IBM POWER7 (and Beyond)

Lai Wei

Updates by John Mellor-Crummey

COMP 522 January 29, 2019
History of IBM POWER series

IBM POWER4, 2001
180 nm, 412 mm²
2 core @ 1~1.3 GHz

The first dual core chip

IBM POWER5, 2004
130 nm, 389 mm²
2 core @ 1.5~1.9 GHz,
2-way SMT

The first SMT chip

IBM POWER6, 2007
65 nm, 341 mm²
2 core @ 3.6~5 GHz,
2-way SMT

Ultra high frequency
POWER7 Multicore Server Processor
POWER7 (2010)

- 45 nm, 567 mm²
- 8 cores per chip
- 4-way SMT per core

Each core has:
  - 32KB L1 I/D cache
  - 256 KB L2 cache
  - 4MB Local L3 region

- 32 MB shared L3
- Multi-socket support

Figure credit: IBM POWER7 multicore server processor, Sinharoy, B. et al. IBM Journal of Research and Development 55(3), May-June 2011, 1:1-1:29.
Overview

- SMT implementation
- Cache hierarchy
- Energy management
- Interconnect
Core design

- To provide good single thread performance
  - 12 execution units: 2 FXU, 2 LSU, etc.
  - Deep out-of-order execution
    - Fetch up to 8 instructions per cycle
    - Decode and dispatch up to 6 instructions per cycle
    - Issue and execute up to 8 instructions per cycle

Simultaneous multi-threading

- A single-thread is unlikely to keep all resources busy
- Provide 4-way SMT to achieve a better throughput
- However, more threads running doesn’t always lead to better performance
  - Resource contention between threads may hurt performance
- POWER7 provides 3 SMT modes: ST, SMT2, and SMT4
- Applications can choose between single thread performance and throughput by choosing SMT mode

Simultaneous multi-threading

- One way to achieve 4-way SMT is to have all 4 threads share all resources
  - Energy consumption is too high
  - Complex logic -> need large area

- POWER7 uses a partitioned approach
  - In SMT4 mode, one physical general-purpose register (GPR) file supports two threads
  - Each GPR file feeds one FXU pipeline and one LSU pipeline
  - A pair of GPR files support four threads total

SMT implementation

- In SMT4 mode
  - GPR0, FX0, LS0 can only be used by thread T0 and T1
  - Thread T2 and T3 use the other set

- In ST and SMT2 mode
  - Two GPR files have identical contents
  - Instructions of any thread can be dispatched to each issue queue

# Cache overview

<table>
<thead>
<tr>
<th><strong>POWER6 (assuming 5-GHz core)</strong></th>
<th><strong>POWER7 (assuming 4-GHz core)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>64 KB store-through L1 D-cache</td>
<td>32 KB store-through L1 D-cache</td>
</tr>
<tr>
<td>0.8ns latency, 80 GB/s private</td>
<td>0.5ns latency, 192 GB/s private</td>
</tr>
<tr>
<td>4 MB store-in L2 cache</td>
<td>256 KB store-in L2 cache</td>
</tr>
<tr>
<td>~5.0-ns latency, 160 GB/s private</td>
<td>2.0-ns latency, 256 GB/s private</td>
</tr>
<tr>
<td>32 MB victim L3 cache</td>
<td>4 MB partial victim local L3 region</td>
</tr>
<tr>
<td>~35-ns latency, 80 GB/s shared by 2</td>
<td>~6.0-ns latency, 128 GB/s private</td>
</tr>
<tr>
<td></td>
<td>32 MB adaptive victim L3 cache</td>
</tr>
<tr>
<td></td>
<td>~30-ns latency, 512 GB/s shared by 8</td>
</tr>
</tbody>
</table>

- **Capacity deduction in L1 and L2 helps:**
  - Reduce L1 and L2 latency
  - Increase bandwidth of L1 and L2

- **On chip L3 has better latency and bandwidth**

L1 data cache

- Objective: provide data at low latency and high bandwidth
- 32KB 8-way set-associative, inclusive L2 cache

Low latency: cache line size is 128B consisting of four 32B sectors
  - A dedicated interface from L2 supplies a sector in each processor cycle
  - Each sector has a valid bit, loads can hit when the corresponding sector is validated

