IBM POWER9 SMT Deep Dive Summit Training Workshop

Brian Thompto
POWER Systems, IBM Systems
POWER9 Processor

Performance Optimized for Cognitive Workloads

- New Core Microarchitecture
- Enhanced cache hierarchy
  - Up to 120 MB / Chip
- On Chip Super-Highway
- Connect Cores, Caches
- And Accelerators / GPU’s
- 14nm silicon technology

Open Interfaces for Accelerated Computing

- 1st processor introduction of PCIeG4
- 25G Coherent Link:
  - Next-gen CAPI technology
- NVLINK2.0 for GPU attach

Family of Scale-out & Scale-up Optimized Offerings

- Dual Memory Subsystems optimized for Scale Out (latency/density) & Enterprise (capacity/bandwidth/RAS)
  - 12 SMT8 or 24 SMT4 cores (96 threads)
- High bandwidth scale-up fabric: 2-16 socket offerings with 2-4x chip-to-chip interconnect bandwidth
POWER9 – AC922 with 6 GPU’s

Images / diagrams modified from:
"IBM POWER9 systems designed for commercial cognitive and cloud", IBM J. Res. & Dev., vol. 62, no. 4/5, 2018
POWER9 – Core and Cache Topology

POWER9 Chip with 22 / 24 Active Cores
Up to 88 Threads / Socket

2 x POWER9 SMT4 Core : 1-4 threads each
L2 Cache (512k) and L3 Cache (10MB) : 1-8 threads

Images / diagrams modified from:
POWER9: Cache Capacity

Caches per pair of SMT4 cores (up to 1-8 threads)
- L2: 512k, 8-way
- L3: 10 MB, 20-way
  - Enhanced L3 Cache Effectiveness with enhanced Replacement
  - Aggregate 110 MB, 11 x 20 way associativity when 22 cores active (out of 24) on Summit

256 GB/s each Core Pair

7 TB/s

© 2018 IBM Corporation
New POWER9 Microarchitecture

Optimized for Cognitive Workloads & Stronger Thread Performance

- Shorter pipeline & improved scheduling / branch prediction for unoptimized code & interpretive languages
- Increased execution bandwidth for a range of workloads including commercial, cognitive and analytics
- Adaptive features for improved efficiency and performance

POWER9 SMT4 Core – Sliced Micro-arch

POWER9 Pipeline (SMT4)

POWER9 SMT4-Core Microarchitecture

POWER9 SMT4 Core – Sliced Micro-arch

Images / diagrams modified from:
SMT4 Core Resources

Fetch / Branch
- 32kB, 8-way Instruction Cache
- 8 fetch, 6 decode
- 1x branch execution

Slices issue VSU and AGEN
- 4x scalar-64b / 2x vector-128b
- 4x load/store AGEN

Vector Scalar Unit (VSU) Pipes
- 4x ALU + Simple (64b)
- 4x FP + FX-MUL + Complex (64b)
- 2x Permute (128b)
- 2x Quad Fixed (128b)
- 2x Fixed Divide (64b)
- 1x Quad FP & Decimal FP
- 1x Cryptography

Load Store Unit (LSU) Slices
- 32kB, 8-way Data Cache
- Up to 4 DW load or store
Subset of cores enabled with Single Thread (ST)

**Individual cores are inactive**

- Inactive cores allow higher socket frequency via WOF Frequency Boost
- One thread gets access to the full Level-2 / Level-3 cache region
Thread Sharing of POWER9 Cores + Cache

1 Thread active per Core (ST)
- Each thread gets ½ of the core execution resources
- Threads share the Level-2 / Level-3 cache

ST x 1 core
11 threads per socket

ST x 2 cores
22 threads per socket
Thread Sharing of POWER9 Cores + Cache

- **2 Threads Active Per Core (SMT2)**
  - Each pair of threads shares ½ of each core’s execution resources
  - 4 threads share the Level-2 / Level-3 cache

**ST x 1 core**
- 11 threads per socket

**ST x 2 cores**
- 22 threads per socket

**SMT2 X 2 cores**
- 44 threads per socket
Thread Sharing of POWER9 Cores + Cache

**4 Threads Active Per Core (SMT4)**
- Each pair of threads shares ½ of each core’s execution resources
- 8 threads share the Level-2 / Level-3 cache
The "SMT Mode" setting limits the maximum number of threads dispatchable to each core. The default "SMT Mode" is SMT4.

Each core automatically switches modes depending on the number of threads dispatched by the OS. The "SMT Mode" setting limits the maximum number of threads dispatchable to each core.

