

Value Compression for Efficient Computation

Ramon Canal, Antonio González and James E. Smith

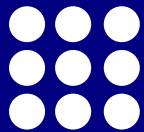
Universitat Politècnica de Catalunya - Barcelona

University of Wisconsin - Madison



Outline

- 1. Introduction**
- 2. Value Compression**
- 3. Experimental Framework**
- 4. Results**
- 5. Summary & Conclusions**



UPC

Introduction

Increasing miniaturization

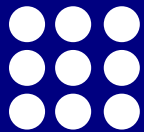
- Portable handhelds (PDAs, notebooks, cell phones, etc.)

Increasing computation requirements

- Multimedia processing (audio, video)
 - Speech-recognition
 - Audio and video encoding/decoding
- Wireless communication

Increasing demand for

- Longer battery life
- Smaller heat dissipation

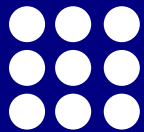
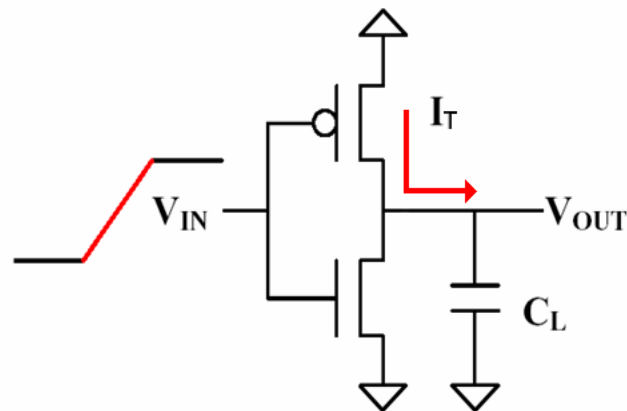


UPC

Introduction

Design for low power

- Becoming a critical design constraint
- Types of energy consumption
 - Dynamic (predominant source for current technology)
 - Charging/Discharging capacitors



UPC

Introduction

Design goal:

POWER EFFICIENCY

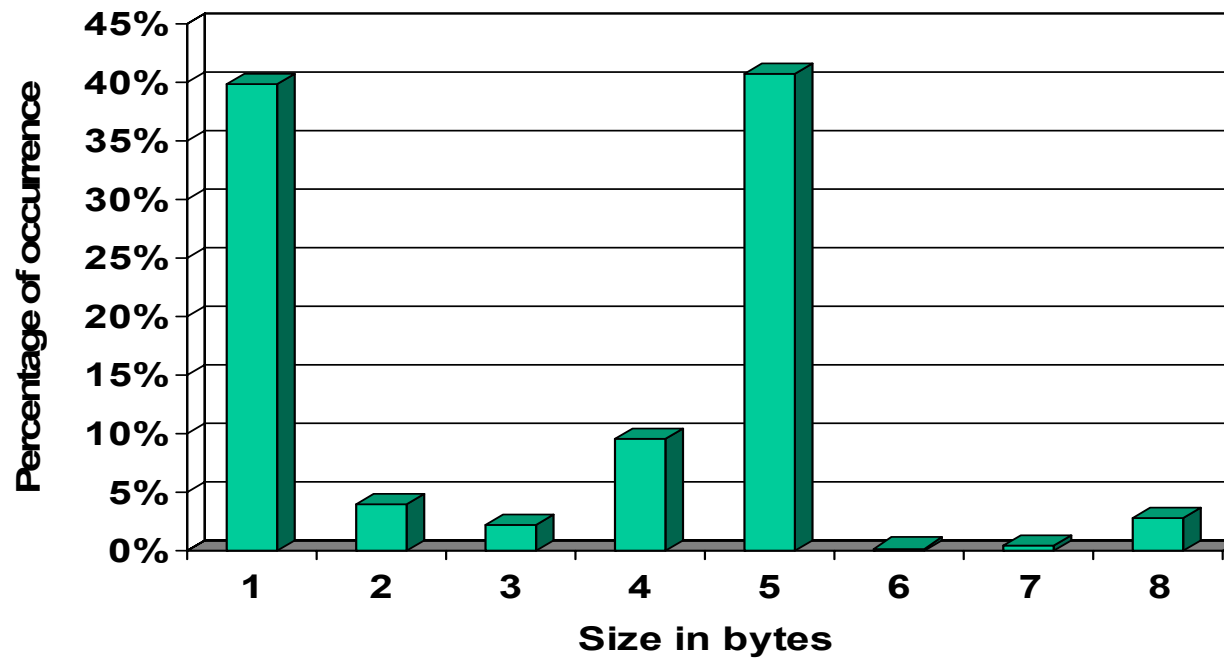
Metrics used:

