ECE 571 – Advanced Microprocessor-Based Design Lecture 17

Vince Weaver

http://www.eece.maine.edu/∼vweaver

vincent.weaver@maine.edu

26 March 2013

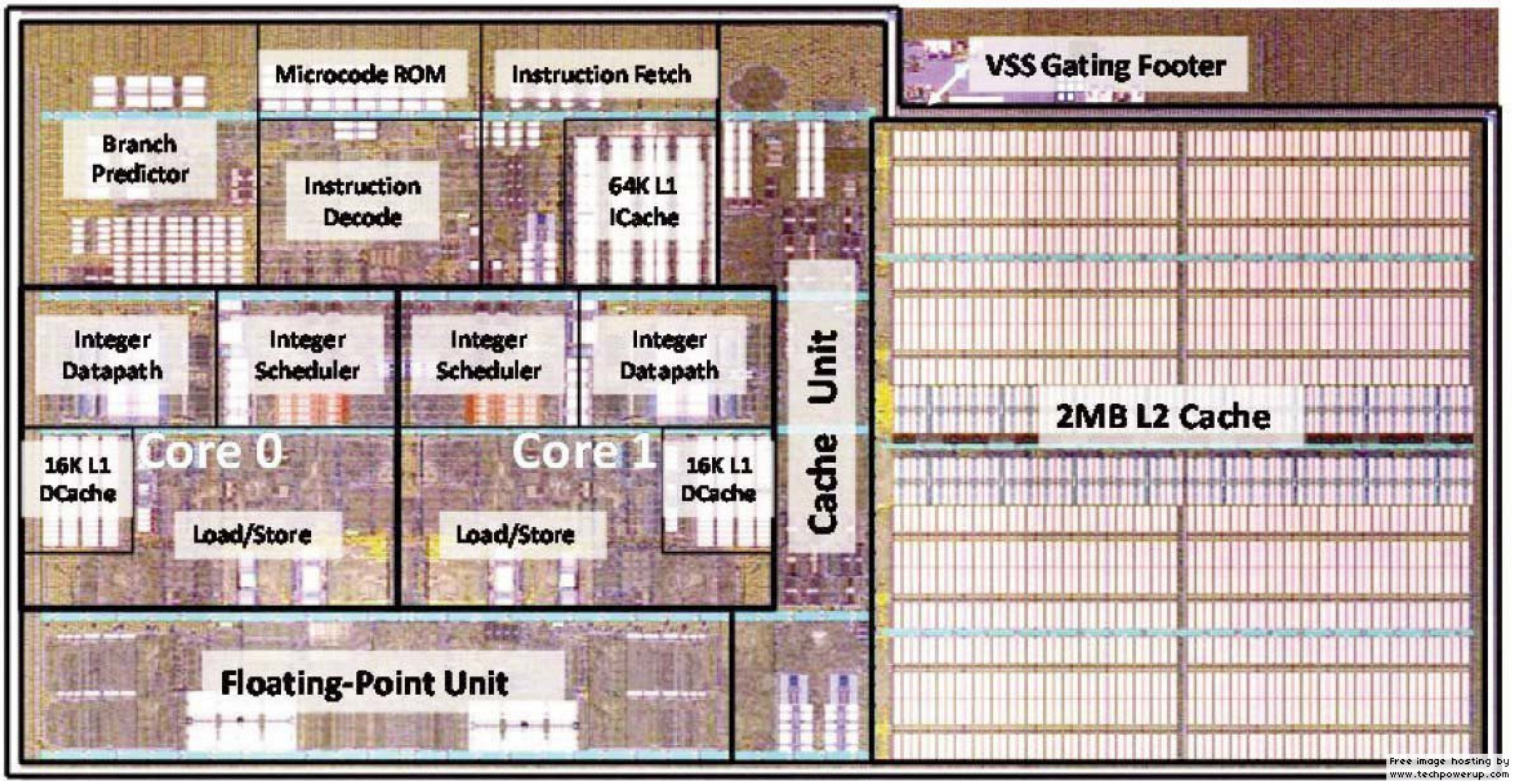
Project Reminder

- Topic Selection by today
- Once you have selected a topic, start making a list of what machines, features, and benchmarks you need to ensure you have enough time to gather results.
  Some test setups will take longer to set up than others.
- Also, start looking at related work.
- The first status update will be part of Homework 4

CPU Power and Energy

- Became a trendy thing to research in 1999-2002 timeframe.
- Before that usually concern was with performance.
- These days energy results are often reported as a core part of any architectural proposal, not as a separate issue.
- The results discussed here are academic and may or may not be implemented in actual chips.

AMD Bulldozer Die Shot



Note which structures are big, using static power.

CPU Energy Breakdown

From Fan, Tang, Huan, Gao (ISLPED'05), Chinese Godson MIPS CPU

- Cache 36%
- TLB 13%
- FALU 10%
- ROQueue 7%
- FMUL 6%

- Float reg 5%
- Gen reg 5%
- MUL 2%
- MCUControl 2%
- ALU 1%
- Other 13%

Thermal Concerns Too

Power density exceed hot plate, approaching rocket nozzle

Methodologies Used in These Papers

It varies, but many of these are from simulations (sometimes validated). Anything from SPICE to “cycle-accurate” simulators.