High bandwidth: for concurrent read and write:
  - Provides two read ports and one write port
  - Divided into 64 banks, which allows for concurrent read and write if they are not using the same bank

Use a store-through design: all stores go directly to L2
  - No L1 cast-out for stores

L2 cache

- Objective: absorb store traffic from L1 at low latency
- 256KB 8-way set-associative, 256 GB/s bandwidth

Use a fully associative 16-deep 32-byte entry store cache:
- Absorb store traffic (up to 16 bytes per cycle)
- If four of these entries comprise updates to the same 128B coherence granule, they will be grouped together into a single coherence dispatch

L3 cache

- 32MB consists of eight 4MB regions
  - Each region is 8-way set-associative and has an 128B cache line
  - Each region is tightly coupled with corresponding L2 cache
- Exclusive or inclusive L3 cache, determined by the partial victim cache (local L3) management policy
  - Exclusive provides more capacity and cache associativity
  - Inclusive helps reduce energy usage
- L3 cache is implemented using eDRAM
  - Access latency and cycle time is slightly worse than conventional SRAM
  - Requires one-third area and one fifth standby energy compared to equivalent SRAM, enabling on-chip L3 cache
    - On-chip cache has a lower latency
    - On-chip cache has better bandwidth and saves off-chip bandwidth

## Cache states

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
<th>Authority</th>
<th>Sharers and scope</th>
<th>Source data</th>
<th>Data cast-out</th>
<th>Scope cast-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Invalid</td>
<td>None</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>None</td>
</tr>
<tr>
<td>ID</td>
<td>Deleted, do not allocate</td>
<td>None</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>None</td>
</tr>
<tr>
<td>S</td>
<td>Shared</td>
<td>Read</td>
<td>Yes, scope unknown</td>
<td>No</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>SL</td>
<td>Shared, local data source</td>
<td>Read</td>
<td>Yes, scope unknown</td>
<td>At request</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>T</td>
<td>Formerly MU, now shared</td>
<td>Update</td>
<td>Yes, probably global</td>
<td>If notified</td>
<td>Yes</td>
<td>Required, global</td>
</tr>
<tr>
<td>TE</td>
<td>Formerly ME, now shared</td>
<td>Update</td>
<td>Yes, probably global</td>
<td>If notified</td>
<td>No</td>
<td>Required, global</td>
</tr>
<tr>
<td>M</td>
<td>Modified, avoid sharing</td>
<td>Update</td>
<td>No</td>
<td>At request</td>
<td>Yes</td>
<td>Optional, local</td>
</tr>
<tr>
<td>ME</td>
<td>Exclusive</td>
<td>Update</td>
<td>No</td>
<td>At request</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>MU</td>
<td>Modified, bias toward sharing</td>
<td>Update</td>
<td>No</td>
<td>At request</td>
<td>Yes</td>
<td>Optional, local</td>
</tr>
<tr>
<td>IG</td>
<td>Invalid, cached scope-state</td>
<td>None</td>
<td>N/A, probably global copies</td>
<td>N/A</td>
<td>N/A</td>
<td>Required, global</td>
</tr>
<tr>
<td>IN</td>
<td>Invalid, scope predictor</td>
<td>None</td>
<td>N/A, probably local copies</td>
<td>N/A</td>
<td>N/A</td>
<td>None</td>
</tr>
<tr>
<td>TN</td>
<td>Formerly MU, now shared</td>
<td>Update</td>
<td>Yes, local</td>
<td>If notified</td>
<td>Yes</td>
<td>Optional, local</td>
</tr>
<tr>
<td>TEN</td>
<td>Formerly ME, now shared</td>
<td>Update</td>
<td>Yes, local</td>
<td>If notified</td>
<td>No</td>
<td>None</td>
</tr>
</tbody>
</table>

L2 misses

For L2 misses, try to fetch them in local L3 first

For local L3 misses, fetch them from remaining L3 region and memory

L2 cache use the local L3 cache as a victim cache

Other L3 region also use the local L3 cache as a victim cache in certain cases

Heuristics give precedence to associated L2 cache while allowing load and capacity balancing among all eight L3 regions

L3 replacement algorithm

- Algorithm categorizes L3 lines into two classes:
  1. Victims via an L2 cast-out operation by the associated L2 cache
  2. Victims via a lateral cast-out operation by other L3 regions; residual shared copies of lines of L2; invalid lines.