- Energy consumption
- Energy-delay²



Value Compression

Data size distribution for the Spec Integer 2000 benchmarks



Value Compression

Analysis of value compression mechanisms for 64-bit architectures

- Size compression
 - Leading sign extension bytes
- Significance compression
 - Sign extension bytes
- Zero compression¹
 - Zero bytes



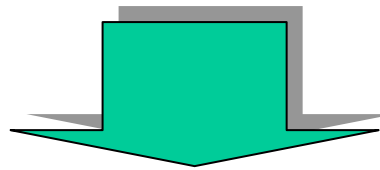
¹ L. Villa, M. Zhang, and K. Asanovic, "Dynamic Zero Compression for Cache Energy Reduction", in Proc. of the 33rd International Symposium on Microarchitecture, Dec.2000.

Size Compression

32-bit Value

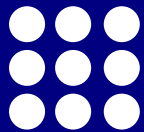
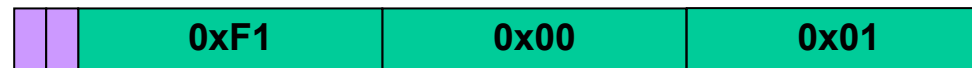


Size compression approach



Sign ext'n byte

Sign extension bit



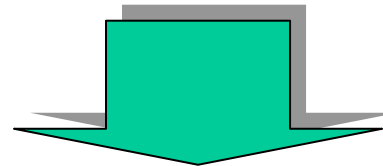
UPC

Significance Compression

32-bit value



Significance compression approach



Sign ext'n byte



Sign extension bit

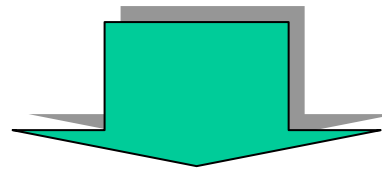


Zero Compression¹

32-bit Value



Size compression approach



zero byte

Zero extension bit

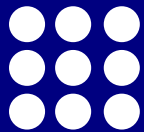


¹ L. Villa, M. Zhang, and K. Asanovic, "Dynamic Zero Compression for Cache Energy Reduction", in Proc. of the 33rd International Symposium on Microarchitecture, Dec.2000.



Components exposed

- Front-End
 - Branch-prediction (BTB)
 - Instruction-cache (compressed instructions and tags)
- Back-End
 - Register File (Holds compressed data)
 - Rename Buffers (Holds compressed data)
 - Functional Units (Operate on compressed data)
 - Data-cache (keeps compressed data and tags)



UPC

Experimental Framework

Simulator

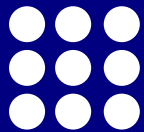
- based on Wattch
- + data width operand gating

