Dynamic Zero Compression for Cache Energy Reduction

Luis Villa
Michael Zhang
Krste Asanovic
{luisv|rzhang|krste}@lcs.mit.edu

Conventional Cache Structure

- Energy Dissipation
  - Bitlines (~75%)
  - Decoders
  - I/O Drivers
  - Wordlines
Existing Energy Reduction Techniques

- Sub-banking
- Hierarchical Bitlines
- Low-swing Bitlines
  - Only for reads, writes performed full swing.
- Wordline Gating

Asymmetry of Bits in Cache

- >70% of the bits in D-cache accesses are “0”s
  - Measured from SPECint95 and MediaBench
  - Examples: small values, data types

- Related work with single-ended bitlines
  - [Tseng and Asanovic ‘00] --- Used in register file design with single-ended bitlines.
  - [Chang et. al. ’99] --- Used in ROM and small RAM with single-ended bitlines.

- Differential bitlines preferred in large SRAM designs.
  - Better Noise Immunity
  - Faster Sensing
**Dynamic Zero Compression**

- **Zero Indicator Bit**
  - One bit per grouping of bits
  - Set if bits are zeros
  - Controls wordline gating

---

**Data Cache Bitline Swing Reduction**

*Calculation includes the bitline swings introduced by ZIB*
Hardware Modifications

- Zero Indicator Bit
- Wordline Gating Circuitry
- Sense Amplifier
- CPU Store Driver
- Cache Output Driver

ZIB and Wordline Gating Circuitry

Small Delay Overhead
Sense Amplifier Modification

- **Zero-valued data:**
  - Not driven onto bus
  - Not in critical path
  - ZIB read w/o delay

CPU Store and Cache Output Drivers

Reduce Data Bus Energy Dissipation
Area Overhead

- **Area Overhead: 9%**
  - Zero-Indicator-Bits
  - Sense Amplifiers
  - WLG Circuitry
  - I/O Circuitry

*Byte slice of the sub-bank (Data,ZIB,WLG)*

Delay Overhead

- **No delay overhead for writes**
  - Zero check performed in parallel with tag check

- **2 F04 gate-delays for reads**
  - A pessimistic 7% worst case delay
Data Cache Energy Savings

- Savings obtained for a low-power cache with sub-banking, wordline gating, and low-swing bitlines

Bits Distribution for Instruction Cache

- Zeros are not as prevalent in I-Cache.
- Use a recoding scheme to increase the zero-byte in I-cache.
- [Panich '99] --- IWLG technique that compacts the instructions.
  - Use two-address form when src reg = dest reg
  - Shorter immediates
  - Three different instruction length: short, medium, long
  - Gate the unused portion of the instruction to avoid bitline swing
  - Faster read-out for top two bytes (opcode, reg. acc., inter-locks)
Adopting IWLG technique for Dynamic Zero Compression

- Small modification on instruction format
  - Use 8-8-8-8 instead of 16-7-9
- Upper two byte are zero-detected
- Lower two bytes are usage-detected
- Able to eliminate bitline swings of zero-valued bytes in 2 upper bytes
  - *Example: Opcode 000000*
- Slower than IWLG due to wordline gating in the critical path

Instruction Cache Bit Swing Reduction
**Conclusion**

- **A novel hardware technique to reduce cache energy by eliminating the access of zero bytes.**
  - Small area and delay overhead
    - Area: 9%, Delay: 2 F04 gate-delays
  - Average energy saving: D-Cache: 26%, I-Cache: 18%
    - Processor wide: ~10% for typical embedded processors
  - Completely orthogonal to existing energy reduction techniques

- **Dynamic Zero Compression is applicable to**
  - Second level caches
  - DRAM
  - Datapath [Canal et. al. Micro-33]
Thank You!

http://www.caq.lcs.mit.edu/scale/