- Every time algorithm needs to allocate a line:
  1. It selects the LRU line from 2nd class
  2. If no such lines exist, it selects the LRU line from the 1st class

L2 cast-out

(1) An L2 cast-out received

(2) Check if there is a copy of the line, or an invalid slot. Allocate a line if not.
Lateral cast-out

(2) Check if there is a copy of the line. If so, edit the line’s state and no data is moved

(1) A lateral cast-out received

(3) If not, decide whether to accept or not depending on capacity usage and traffic volume

(4) If accepted, allocate a line. Convert LRU first-class line to most recently used second class to enlarge the supply of second-class lines
Line eviction

If a first-class line is evicted, determine whether to laterally cast-out the line or write it back to memory if dirty.

If a dirty second-class line is evicted, write back to memory.

If the line is laterally cast-out, determine which of other 7 regions to target.

Energy management

- While having 4 times as many cores on the chip, POWER7 chip is in the same power envelope as POWER6
  - Reduced frequency (From 3.6 ~ 5 GHz to 2.4–4.25 GHz)
  - Microarchitecture changes (e.g., combine GPR, FPR and VR)
  - On chip eDRAM (one fifth standby energy compared to equivalent SRAM)
  - State of each chiplet (core with its L2 and local L3 cache) is configurable: 4 idle states and 8 performance states.

Idle states

- **Nap**
  - Turn off clocks to execution units
  - Caches remain coherent

- **Fast sleep**
  - Turn off core plus L2 cache clocks
  - L3 cache remains operational

- **Deep sleep**
  - Power off voltage to core and L2

- **Winkle**
  - Power off entire chiplet

Content & figure credit: Luigi Brochard. HPC - Innovative technologies for power management based on Power Architecture. March 29, 2011.
Performance states

- 8 P-states
- Each state has a frequency and voltage

<table>
<thead>
<tr>
<th>Power Save States</th>
<th>Freq (Max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pstate0</td>
<td>1.1</td>
</tr>
<tr>
<td>Pstate1</td>
<td>1.0</td>
</tr>
<tr>
<td>Pstate2</td>
<td>0.9</td>
</tr>
<tr>
<td>Pstate3</td>
<td>0.8</td>
</tr>
<tr>
<td>Pstate4</td>
<td>0.7</td>
</tr>
<tr>
<td>Pstate5</td>
<td>0.6</td>
</tr>
<tr>
<td>Pstate6</td>
<td>Fmax@vmin</td>
</tr>
<tr>
<td>Pstate7</td>
<td>0.50</td>
</tr>
</tbody>
</table>

Content & figure credit: Luigi Brochard. HPC - Innovative technologies for power management based on Power Architecture. March 29, 2011.
Frequency and voltage adjustment

- Critical path monitor (CPM) circuit measures in real time
- Adjust clock frequency according to the output of CPM
- Adjust the processor voltage level periodically to achieve a specified average clock frequency target

Content and figure credit: Active Management of Timing Guardband to Save Energy in POWER7, Charles R. Lefurgy et al. MICRO-44, pp. 1–11, 2011.
Critical path monitor
Measure timing margin

- Use Critical Path Monitor (CPM) circuit. Mimics behavior of real critical path.
- Each cycle: generate pulse, traverse synthesized critical path and calibrated delay, capture in edge detector

- Edge detector 12-bit output: (bit 0 = less margin, bit 11 = more margin)
On-chip interconnect

- Utilizes a non-blocking broadcast-based coherence-transport mechanism
  - **Coherence request**
    - (1) Send inward toward the even/odd arbitration logic
    - (2) Up to one even and one odd request may be granted at each on-chip bus cycle.
    - (3) Requests are broadcast outward within the chip
  - **Coherence responses**
    - (1) Send inward toward the even/odd coherence logic
    - (2) Once a final coherence decision is made in response to a given request, a notification is broadcast outward in the chip.

