Composite-ISA Cores: Enabling Multi-ISA Heterogeneity using a Single ISA

Ashish Venkat, Harsha Basavaraj, Dean Tullsen
The Landscape of Modern Computing

Software (rapidly evolving, more complex, and diverse)
The Landscape of Modern Computing

Hardware (traditionally homogeneous)  Software (rapidly evolving, more complex, and diverse)

Area: 263 mm²  Transistors: 731 M  Technology: 45 nm

Intel Nehalem

Software (rapidly evolving, more complex, and diverse)
The Landscape of Modern Computing

Area: 240 mm²
Transistors: 1.17 B
Technology: 32 nm

Hardware (traditionally homogeneous)  Software (rapidly evolving, more complex, and diverse)
As we continue to shrink transistors, power density will shoot up.

Power efficiency is key

Hardware (leakage-limited era)

Software (rapidly evolving, more complex, and diverse)
As we continue to shrink transistors, power density will shoot up.

**Cost of Generality**

Hardware (leakage-limited era)  
Software (rapidly evolving, more complex, and diverse)
The Landscape of Modern Computing

Hardware (more and more specialized)  Software (rapidly evolving, more complex, and diverse)
Hardware Specialization

– Domain-specific specialization:
  accelerate the performance of a particular class of computation
Hardware Specialization

– Domain-specific specialization:
  accelerate the performance of a particular class of computation

– Microarchitectural heterogeneity:
  use small power-efficient and large high performance cores that cater to diverse execution characteristics
Hardware Specialization

– Domain-specific specialization:
  accelerate the performance of a particular class of computation

Intel HD Graphics (CPU+GPU)  AMD APU (CPU+GPU)  Huawei Kirin (Neural Acceleration)  Google Cloud TPU (ML acceleration)  Microsoft Catapult (Bing search acceleration)

– Microarchitectural heterogeneity:
  use small power-efficient and large high performance cores that cater to diverse execution characteristics

Qcomm Snapdragon (A57 + A53)  Intel GoTM (Xeon + Atom)  Samsung Exynos 7 (A73 + A53)  Nvidia Tegra 3 (A-9 Variable SMP)  Apple A11 (Monsoon+Mistral)
Hardware Specialization vs Programmability

Traditional Hardware

Traditional Programming/Execution Model

- Algorithms
- High Level Language Code
- Assembly Code
- Instruction Set Architecture (Machine Language)
- CPU Microarchitecture
- Hardware Logic (Gates/Registers)
- Devices (Silicon)
Hardware Specialization vs Programmability

Specialized Hardware

"Big" Cores

"Little" Cores

Rapidly diverging Programming/Execution Model

1. Algorithms
2. High Level Language Code
3. Assembly Code
4. Instruction Set Architecture (Machine Language)
5. CPU Microarchitecture
6. Hardware Logic (Gates/Registers)
7. Devices (Silicon)

Specialized Hardware

- Cryptographic Acceleration
- Image Processing
- Multimedia Processing
- Digital Signal Processing
- GPU

"Big" Cores

"Little" Cores

Algorithms

High Level Language Code

Assembly Code

Instruction Set Architecture (Machine Language)

CPU Microarchitecture

Hardware Logic (Gates/Registers)

Devices (Silicon)
Hardware Specialization vs Programmability

How can we benefit from more specialization while preserving our traditional models of programming?
Evolution of Architectural Heterogeneity

Restricting cores to a single ISA eliminates an important dimension of heterogeneity

63% speedup
OR
69% energy savings with 3% performance loss*

Same ISA
Same Microarchitecture

Single-ISA heterogeneous multicore
- ARM Cortex A15
- ARM Cortex A9
- ARM Cortex A5
- ARM Cortex A12

Homogeneous multicore
- x86-64 core-i7
- x86-64 core-i7
- x86-64 core-i7
- x86-64 core-i7

Different ISAs
Different Microarchitectures

Heterogeneous-ISA multicore
- MIPS R10000
- Thumb Cortex-A7
- x86-64 ev6
- x86-64 core-i7

*Ashish Venkat, Matthew De Vuyst, Sriskanda Shamasunder, Kazem Taram, Dean M. Tullsen, ASPLOS’12, ISCA’14, ASPLOS ‘16, ISCA’18

Rakesh Kumar, Keith Farkas, Norm P. Jouppi, Partha Ranganathan, Dean M. Tullsen, MICRO’03
Our contention is . . .

• Restricting cores to a single ISA eliminates an important dimension of heterogeneity

• ISAs are designed for different goals:
  – High performance (e.g., x86-64)
  – Low power (e.g., ARM)
  – Reduced code size - Thumb ISA saves 30% in instruction fetch energy
  – Domain specific instructions
  – Compute bound vs memory bound
  – Instruction-level parallelism vs Data-level parallelism
Harnessing ISA Diversity (ISCA 2014)

- **Exploits ISA Affinity**
  - Application code regions have a natural ISA preference
- **Enables ISA-microarchitecture co-design**
  - Significant synergy in combining heterogeneous ISAs w/ heterogeneous hardware
- **21% Performance Improvement and 23% Energy Savings on average**
Why is cross-ISA process migration a hard problem?