Clock Generation

- Driving high-frequency load against capacitance, trying to keep whole chip in sync.
- Extreme Case: Alpha 21264 H-tree, 32% of power?
- Half-frequency clocks (on both edge, so clock run half as fast) (Mudge 2001)
- Asynchronous
- Locally Asynchronous (Divide to multiple clock domains)

Cache Power and Energy

Large area, low-hanging fruit

Decay Caches

- Kaxiras, Ho, Martinosi (ISCA 2001)
- Turn off cache lines not being used to reduce leakage
- DRAM cache with no refresh
- Decayed values can be re-fetched from memory. Tradeoff.

Drowsy Caches

- Flautner, Kim, Martin, Blaauw, Mudge. ISCA 2002.
- Move cold cache lines into "drowsy" mode.
  Lower power enough to hold state, not enough to lose contents. Reduce leakage. Better than decay as not lose data.

Adaptive Caches

- Albonesi (Micro 1999). Manually turn off ways in cache with an instruction.
- Size the caches

Cache Compression

- Dynamic zero compression for cache energy reduction (L Villa, M Zhang, K Asanović. Micro 2001).
- Cache Compression ("sign compression" – top bits) Energy savings 20% (simulated) (Kim, Austin, Mudge WMPI 2002)

Banking and Filtering

- Filter cache, banking (only have half of cache active) (Mudge 2001)
- Slowing Down Cache Hits, Banked Data Cache. (Huang, Renau, Yoo, and Torrellas. Micro 2000.)
- Vertical Banking, Horizontal Banking (Su and Despain, ISLPED 1995).

Code Scheduling

- Can Schedule code for lower power.

- Better cache rates lower power. performance/power can go hand in hand. (Kandemir, Vijaykrishnan, Irwin)

Branch Predictors

- Parikh, Skadron, Zhang, Barcella, Stan

- 4 concerns:

  1. Accuracy. Not affect power, but performance
  2. Configuration (may affect power)
  3. Number of lookups
  4. Number of updates

- Tradeoff power vs time.

- brpred can be size of small cache, 10% of power
- Can use banking to mitigate

Branch Predictors

- can watch icache, not activate predictor if nobranches
- Pipeline gating, keep track of each predicted branch confidence. If confidence hits certain threshold, stop speculating. Show this may or may not be good.
- Integer code, large predictors good
- FP, tight loops, predictors not as important.

Branch Predictor Evaluation

- (Strasser, 1999). Simulation, small branch predictor can help energy.

- (Co, Weikle, Skadron) Formula for break even point. Leakage matters, what brpred hides is stall cycles.

- SEPAS: A Highly Accurate Energy-Efficient Branch Predictor (Baniasadi, Moshovos. ISLPED 2004). Once a branch prediction reaches steady state (unlikely to change) stop accessing/updating predictor, saving energy.

- Low Power/Area Branch Prediction Using Complementary Branch Predictors (Sendag, Yi, Chuang, Lija. IPDPS 2008)
  Complementary Branch Predictor to handle the tough cases.

TLB Energy

TLB Optimization – Assume in Same Page

- (Kadayif, Sivasubramaniam, Kandemir, Kandiraju, Chen. TODAES 2005).
  Don't access TLB if not necessary. Compare to last access (assume stay in same page) Circuit improvements

- (Kadayif,Sivasubramaniam, Kandemir, Kandiraju, Chen. Micro 2002)
  Cache page value.

TLB Optimization – Use Virtual Caches

- (Ekman and Stenström, ISLPED 2002) Use virt address cache. Less TLB energy, more snoop energy. TLB keeps track of shared pages.

TLB Optimization – Reconfiguring

- (Basu, Hill, Swift. ISCA 2012) Have the OS select if memory region physical or virtual cached.
- (Delaluz, Kandemir, Sivasubramaniam, Irwin, Vijaykrishnan. ICCD 2013) Reducing dTLB Energy Through Dynamic Resizing.
  Size TLB as needed, shutting off banks. Easier if fully-associative.

TLB Optimization – Memory Placement

- (Jeyapaul, Marathe, Shrivastava, VLSI'09) Try to keep as much in one page as possible via compiler.

- (Lee, Ballapuram. ISLPED'03) Split memory regions by region (text/data/heap). Better TLB performance, better energy.

Bus Protocols

- Bus Protocols
- Cache-Coherence Protocols

Busses

- Grey Code, only one bit change when incrementing. Lower energy on busses? (Su and Despain, ISLPED 1995).

Prefetching

- Prefetching does not get looked at as closely. Various studies show it can be a win energy wise, but it is a close thing.
- (Guo, Chheda, Koren, Krishna, Moritz. PACS'04) HW Prefetch increase power 30%; have compiler help augment with hints, filters.
- (Tang, Liu, Gu, Liu, Gaudiot. Computer Architecture Letters, 2011).

Mixed results.