Multi-chip interconnect

- Two-level topology support up to 32 chips to form a 256-way SMP system
  - First level nodal structure
    - Fully connected 4 chips
  - Second level system structure
    - Fully connected 8 nodes
  - Coherence domain is the system.

Cluster interconnect

- Three-level topology support up to a 512K processor cluster.
  - First level nodal structure
    - Fully connected 4 chips and cluster chip
  - Second level SuperNode structure
    - Fully connected 32 nodes
  - Third level cluster structure
    - Fully connected 512 SuperNodes.
  - Coherence domain is within each nodal.
IBM POWER7 performance modeling, verification, and evaluation
Overview

- Performance modeling
  - Used for comparing alternative designs, code tuning, etc.

- Performance verification

- Performance monitoring

- Performance evaluation
Goal: analyze application performance on a complex out-of-order core
  ◦ Pinpoint instructions causing losses, e.g. cache misses

Challenge: precise attribution of events is hard
  ◦ Multiple instructions and events per cycle

Sampling-based performance analysis

- Asynchronous sampling is commonly used to measure performance
  - Periodically sample activity
  - Sampled activity is likely representative of aggregate behavior

- For a specific instruction, POWER7 provides:
  - Continuous sampling – sample every instruction executed.
  - Random sampling – randomly sample instructions.
Instruction sampling

- An instruction is (randomly) marked at dispatch.

- Special registers record IP and effective data address.

- Can record up to 32 events of a marked instruction during its lifetime in the pipeline.
  - Cache miss indication and reload information
  - TLB miss and reload information
  - etc.

Hierarchical breakdown of cycles

SMT performance

- Using SAP SD benchmark
  - ST mode: 2.7 GIPS (gigainstruction per second)
  - SMT2 mode: 3.8 GIPS, 1.4× performance gain
  - SMT4 mode: 4.7 GIPS, 1.7× performance gain
- ST has the best single-threaded performance

SMT performance

- Using OLTP workloads
  - ST mode: 1.3 GIPS (gigainstruction per second)
  - SMT2 mode: 1.9 GIPS, 1.5× performance gain
  - SMT4 mode: 2.4 GIPS, 1.9× performance gain
- ST has the best single-threaded performance

**Turbo Core mode**

- In *Turbo Core* mode, 4 out of 8 cores are disabled.
  - Cache of disabled cores becomes an extension to the local L3 cache for the running cores.
  - Only 4 threads are sharing resources like memory capacity, memory bandwidth, etc.
  - About 10% frequency boost by shifting power from the disabled cores to the active components.

SPECjbb gains in Turbo Core

- Single thread performance of SPECjbb improved by 19%

SPECjbb L1 D-cache miss reloads

OLTP gains in *Turbo Core*

- Performance of OLTP workload improved by 13%

OLTP L1 D-cache miss reloads

Single-precision FP performance

- Performance of Fast Fourier transforms (FFTs)
  - N is the size of the Fourier transform

Double-precision performance

- Performance of Fast Fourier transforms (FFTs)
  - N is the size of the Fourier transform

Scalability

- SPECjbb scalability
- SAP workload scalability

Evolution of POWER Processors
POWER7+ (2012)

- 32 nm, 567 mm²
- 8 cores per chip
- 4-way SMT per core

Each core has
  - 32KB L1 I/D cache
  - 256 KB L2 cache
  - 10MB Local L3 region

- 80 MB shared L3

Figure credit: Power7+, Scott Taylor, Hot Chips 24, August 2012. http://www.hotchips.org/archives/hc24
An improved core

- Up to 25% frequency gain due to mapping into 32nm technology and power management improvements

- L3 cache capacity increased by 2.5x

- Doubled single precision floating-point performance
Optimized in two dimensions

- Increased Clock Speed
  - 2.5x L3 cache
  - Thread Strength
  - Thread Strength
  - SMP Scaling

Scale UP

Scale OUT

2x Cores / socket  →  Throughput
5x L3 cache  →  Bandwidth Amplifier

Figure credit: Power7+, Scott Taylor, Hot Chips 24, August 2012. http://www.hotchips.org/archives/hc24
Performance comparison