• Different machine code
• Different data formats (types, widths, endianness, alignment)
• Different register sets
• Different stack frame layouts
Other deployment concerns

• Multi-vendor Licensing
• Legal Barriers
• Verification Costs
• Differences in ABI
• Heterogeneous Memory Consistency Models
This research . . .

Composite-ISA Cores: Enabling Multi-ISA Heterogeneity using a Single ISA

- Avoid multi-vendor licensing issues.
- Significantly reduces binary translation costs.
- Greater flexibility allows us to match/supersede the performance and efficiency advantages of multi-vendor ISA heterogeneity.
Outline

ISA Feature Set Derivation

Compiler and Runtime Strategy

Architectural Design Space Exploration
ISA Feature Set Derivation

• Start with a baseline (x86-like) superset ISA
• Customize along 5 different dimensions
  – Register Depth
  – Register Width
  – Addressing Mode Complexity
  – Predication
  – Data-Parallel Execution
• 26 different composite ISAs
Feature Diversity: Register Depth

- The number of programmable registers exposed by the ISA
- Performance/Power Implications:
  - Impacts a number of machine-specific and machine-independent compiler optimizations
  - Increasing the register depth from 16 to 32 results in 10.3% fewer loads, 3.7% fewer stores, 3.5% fewer integer arithmetic, and 2.7% fewer branches.
  - Greater register depth typically implies a larger register file
Feature Diversity: Register Width

- **Wider Types (64-bit)**
  - Allows access to larger virtual memory
  - May allow for better register usage via sub-register coalescing (improves performance)
  - Potentially larger cache working set (e.g., when pointers are members of a large structure)

- **Smaller Types (32-bit)**
  - Require emulation of wider types (hurts performance)
  - Enable compact register files (consumer 6.4% less power than a 64-bit organization)
Feature Diversity: Addressing Mode Complexity

- Reduced set of addressing modes (microx86 – a RISC version of x86)
  - 1:1 macro-op to micro-op encoding (simpler decoders)
  - 9.8% reduction in peak power and 15.1% reduction in area
- Complete set of addressing modes (CISC x86)
  - Compact code generation (fewer instruction cache accesses)
  - Multiple bandwidth optimizations (micro-op cache, micro-op fusion, loop buffer, etc.)
Feature Diversity: Predication

- **Partial Predication**
  - x86 already implements partial predication via CMOV instructions (predicated on condition codes)

- **Full Predication**
  - Any instruction can be predicated on any architectural register
  - Enables more aggressive if-conversion (6.5% fewer branches and 0.6% more integer arithmetic)
  - Allows the designer to choose simpler branch predictors in tightly power-constrained environments
Feature Diversity: Data-Parallel Execution

- microx86 cores do not implement SIMD instructions
  - Saves 7.4% in peak power and 17.3% in area
  - Execute a pre-compiled scalarized version when available
  - Migrate to an x86 core that implements SIMD during vector phases
ISA Encoding

- Standard ISA extension methodology
- Two new prefix bytes (for predication and register depth) that leverage unimplemented opcodes
- Compiler/Assembler is code density-aware
Decoder Design

- Impact on the x86 front end
  - More Prefix Decoding Logic
  - Wider queues and buffers
  - Wider Micro-Op Cache
  - Mix of simple/complex macro-op decoders (microx86 vs x86)

- Decoder Power and Area estimates with our customizations
  - Pre-decoder (Full RTL Design): 0.87% increase in peak power and 0.65% in area
  - Smallest ISA (microx86-32) consumes 0.66% less peak power and 1.12% less area than x86-64
  - Largest ISA (superx86) consumes 0.3% more peak power and 0.46% more area than x86-64
Outline

ISA Feature Set Exploration

Compiler and Runtime Strategy

Architectural Design Space Exploration
Compiler Strategy

Vectorization

Type Legalization

Instruction Selection

Register Allocation

Simple, Triangle, Diamond If-Conversion

Machine Code Generation (LLVM-MC)

Composite-ISA Features:
- Data Parallelism: {SIMD, no SIMD}
- Register Width: {32-bit, 64-bit}
- Addressing Mode Options: {x86, microx86}
- Register Depth: {8, 16, 32, 64 registers}
- Predication: {partial (CMOV), full predication}

Composite-ISA Encoding Prefixes and Options
Feature Affinity
Migration Strategy

Feature Upgrade
- Common Case (91.5% of migrations)
- No binary translation required

Feature Downgrade
- Minimal binary translation required
- Average Performance Impact: 0.46%
Outline