Benchmarks

- Spec Integer 2000 benchmark suite

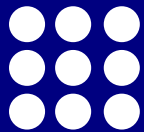
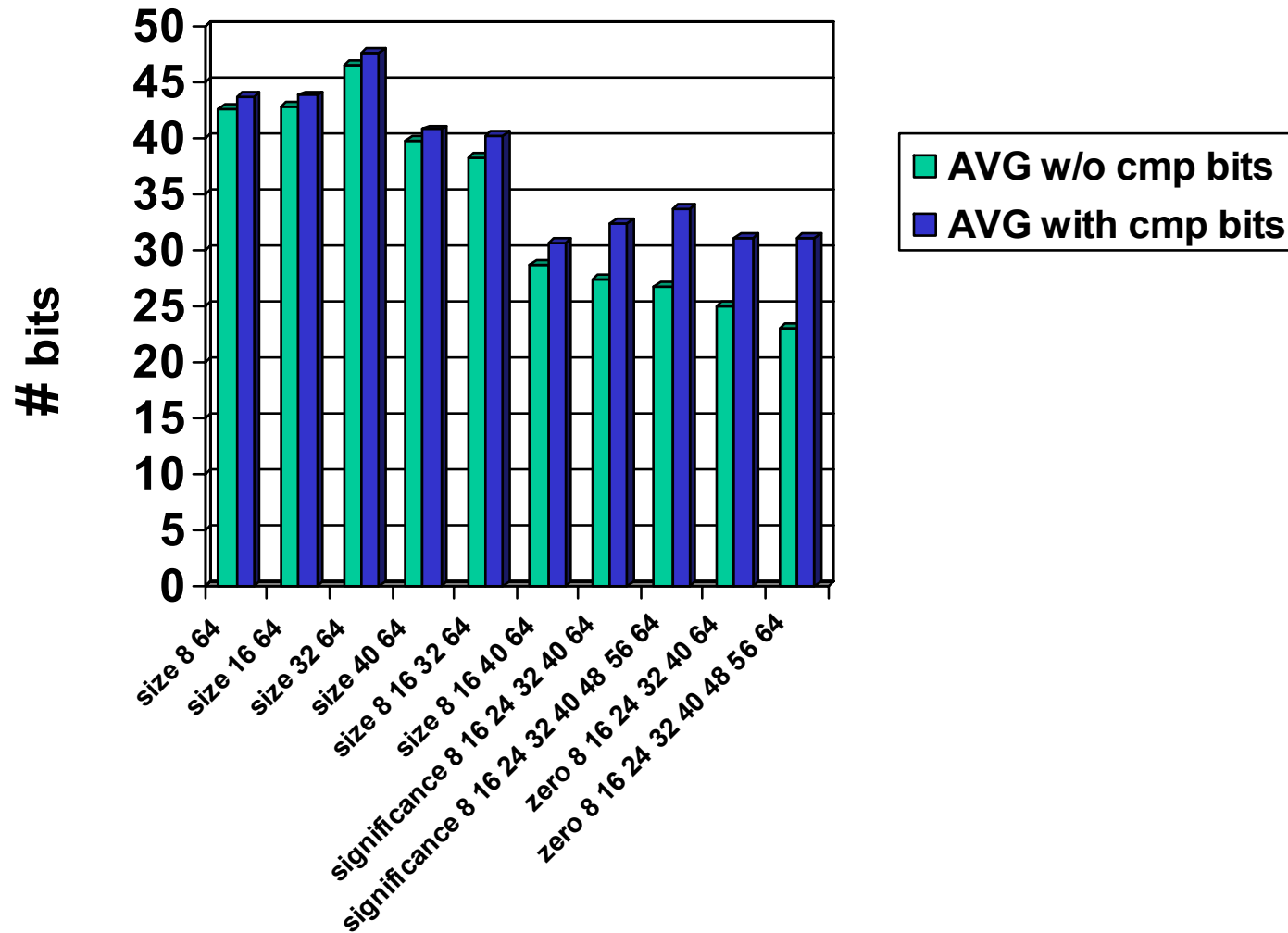
Configuration