© 2018 IBM Corporation
POWER9 – Core Compute

(1-2 threads) ST,SMT2
Fully Shared Execution Resources

Fetch / Branch
- 32kB, 8-way Instruction Cache
- 8 fetch, 6 decode
- 1x branch execution

Slices issue VSU and AGEN
- 4x scalar-64b / 2x vector-128b
- 4x load/store AGEN

Vector Scalar Unit (VSU) Pipes
- 4x ALU + Simple (64b)
- 4x FP + FX-MUL + Complex (64b)
- 2x Permute (128b)
- 2x Quad Fixed (128b)
- 2x Fixed Divide (64b)
- 1x Quad FP & Decimal FP
- 1x Cryptography

Load Store Unit (LSU) Slices
- 32kB, 8-way Data Cache
- Up to 4 DW load or store

SMT4 Core x 22 per Socket for Summit Systems
(4 threads) SMT4
Execution Resource Split by Thread Pair

Fetch / Branch
• 32kB, 8-way Instruction Cache
• 8 fetch, 6 decode
• 1x branch execution

Slices issue VSU and AGEN
• 4x scalar-64b / 2x vector-128b
• 4x load/store AGEN

Vector Scalar Unit (VSU) Pipes
• 4x ALU + Simple (64b)
• 4x FP + FX-MUL + Complex (64b)
• 2x Permute (128b)
• 2x Quad Fixed (128b)
• 2x Fixed Divide (64b)
• 1x Quad FP & Decimal FP
• 1x Cryptography

Load Store Unit (LSU) Slices
• 32kB, 8-way Data Cache
• Up to 4 DW load or store

SMT4 Core x 22 per Socket for Summit Systems

SMT4 Core

128b Super-slice
Thread Sharing of POWER9 Cache + Prefetch

**Cache**
- **L1 caches are per SMT4 core, 1-4 threads**
- **L2/L3 caches are per pair of SMT4 cores**
  - 64B reload bus from L2/L3 is shared by 1-2 cores
  - 16B store bus per SMT4 core to L2

<table>
<thead>
<tr>
<th>Cache</th>
<th>Domain</th>
<th>Size</th>
<th>Threads Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1 I-Cache</td>
<td>Core</td>
<td>32k x 8 way</td>
<td>4</td>
</tr>
<tr>
<td>L1 D-Cache</td>
<td>Core</td>
<td>32k x 8 way</td>
<td>4</td>
</tr>
<tr>
<td>L2 Cache</td>
<td>Core Pair</td>
<td>512k x 8 way</td>
<td>8</td>
</tr>
<tr>
<td>L3 Cache</td>
<td>Core Pair</td>
<td>10M x 20 way</td>
<td>8</td>
</tr>
</tbody>
</table>

**Prefetch**
- **L1 Data Cache Miss Queue (LMQ) per core**
  - Supports Demand and L1 Prefetch requests
- **L3 Data Prefetch Queues are per pair of SMT4 cores**

<table>
<thead>
<tr>
<th>Queue</th>
<th>Domain</th>
<th>Size</th>
<th>Threads Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMQ</td>
<td>Core</td>
<td>8, 12 (w/ atomics)</td>
<td>4</td>
</tr>
<tr>
<td>L3 Prefetch</td>
<td>Core Pair</td>
<td>32</td>
<td>8</td>
</tr>
</tbody>
</table>
CPU Threading Awareness – SMT Choice

• Awareness: Linux Thread numbering (affinity):
  – Core 0 is numbered threads: 0, 1, 2, 3
    So one thread per core will be 0, 4, 8, …

• js_run handles affinity – automatically spreads out threads

• When to use SMT?
  – **Recommend experimentation!**
  – Branch heavy, serial (dependency heavy) and cache miss heavy codes tend to benefit from SMT
  – Highest SMT is not always best performance, but **often is** for throughput
    o For latency sensitive, e.g. feeding GPU – ST often best, but not always
    o For CPU, throughput may be limited by specific resources depending on code

• Some notes on variability
  – Balance parallel threads in the same thread mode
  – SMT may exacerbate run variability
    o e.g. easier to hit corner cases of cache capacity by way, etc
  – Turning off ASLR may help limit run variability for CPU dominated codes (corner cases)
Special notices

This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquiries, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied.

All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client's credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

All prices shown are IBM's United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

Revised September 26, 2006
Special notices (continued)

IBM, the IBM logo, ibm.com AIX, AIX (logo), IBM Watson, DB2 Universal Database, POWER, PowerLinux, PowerVM, PowerVM (logo), PowerHA, Power Architecture, Power Family, POWER Hypervisor, Power Systems, Power Systems (logo), POWER2, POWER3, POWER4, POWER4+, POWER5, POWER5+, POWER6, POWER6+, POWER7, POWER7+, and POWER8 are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries.

A full list of U.S. trademarks owned by IBM may be found at: http://www.ibm.com/legal/copytrade.shtml.

NVIDIA, the NVIDIA logo, and NVLink are trademarks or registered trademarks of NVIDIA Corporation in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries or both.

PowerLinux®™ uses the registered trademark Linux® pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the Linux® mark on a world-wide basis.

The Power Architecture and Power.org wordmarks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

The OpenPOWER word mark and the OpenPOWER Logo mark, and related marks, are trademarks and service marks licensed by OpenPOWER.

Other company, product and service names may be trademarks or service marks of others.