ISA Feature Set Exploration

Compiler and Runtime Strategy

Architectural Design Space Exploration
Design Space Exploration

Micro-architectural parameters
+ ISA parameters
+ Budget constraints

DSE

Homogeneous multicore
- x86-64
- x86-64
- x86-64
- x86-64

Single-ISA heterogeneous multicore
- x86-64
- x86-64
- x86-64
- x86-64

Multi-vendor heterogeneous-ISA multicore
- Alpha
- Thumb
- Alpha
- x86-64

Composite-ISA heterogeneous multicore
- x86 custom-1
- x86 custom-2
- x86 custom-3
- x86 custom-4
## Design Space Exploration

Choice of micro-architectural parameters

<table>
<thead>
<tr>
<th>Design Parameter</th>
<th>Design Choice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution Semantics</td>
<td>In-order, Out-of-order</td>
</tr>
<tr>
<td>Issue Width</td>
<td>1, 2, 4</td>
</tr>
<tr>
<td>Branch Predictor</td>
<td>2-level local, gshare, tournament</td>
</tr>
<tr>
<td>Instruction Queue Size</td>
<td>32, 64 entries</td>
</tr>
<tr>
<td>Reorder Buffer Size</td>
<td>64, 128 entries</td>
</tr>
<tr>
<td>Physical Register File Configurations</td>
<td>(96 INT, 64 FP/SIMD), (64 INT, 96 FP/SIMD)</td>
</tr>
<tr>
<td>Integer ALUs</td>
<td>1, 3, 6</td>
</tr>
<tr>
<td>Integer Multiply/Divide Units</td>
<td>1, 2</td>
</tr>
<tr>
<td>FP/SIMD ALUs</td>
<td>1, 2, 4</td>
</tr>
<tr>
<td>FP Multiply/Divide Units</td>
<td>1, 2</td>
</tr>
<tr>
<td>Load/Store Queue</td>
<td>16, 32 entries</td>
</tr>
<tr>
<td>Instruction Cache</td>
<td>32KB 4-way, 64KB 4-way</td>
</tr>
<tr>
<td>Private Data Cache</td>
<td>32KB 4-way, 64KB 8-way</td>
</tr>
<tr>
<td>Shared Last Level (L2) cache</td>
<td>4-banked 4MB 4-way, 4-banked 8MB 8-way</td>
</tr>
</tbody>
</table>

4680 distinct single core design points and a 102.5 trillion 4-core configurations

49733 core hours on the 2 petaflop Comet Cluster at the San Diego Supercomputing Center
We generally gain more from ISA feature diversity than hardware heterogeneity
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

- **Feature affinity**: different code regions have a natural affinity for one feature or another

- **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

Both designs are constrained at a peak power budget of 40W
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

- **Feature affinity**: different code regions have a natural affinity for one feature or another
- **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

Both designs are constrained at a peak power budget of 40W
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

• **Feature affinity**: different code regions have a natural affinity for one feature or another

• **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

Both designs are constrained at a peak power budget of 40W
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

• **Feature affinity**: different code regions have a natural affinity for one feature or another

• **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

Both designs are constrained at a peak power budget of 40W
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

- **Feature affinity**: different code regions have a natural affinity for one feature or another

- **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

Both designs are constrained at a peak power budget of 40W
Design Space Exploration
Multi-programmed workload throughput

Benefits of composite-ISA cores come from:

- **Feature affinity**: different code regions have a natural affinity for one feature or another
- **ISA-microarchitecture co-design**: squeeze in more powerful cores into the same budget

![Diagram of best single-ISA heterogeneous CMP and best composite-ISA CMP](image)

Both designs are constrained at a peak power budget of 40W
Feature Sensitivity

The best performing designs typically employ most features.
Processor Transistor Investment

Area (Normalized to Full Feature Diversity)

Feature Constraint
- Fetch
- Decode
- Branch Predictor
- Scheduler
- Register File
- Functional Units

Register Depth: 8, 16, 32, 64
Register Width: 32, 64
Instruction Complexity: microx86, x86
Predication: Partial, Full
Full Feature Diversity
Multi-programmed Workload Efficiency

- 31% energy savings and 35% reduction in EDP at ZERO performance loss
- We gain performance and save energy simultaneously

![Normalized EDP vs Peak Power Budget](chart)
In summary . . .

**Composite-ISA Cores: Enabling Multi-ISA Heterogeneity using a Single ISA**

- Effectively avoids multi-vendor licensing issues, verification, binary translation costs
- Gives the processor designer and the compiler a rich set of ISA feature options
- Greater flexibility allows us to match/supersede the performance and efficiency advantages of multi-vendor ISA heterogeneity.
Composite-ISA Cores: Enabling Multi-ISA Heterogeneity using a Single ISA

Ashish Venkat, Harsha Basavaraj, Dean Tullsen