PowerPC based micro-architectures
Godfrey van der Linden
Presentation for COMP9244 — Software view of Processor Architectures
2006-05-25

History
- 1985 – IBM started on AMERICA
- 1986 – Development of RS/6000
- 1990 – First RS/6000 released, with Power
- early ’90s – AIM alliance, PowerPC 64 ISA defined
- 1993 – POWER3 implements full PPC–64
- 2001 – POWER4 released
- 2002 – PowerPC 970 “G5”
- 2005 – POWER5 released, Bluegene topped TOP500 list, Apple goes Intel

Introduction
- A brief history
- The PowerPC ISA
- Big Iron POWER: POWER4, PPC970, POWER5, POWER6
- Embedded PowerPC: 440, Bluegene
- Programming Issues

PowerPC ISA
- Second Generation RISC architecture
- Some atypical multi-cycle instructions
  - lmw, stmw
  - Load & store with register update
- Cool instructions
  - Rotates with mask, C bitfields
  - cntlz, Count leading zeros, i.e. log2
  - Complemented Logicals, x <op> ~y
- D-cache prefetch instructions
**PowerPC ISA Continued**

- Synchronisation
- isync, eieio, lwsync, sync
- More Load/Store variations
- Unaligned bytes to/from registers
- Swap bytes to/from memory
- Branch unit registers, lr & ctr
- used for indirect branches too
- PC is not directly accessible from ISA

• isync: Complete all previous instructions, Complete instruction cache invalidations, load barrier
eieio: Enforce In-Order Execution of I/O, for device memory guarantee all previous loads and
lwsync: Lightweight synchronisation, write barrier for cached system memory
sync: Heavyweight synchronisation, write barrier against Memory Mapped Device registers.

**POWER4**

- 2-way multi-core Chips
- High-frequency 16-stage pipeline
- 8 issues per cycle
- 5 commits per cycle
- > 200 inst. window
- Massive caches - 128b
  - lcache 64K direct
  - D-cache 32K 2-way
  - L2 shared unified 8-way 1.41Mb
  - L3 shared on-chip directory 8-way 32Mb
  - 512b line

**POWER5**

- Enhancement of POWER4
- 2-way multi-core with 2-way prioritised SMT
- Unified shared 1.875-MB L2 cache, still 3 unit L2 units
- Unified shared 32Mb L3 Cache Directory with dedicated memory ports
- L3 “victim cache” for parallel L2
- On-chip memory controller

The manufacturing process is more aggressive on the PowerPC, the chips will be less ‘reliable’

than the POWER series.
POWER5 Memory System

- Embedded processor core
- 2-way superscalar, dual issue - 7 stage
- 32–Bit PowerPC compliant
- Modular design
- 2.5mW/MHz power consumption, 555MHz nominal @ 1.4W, 0-400MHz “slow silicon”
- Caches 0-64KB, 32-way to 128-way associative
- 4.0 mm² for CPU only

POWER6

- Probably mid-2007 delivery
- 4-5 GHz clock rate, 6GHz in lab
- Will support VMX(Altivec)
- That’s about it. IBM is playing it very close to their chests.

PowerPC 440

- Embedded processor core
- 2-way superscalar, dual issue - 7 stage
- 32–Bit PowerPC compliant
- Modular design
- 2.5mW/MHz power consumption, 555MHz nominal @ 1.4W, 0-400MHz “slow silicon”
- Caches 0-64KB, 32-way to 128-way associative
- 4.0 mm² for CPU only

Bluegene

- Cluster computer, max 64K 2-way nodes
- Based on 2.700MHz PPC 440 cores
- Dual–processor “System on a chip”
- Integrated 4Mb L3 Cache
- On-chip DDR Memory Controller
- Built-in Gigabit DMA Ethernet controller
- With 128K processors 367,000Gflop
- Target 27KW per rack

Selectable cache sizes, timers and trace facilities

The 1,034-bit-wide data port of the embedded DRAM provides 22.4 GB/s bandwidth to serve demands of the two processor cores. Seems to be current maximum for air cooled processors. Didn’t research the Cell processor in detail.
Programming Issues

- Only POWER4 derivative
- Shared Storage
- Cache and OS implementation
- Function indirection
- Instruction Groups
- PowerPC performance tools

Cache and OS implementation

- Prefetch the data and instruction caches
- Structuring the data structures appropriately should reduce cache misses
- Each cache miss to main memory takes hundreds of cycles and should be avoided.
- Engineering trade-off, must avoid cache pollution of unnecessary prefetching.

Just hit some high points

Shared Storage

- The PowerPC ISA only specifies a weakly consistent storage model.
- Can't rely on the order of stores
- This is a big issue with the massively superscalar P4 design, with write buffering
- Must use appropriate syncing instructions for cross-processors coherence
- Device memory coherence is particularly slow

Function indirection

- P4 has dedicated Fetch/Branch unit registers: lr & ctr. Used for indirection
- Unfortunately the target of branches are always predicted!
- Cost of missed branch prediction is 12 cycles, 60 instruction commits
- May be better to encourage compiler to not predict indirections

sync is required for device memory, processor blocks until the memory controller acknowledges the sync request
Instruction Groups

- P4 issues iops in groups of 5 iop slots
- The last slot must be a branch or nop
- Complicated rules for group construction
- gcc does not model this grouping well, it is better to use IBM’s compiler if possible
- Hand-coding hot paths will be a win

PowerPC performance tools

- Apple has some excellent, free, PowerPC performance tools, CHUD
- SHARK can associate performance problems to particular lines of code. Has in-OS hooks
- amber is the G5 cycle accurate simulator

Conclusion

- The PowerPC ISA has turned out to be very flexible
- It has implementations from 1.4W up to 100s of Watts.
- Aside from weird function pointer caching it is remarkably good for C
- I wonder if how many pipeline stages POWER6 will have for a 5GHz clock?

iops are internal instructions mostly 1:1 for RISC operations but some instructions have more than one iop, like load or load/store with update.

Which means that it won’t work for instrumenting L4 unfortunately.