- 4-way issue
- ROB of 64 entries



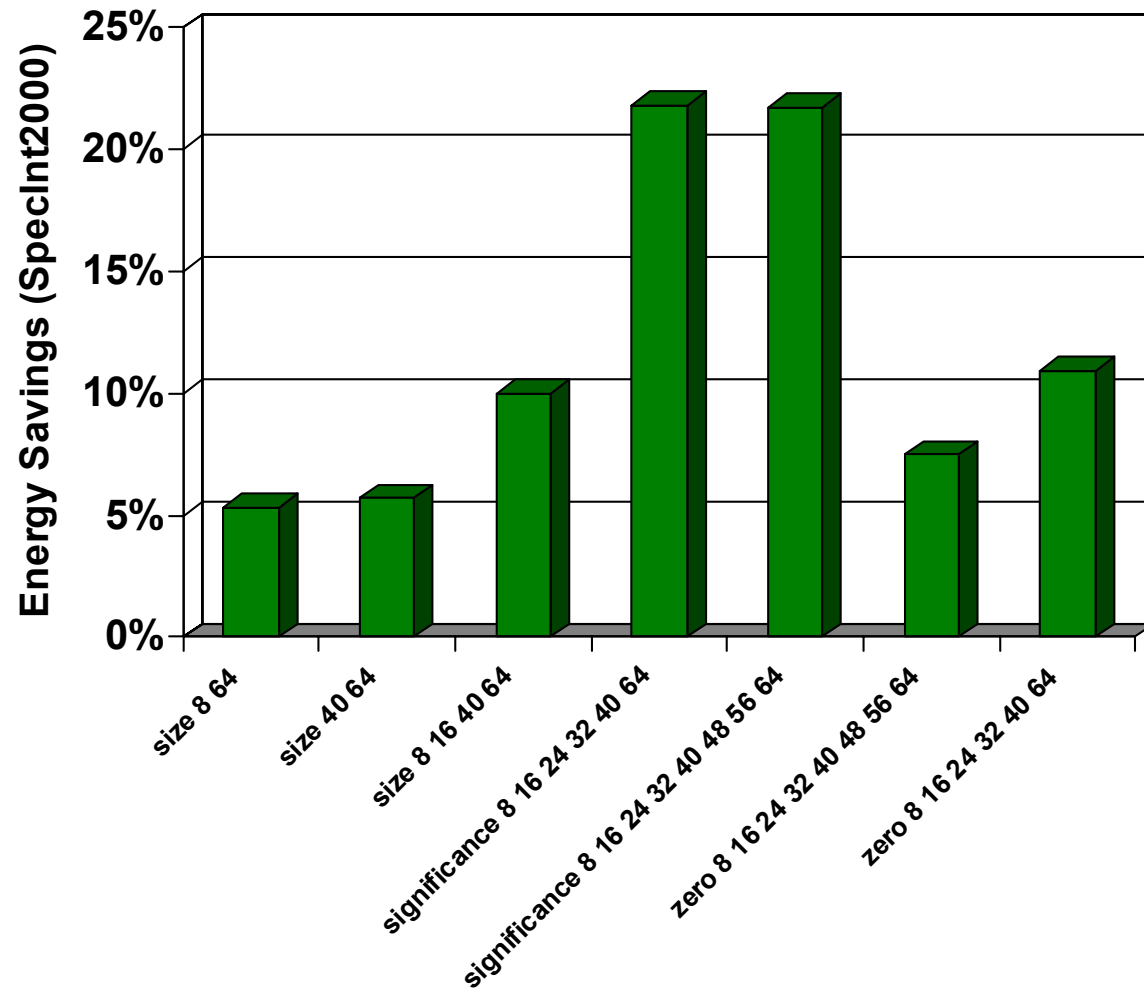
UPC

Average Data Size

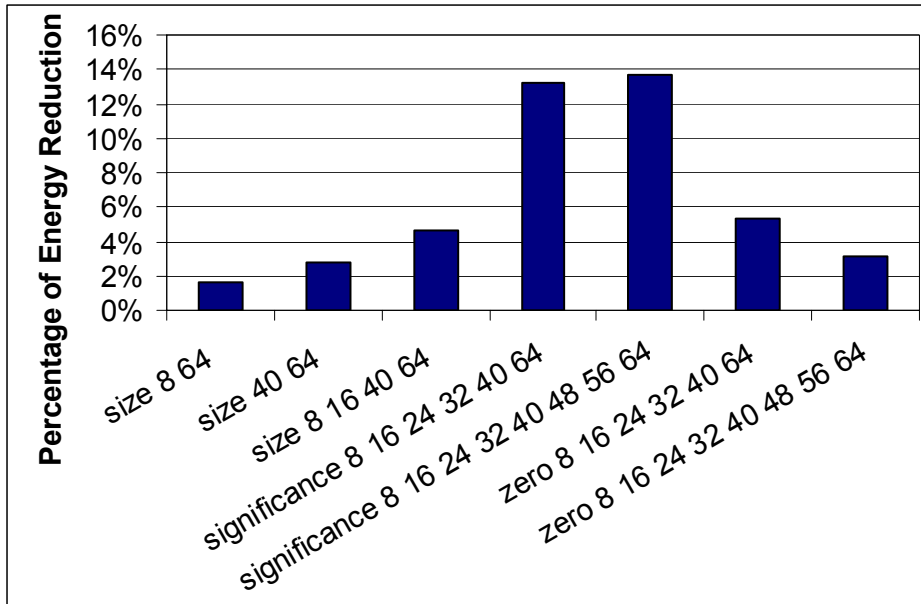


UPC