![Graph showing performance comparison between different workloads and architectures.](http://www.hotchips.org/archives/hc24)

Figure credit: Power7+, Scott Taylor, Hot Chips 24, August 2012.
http://www.hotchips.org/archives/hc24
POWER7+ accelerators

- Dedicated hardware accelerators that provide significant speedup for SSL and encrypted file system. Support in three cryptography aspects:
  - Asymmetric Math Functions (AMF)
    - Support RSA cryptography and ECC cryptography
  - Advanced Encryption Standard (AES) / Secure Hash Algorithm (SHA)
    - Provide encryption and verification respectively
  - Random Number Generator (RNG)
    - Can’t be algorithmically reverse engineered.

- Speed up for active memory expansion (AME)
  - Highly efficient hardware implementation of proprietary 832 compression algorithm

POWER8 (2014)

- 22 nm, 650 mm²
- 12 cores per chip
- 8-way SMT per core

Each core has:
- 32KB L1 I cache
- 64KB L1 D cache
- 512KB L2 cache
- 8MB Local L3 region

- 96 MB on-chip shared L3
- 128MB off-chip L4 cache

- Improved bandwidth

Content and figure credit: You won't find this in your phone: A 4GHz 12-core Power8 for badass boxes, Timothy Prickett Morgan, 2013. [http://www.theregister.co.uk/2013/08/27/ibm_power8_server_chip/?page=1](http://www.theregister.co.uk/2013/08/27/ibm_power8_server_chip/?page=1)
POWER9 variants (2016)

Four targeted implementations

SMP scalability / Memory subsystem

- **Scale-Out – 2 Socket Optimized**
  - Robust 2 socket SMP system
  - Direct Memory Attach
  - Up to 8 DDR4 ports
  - Commodity packaging form factor

- **Scale-Up – Multi-Socket Optimized**
  - Scalable System Topology / Capacity
  - Large multi-socket
  - Buffered Memory Attach
  - 8 Buffered channels

Core Count / Size

- **SMT4 Core**
  - 24 SMT4 Cores / Chip
  - Linux Ecosystem Optimized

- **SMT8 Core**
  - 12 SMT8 Cores / Chip
  - PowerVM Ecosystem Continuity

POWER9, Processor for the Cognitive Era, Brian Thompto, Hot Chips 28, 2016.
POWER9 SMT

Available with SMT8 or SMT4 Cores
8 or 4 threaded core built from modular execution slices

POWER9 SMT8 Core
- PowerVM Ecosystem Continuity
- Strongest Thread
- Optimized for Large Partitions

POWER9 SMT4 Core
- Linux Ecosystem Focus
- Core Count / Socket
- Virtualization Granularity

POWER9, Processor for the Cognitive Era, Brian Thompto, Hot Chips 28, 2016.
POWER9 cache hierarchy

- 512K L2 per SMT8 Core
- 120MB NUCA L3 architecture
  - partitioned into 10MB blocks
  - each shared by 2 cores
  - 12 x 20-way associative regions
- Advanced replacement policies
- Fed by 7 TB/s on-chip bandwidth
- Move data in/out at 256 GB/s per SMT8 core

POWER9, Processor for the Cognitive Era, Brian Thompto, Hot Chips 28, 2016.
POWER9 connectivity

- Extreme Processor / Accelerator Bandwidth and Reduced Latency
- Coherent Memory and Virtual Addressing Capability for all Accelerators
- OpenPOWER Community Enablement – Robust Accelerated Compute Options

- State of the Art I/O and Acceleration Attachment Signaling
  - PCIe Gen 4 x 48 lanes – 192 GB/s duplex bandwidth
  - 25G Link x 48 lanes – 300 GB/s duplex bandwidth

- Robust Accelerated Compute Options with OPEN standards
  - On-Chip Acceleration – Gzip x1, 842 Compression x2, AES/SHA x2
  - CAPI 2.0 – 4x bandwidth of POWER8 using PCIe Gen 4
  - NVLink 2.0 – Next generation of GPU/CPU bandwidth and integration
  - New CAPI – High bandwidth, low latency and open interface using 25G Link

POWER9, Processor for the Cognitive Era, Brian Thompto, Hot Chips 28, 2016.