f variation at NTV
- Memories more vulnerable
- Power varies as much
Multiple Vdd Domains at NTV: Costly [HPCA13]

• On chip regulators have a high power loss (10+%) 

• Large chip:  
  • If coarse-grain (multiple-core) domains → already has variation inside the domain 

• Small Vdd domain more susceptible to load variations  
  • Larger Vdd droops → need increase Vdd guardband
Needed: Efficient On-Chip $V_{dd}$ Regulation

- Voltage regulators (VRs) with a hierarchical design:
  - First level VRs: placed on a different die of 3D chip
  - Second level VRs: small range, high efficiency, fast (Low-dropout VRs)

- Energy-efficient design requires short Vdd guardbands
  - Need to tackle voltage droops due to load variation

From Nam Sung Kim, Univ. Wisconsin
Streamlined 1K-core Architecture

- Very simple cores (no structures for speculative execution)
- Cores organized in clusters with memory to exploit locality
- Each cluster is heterogeneous (has one large core)
- Special instructions for certain ops: fine-grain synch
- Exploring single address space without full hardware cache coherence
Managing Energy of On-Chip Memory

- On-chip memory leakage: major contributor of the NTV chip energy
- Industry is moving to dynamic memory for last-level caches
  - We propose Intelligent Refresh

- Use Intelligent Refresh
  - Do not refresh data that is not used (*Reprint*: HPCA-2013)
  - Asymmetric refresh leveraging spatial variations (*Mosaic*: HPCA-2014)
  - Asymmetric refresh leveraging temperature variations
Asymmetric Refresh Leveraging Spatial Variation

- Insight: retention time has spatial correlation. Why?
  - Retention time is a function of $V_{\text{th}}$
  - $V_{\text{th}}$ has spatial correlation due to process variation

Loss of charge in cell depends on the $V_{\text{th}}$ of access transistor
Mosaic: Organize the eDRAM in Tiles

- Organize eDRAM into tiles and profile the retention time
- Use different refresh rate per tile
- Eliminates 90+% of refresh
Managing Energy in On-Chip Network

• On-chip networks are especially vulnerable to variation:
  – They connect distant parts of the chip

• Proposal:
  – Organize network into multiple Vdd domains
  – Dynamically reduce Vdd of each domain differently while watching for errors
  – Each domain converges to a different Vdd
Motivation: Error Rate as Function of Vdd

- Process variation has a major impact on the network
Algorithm

• Independently change the Vdd for each domain
  – Periodically decrease Vdd of all domains

• Use switch-to-switch CRC to detect errors in a router

• On error: Controller increases Vdd of that domain

• Result for a 64-node mesh (1 router/domain):
  – Reduce the network energy consumption by avg. 35%
Minimizing Data Movement

- Thrifty has several techniques to minimize data movement:
  - Many-core chip organization based on clusters
  - Mechanisms to manage the cache hierarchy in software
  - Simple compute engines in the mem controllers ➔ Processing in Memory (PIM)
  - Efficient synchronization mechanisms
Micron’s Hybrid Memory Cube (HMC)
- Memory chip with 4 or 8 DRAM dies over 1 logic die
- Logic die handles DRAM control

Future use of logic die:
- Support for Intelligent Memory Operations?
  - Preprocessing data as it is read from memory
  - Performing processor commands “in place”
Programmability

• Programming highly-concurrent machines has required heroic efforts
• Extreme-scale architectures, with emphasis on power-efficiency, may make it worse
  – Need carefully manage locality and minimize communication
How to Program for High Parallelism?

- Expert programmers
  - Hooks to manage power and Vdd/frequency
  - Ability to map and control tasks
- Novice programmers:
  - High level programming models that express locality
    - *Hierarchical Tiled Arrays (HTA)*: computes in recursive blocks
    - *Concurrent Collections (CnC)*: computes in a dataflow manner
- Autotuning?
- ... open problem
Conclusion

• Presented the challenges of Extreme Scale Computing:
  • Designing computers for energy efficiency from the ground up
  • Described some of the architecture and design ideas
  • Programmability may suffer: need focus on the software
  • There is a tradeoff between energy efficiency and resilience
Extreme Scale Computer Architecture: Energy Efficiency from the Ground Up

Josep Torrellas
Department of Computer Science
University of Illinois at Urbana-Champaign
http://iacoma.cs.uiuc.edu

Design, Automation & Test in Europe (DATE)
March 2014
Energy-Efficiency Gap

• Goal:
  • 20W Tera-Op (sustained)
  • 20 pJoules/operation

• In comparison:
  • Power7 (2010): MCM 800W for 1TFlop Peak
    • Problem is harder than it looks:
      • Machines spend much of the energy transferring data
      • Minimizing E in data transfer, not ALU op is the challenge
Recap: How Did We Get Here?

- **Ideal Scaling (or Dennard Scaling):** Every semiconductor generation:
  - **Dimension:** 0.7
  - **Area of transistor:** $0.7 \times 0.7 = 0.49$
  - **Supply Voltage ($V_{dd}$), C:** 0.7
  - **Frequency:** $1/0.7 = 1.4$

**Constant dynamic power density**

\[
P_{dyn} \propto CV_{dd}^2 f
\]

Area: $A$

\[
\text{Power density: } CV_{dd}^2 f / A
\]

Area: $0.7^2 A$

\[
\text{Power density: } 0.7C 0.7^2 V_{dd}^2 1.4f / 0.7^2 A = CV_{dd}^2 f / A
\]
Recap: How Did We Get Here? (II)

• **Real Scaling**: $V_{dd}$ does not decrease much.
  
  – If too close to threshold voltage ($V_{th}$) $\rightarrow$ slow transistor
  
  – Delay of transistor is inversely prop to ($V_{dd} - V_{th}$)

  \[ T_g \propto \frac{V_{dd} L_{eff}}{\mu (V_{dd} - V_t)^\alpha} \]

  – Dynamic power density increases with smaller tech

• **Additionally**: There is the static power

  Power density increases rapidly
VARIUS-NTV Model [DSN-2012]

An architectural model of parameter variation tailored for NTV

Path delay

Vth, Leff maps

Model all failure types
- Rd Timing, WR Timing
- WR Stability, Hold
Energy Efficiency with a Single $V_{dd}$ Domain?

One $V_{dd}$ domain, many $f$ domains:
- Simple hardware, simple core allocation

- Each cluster in the chip is a $f$ domain
- Allocation in units of multiples of clusters called Ensembles
  - Whole ensemble clocked at a single $f$
- Simpler variation-aware core allocation
Effectiveness of Single $V_{dd}$ Domain per Chip

Single $V_{dd}$ is hard to beat.

288-core chip with 8-core clusters
Mosaic: Organize the eDRAM in Tiles

- Organize eDRAM into tiles and profile the retention time
- Use different refresh rate per tile
- Eliminates 90+% of refresh
Software Managed Caches (SMC)

- Instructions to perform explicit write-back and invalidation
- Read brings copy of line to cache from first level of cache it finds
- Writes do not invalidate/update other copies of the line

- Programmer/compiler inserts data-movement instructions at synchronization points
Supporting Fine-Grain Parallelism

- Synchronization and communication primitives
  - Efficient point-to-point synch between two cores
  - Dynamic hierarchical hardware barriers