Energy Savings

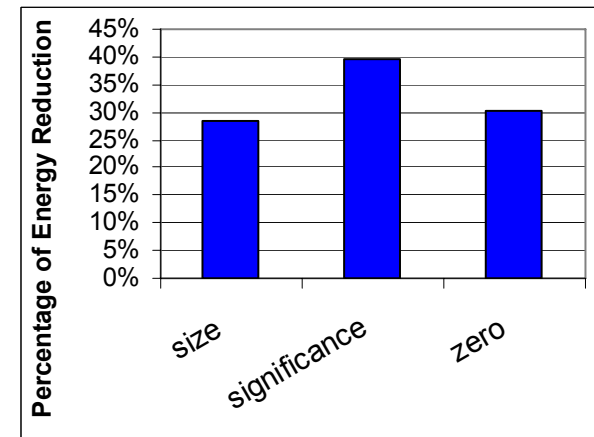


Energy Savings – detailed –



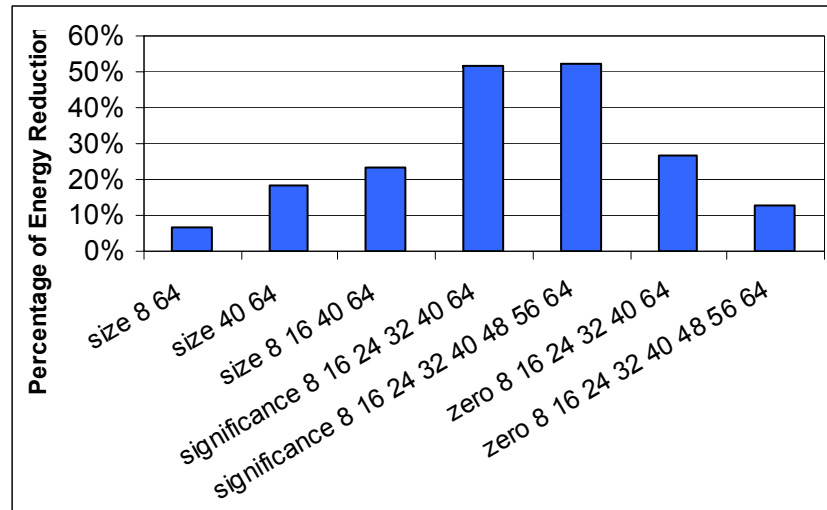
← D-cache (L1)

