Learning from Experimental Silicon like the SCC

Sebastian Steibl
Director Germany Microprocessor Lab
Intel Labs
Today’s presentations contain forward-looking statements. All statements made that are not historical facts are subject to a number of risks and uncertainties, and actual results may differ materially.

NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL’S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

Intel does not control or audit the design or implementation of third party benchmarks or Web sites referenced in this document. Intel encourages all of its customers to visit the referenced Web sites or others where similar performance benchmarks are reported and confirm whether the referenced benchmarks are accurate and reflect performance of systems available for purchase.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See www.intel.com/products/processor_number for details.

Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Intel Xeon, Intel Core microarchitecture, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others

Copyright © 2012, Intel Corporation. All rights reserved.
Agenda

Experimental Processors by Intel Labs
Lessons Learned from Experimental Silicon
  • Variability
  • Power Management
  • Energy Efficiency
  • Many-core Applications Research Community
Case Studies – Emerging Multicore / Manycore
Summary
Motivating Multicore & Manycore
Case Study: High Performance Computing
The Manycore Design Challenge

• **Hardware:** How many cores to put onto one die?
  - How many are useful? How to connect them for scalability?

• **Software:** Will anyone care?
  - Can “general purpose” programmers take advantage of the concurrency?

• **Design:** How to best implement them?
  - Can we use tiled architectures to reduce costs for design & validation?

Intel Labs “TeraScale” research program is addressing these questions with a series of experimental chips.
80 Core TeraScale Processor by Intel Labs

- **Goals**
  - Achieve 1+ TeraFLOPS @ <100 W
  - Demonstrate energy efficient architecture with fine-grain power management
  - Prototype high performance & scalable on-chip interconnect
  - Explore design methodologies for network on a chip architectures

- **Basic features**
  - 65 nm, 100 Million transistors
  - 8x10 tiles, 275 mm², 3mm²/tile
  - Mesochronous clock
  - 1.6 SP TFLOP @ 5 GHz and 1.2 V
  - 320 GB/s bisection bandwidth
  - Variable voltage, multiple sleep states

Single-chip Cloud Computer (SCC), Intel Labs

• **Achievements**
  – Scalable research microprocessor
  – 48 IA cores
  – Processors network resembles chip-level cloud computing
  – Fine-grained power management

• **Basic features**
  – Tiled architecture
  – 45 nm, 1.3 Billion transistors
  – 6x4 tiles, 567 mm$^2$, 18.7 mm$^2$/tile
  – Message passing architecture
  – 6 voltage domains for cores

Sharing as software research platform with industry and academic collaborators
Inside the SCC

Dual-core SCC Tile

- 24 Dual-core tiles (48 IA cores)
- 24 Routers
- Mesh network with 256 GB/s bisection bandwidth
- 4 Integrated memory controllers

Dual-core SCC Tile

- L2 Cache
- Core 1
- ROUTER
- Message Buffer
- Core 2

Inside the SCC

- 16K L2$ per core
- 256K L2$ per core
- 16K Message passing buffer

Inside the SCC

- Intel Pentium® class cores
  - 16K L1$ per core

Inside the SCC

- 1TILE

Inside the SCC
SCC $F_{\text{max}}$ Variation Small Compared to 80-core

[S. Dighe et al., A 45 nm 48-core IA Processor with Variation-Aware Scheduling and Optimal Core Mapping, 2011 Symposium on VLSI Circuits Digest of Technical Papers]
... but SCC $F_{\text{max}}$/Leakage Variation still Significant

[S. Dighe et al., A 45 nm 48-core IA Processor with Variation-Aware Scheduling and Optimal Core Mapping, 2011 Symposium on VLSI Circuits Digest of Technical Papers]
SCC: $F_{\text{max}}$ vs. Voltage Variation

Frequency spread of 8% at 1.1V and 30% at 0.65V
SCC Variation-Aware Application Mapping

- **VU**: variation unaware mapping (baseline).
- **VA-LV**: use cores with lowest leakage and V domain with lowest V for a given F.
- **MS**: mesh V/F adjusted dynamically to match bandwidth needs of the app.
- **PG**: set idle cores to lowest Voltage

---

[S. Dighe et al., A 45 nm 48-core IA Processor with Variation-Aware Scheduling and Optimal Core Mapping, 2011 Symposium on VLSI Circuits Digest of Technical Papers]
SCC Power Management

- On package voltage regulators
- 8 voltage islands and 24 frequency islands
- 8-cores per voltage domain
- 2-cores per frequency domain
- 2D-mesh on a separate voltage/ frequency domain

[S. Dighe et al., A 45 nm 48-core IA Processor with Variation-Aware Scheduling and Optimal Core Mapping, 2011 Symposium on VLSI Circuits Digest of Technical Papers]
Impact of Routers and Clocks

80 core, TeraScale testchip

- Global Clocking (1.3%)
- Memory (20.7%)
- Routers+mesh (28%)
- Cores* (50%)

Full Power, 1.2 V and 152 W

[Dighe et. al. Lessons Learned from the 80-core TeraScale processor, ITJ, Vol 13, 2009]

48 core, SCC testchip

- Global Clocking (2%)
- Memory (19%)
- Routers+mesh (10%)
- Cores (69%)

Full Power, 1.14 V and 125 W

[Howard et. al. 48 core SCC processor, ISSCC 2010]