I-cache ↓

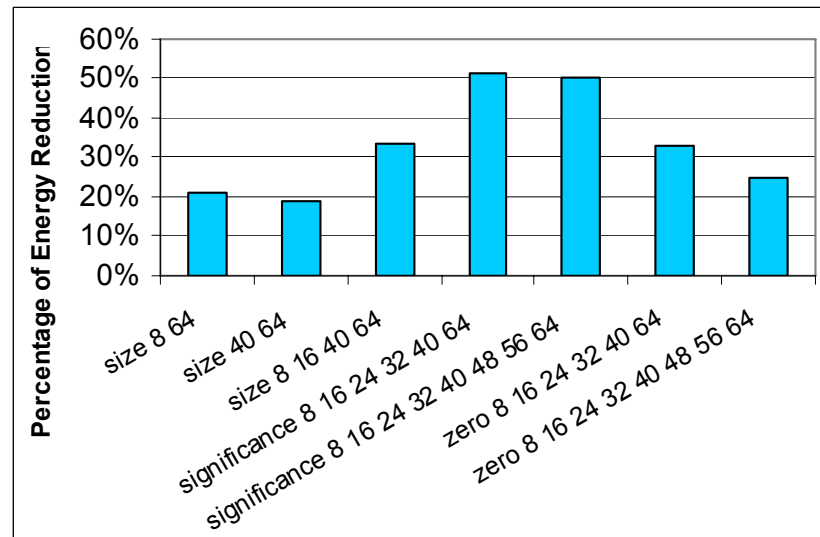


Energy Savings – detailed –

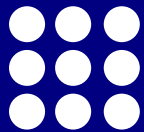
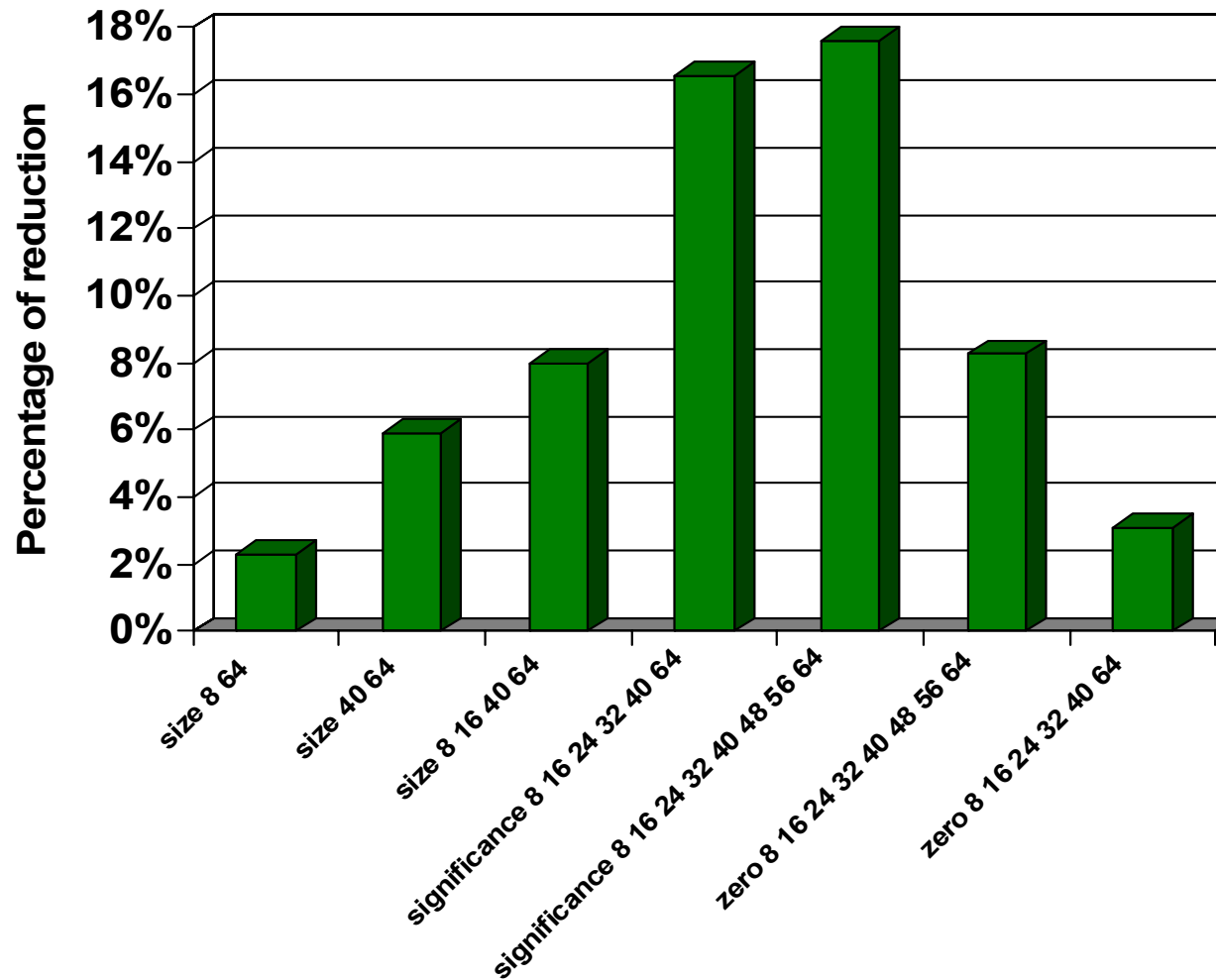
ALU →



Register File →



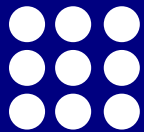
Peak Power



UPC

Summary & Conclusions

- Value compression is an effective way of reducing dynamic power requirements
- Value compression enables simpler (or smaller) components –static power.
- Value compression reduces peak power – making it a good candidate for temperature-aware computing.



UPC

Q & A

“Value Compression for Efficient Computation”

Ramon Canal, Antonio González, James E. Smith

Universitat Politècnica de Catalunya - Barcelona

University of Wisconsin - Madison