*Core = FMAC+Registers + on-tile-synchronous clock (35%+4% + 11%)
Application-driven Power Management on SCC

- Fine-grain DVFS orchestrated by activities in application
  - MPI call patterns analyzed by phase predictor during runtime
- Hierarchical coordination of DVFS requests in software
  - Individual cores request state change
  - Managers per voltage domain select next DVFS states
- Significant energy improvements of 15% on average
  - for NAS Parallel Benchmarks
  - Joined work with the University of Edinburgh

ENERGY EFFICIENCY
Motivation for Lower $V_{\text{min}}$

- Today silicon operates over a constrained voltage range (typically 1.3 V to 0.7 V)
- Reducing $V_{\text{min}}$ can improve scaling and efficiency
  - Scaling: Compute scales to match varying workloads
  - Efficiency: Compute efficiency improves at lower voltages
- Coming Near Threshold Voltage (NTV)

**Goal of NTV research**
Develop circuits & architectures that enable a wide dynamic voltage range while preserving peak performance.
Prototyping a NTV Core

CHALLENGES

- Increased delay variability
- Low static noise margins
- Performance degradation
- Memory read/write failures
- Flops setup/hold delay degradation
- Failures in contention circuits

SOLUTIONS

- Modified Caches
- NTV-aware design optimizations
- Variation-aware design
Near Threshold Voltage Core

• IA concept chip that can tune from full/turbo performance to low power modes <10mW – wide dynamic range
• First processor to demonstrate benefits of Near Threshold Voltage (NTV) circuits
• Enables ultra low-power devices with wide dynamic operating range
• 32nm SoC low leakage technology

Capable of running off this solar cell
Measured Power and Performance

32nm CMOS, 25°C

[S. Jain et. al “A 280mV to 1.2V Wide Operating Range IA-32 Processor in 32nm CMOS”, ISSCC 2012]
Measured Energy Efficiency

32nm CMOS, 25°C

$\text{Energy/Cycle (nJ)}$

4.7X

$\frac{\text{Logic Vcc}}{\text{Memory Vcc}}$ (V)

[S. Jain et. al “A 280mV to 1.2V Wide Operating Range IA-32 Processor in 32nm CMOS”, ISSCC 2012]
MARC
MANY-CORE APPLICATIONS
RESEARCH COMMUNITY
Software Research on our Experimental Chips

• The 80 core chip ... 5 people at Intel did the SW research
  - Focus on computational kernels

• For SCC ... Embracing the community with external research program
  - Applications, OS, programming systems, middleware

Many-core Applications Research Community

• 154 contracts signed
• 102 unique institutions
• 50 Research partners in Europe
• 33 Research partners in US
• 19 Research partners in Other Countries
• 316 MARC website participants
MARC Symposia Publications

Papers

- Performance/Scalability: 34.1%
- Energy Efficiency/Thermal Management: 11.4%
- OS/Runtime: 34.1%
- Programming Models: 20.5%

as of Dec 2011
Upcoming MARC Events

• TACC-Intel Highly Parallel Computing Symposium, April 10\textsuperscript{th} – 11\textsuperscript{th} 2012, Austin, USA
• MARC China Symposium, May 17\textsuperscript{th}, Wuxi, China
• MARC Symposium July 19\textsuperscript{th} - 20\textsuperscript{th} 2012, Toulouse, France

http://communities.intel.com/community/community/marc
Case Study: Multicore in Automotive ECU Consolidation

Today:
- Multitude of different ECU systems (OS, architecture, network)

Near term:
- Consolidate compute functions
- Many VMs on one shared multicore processor

ECU: Embedded Control Unit
ECU Consolidation: A First Closer Look

New Requirements:
• Realtime
• Unsecure content
• Growing applications performance
• Heterogeneous
A new World of Software

New applications paradigms for mobile / embedded software
• “Apps” are it!
• Apps have shorter lifetimes than programs today
• Implies smaller budgets for development
• Requires productive programming abstractions

Mobile platforms evolve fast
• Quicker transition to multicore (staring at ~1GHz)
• Energy efficiency becomes even more important

Efficient parallel programming systems even more important for mobile / embedded environments
Case Study: Dynamic Web Delivered Content
River Trail

- Unlocks parallel hardware to HTML/JavaScript* applications
  - Multi-core, SSE/AVX
- Gently extends JavaScript with data-parallel constructs
  - Preserves safety and security of existing web development model
  - Interoperates with HTML5 and WebGL
- Leverages existing low-level software layers
  - Compiles JavaScript kernels to OpenCL*
- Targets application domains with abundant data parallelism
  - 3D gaming, physics simulations, photo and video editing, augmented reality
River Trail Status

- Research prototype implemented as Firefox extension
- Development moved to open source in September, 2011
  - [github.com/rivertrail/rivertrail/wiki](https://github.com/rivertrail/rivertrail/wiki)
  - Announced at IDF October 2011
- More than 3000 downloads and 300 followers since then

IDF particle demo
Physics simulation in Firefox
15x speedup
Summary

• On-die variations are real
• Application-level power management feasible and promising
• Research indicates significant increase in energy efficiency by NTV structures
• Parallel programming systems and OS research continues to be more relevant than ever
• Rising performance requirements of applications in mobile/embedded increasing need for parallel computing