

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MERCEDES-BENZ USA, LLC,

Petitioner,

v.

DAEDALUS PRIME LLC,

Patent Owner.

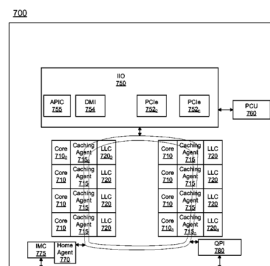
Case No.: IPR2023-01344

U.S. Patent No. 9,575,895

Patent Owner's Demonstrative Exhibits

'895 Patent

(12) United States Patent Liu et al.	(10) Patent No.: US 9,575,895 B2 (45) Date of Patent: *Feb. 21, 2017
(54) PROVIDING COMMON CACHING AGENT FOR CORE AND INTEGRATED INPUT/OUTPUT (IO) MODULE	(52) U.S. CL. CPC <i>G06F 12/084</i> (2013.01); <i>G06F 12/0815</i> (2013.01); <i>G06F 12/0831</i> (2013.01); <i>G06F 2212/452</i> (2013.01); <i>G06F 2212/621</i> (2013.01); <i>Y02B 60/1225</i> (2013.01)
(71) Applicant: Intel Corporation , Santa Clara, CA (US)	(58) Field of Classification Search CPC G06F 12/0831; G06F 12/0815; G06F 12/084; G06F 2212/452; G06F 2212/621 See application file for complete search history.
(72) Inventors: Yen-Cheng Liu , Portland, OR (US); Robert G. Blankenship , Tacoma, WA (US); Geeyarpuram N. Santhanakrishnan , Mercer Island, WA (US); Ganapati N. Srinivasa , Portland, OR (US); Kenneth C. Creta , Gig Harbor, WA (US); Sridhar Muthrasanallur , Bangalore (IN); Bahaa Fahim , San Jose, CA (US)	(56) References Cited U.S. PATENT DOCUMENTS 6,009,488 A 12/1999 Kavipurapu 7,165,131 B2 1/2007 Creta 7,210,000 B2 4/2007 Creta 7,689,778 B2 3/2010 Liu et al. 7,937,534 B2 5/2011 Madhukaramukumana (Continued)
(73) Assignee: Intel Corporation , Santa Clara, CA (US)	(57) ABSTRACT In one embodiment, the present invention includes a multicore processor having a plurality of cores, a shared cache memory, an integrated input/output (IIO) module to interface between the multicore processor and at least one IO device coupled to the multicore processor, and a caching agent to perform cache coherency operations for the plurality of cores and the IIO module. Other embodiments are described and claimed.
(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 69 days. This patent is subject to a terminal disclaimer.	OTHER PUBLICATIONS Intel Corporation, "An Introduction to the Intel QuickPath Interconnect," Jan. 2009, pp. 1-22. (Continued) <i>Primary Examiner</i> — Edward Dudek, Jr. (74) <i>Attorney, Agent, or Firm</i> — Trop, Pruner & Hu, P.C.
(21) Appl. No.: 14/609,620	
(22) Filed: Jan. 30, 2015	
(65) Prior Publication Data US 2015/0143051 A1 May 21, 2015	
Related U.S. Application Data	
(63) Continuation of application No. 13/324,053, filed on Dec. 13, 2011, now Pat. No. 8,984,228.	
(51) Int. CL. G06F 12/08 (2016.01)	17 Claims, 7 Drawing Sheets



The '895 patent describes a solution for particular “problems [that] arise once an IO component is integrated on the same chip with a multiprocessor.”

The solution employs “a CPU caching agent [] configured to support both CPU traffic and IO traffic as well, thereby resolving the scaling issue.”

The single caching agent can “be distributed such that each of the different portions of the caching agent can be associated with a corresponding core and LLC [(last level cache)] bank or slice.”

Ex. 1001: 1:25-26, 62-64; 2:24-26, 64-66; 3:7-19 ; 6:32-39
POR: 3-4

'895 Claim Construction

a single caching agent means one caching agent

1. A processor comprising:

a plurality of cores, a shared cache memory, a memory controller to interface with a memory coupled to the processor, an integrated input/output (IIO) module to interface between the processor and an IO device coupled to the processor and a caching agent to perform cache coherency operations for the plurality of cores and the IIO module, wherein the processor is to receive an allocation transaction from the IO device and directly store data of the allocation transaction into the shared cache memory, wherein the caching agent is a single caching agent for the processor and includes a plurality of distributed portions each associated with a corresponding one of the plurality of cores.

11. A non-transitory machine-readable medium having stored thereon instructions, which if performed by a machine cause the machine to perform a method comprising:

receiving an allocation transaction in an integrated input/output (IIO) module of a multicore processor from an IO device coupled to the multicore processor;

forwarding the allocation transaction to a caching agent of the multicore processor to cause the data to be stored into a shared cache memory of the multicore processor, the caching agent shared by a plurality of cores of the multicore processor and the IIO module, wherein the caching agent is a single caching agent for the multicore processor and includes a plurality of distributed portions each associated with a corresponding one of the plurality of cores; and

directly storing the data of the allocation transaction into the shared cache memory of the multicore processor without sending the data to a system memory coupled to the multicore processor.

15. A system comprising:

a first multicore processor including a first plurality of cores, a first shared cache memory, a first integrated input/output (IIO) module to interface between the first multicore processor and a first plurality of IO devices coupled to the first multicore processor, and a first caching agent to perform cache coherency operations for the first plurality of cores and the first IIO module, wherein the first caching agent is to receive an allocation transaction from the first IIO module and directly store data of the allocation transaction into the first shared cache memory, wherein the first caching agent is a single caching agent for the first multicore processor and includes a plurality of distributed portions each associated with a corresponding one of the first plurality of cores, and the first shared cache memory comprises a last level cache (LLC) including a plurality of distributed portions each associated with a corresponding one of the first plurality of cores;

a second multicore processor including a second plurality of cores, a second shared cache memory, a second IIO module to interface between the second multicore processor and a second plurality of IO devices coupled to the second multicore processor, and a second caching agent to perform cache coherency operations for the second plurality of cores and the second IIO module; and

a peripheral controller coupled to at least one of the first and second multicore processors.

Ex. 1001: 7:47-61; 8:26-45; 8:63-10:5
POR: 4-5, 7-10

'895 Challenges

Claims Challenged	35 U.S.C. §	References
1, 3-7, 11-13, 15-17	102/103	Sinharoy
2, 10	103	Sinharoy in view of Xu
8, 9, 14	103	Sinharoy in view of Tang
1, 5-9, 11,14, 15, 17	102/103	Tang
2-4, 10, 16	103	Tang in view of Harikumar

Sinharoy

Cache hierarchy

POWER7 (assuming 4-GHz core)

32 KB store-through L1 D-cache
0.5ns latency, 192 GB/s private

256 KB store-in L2 cache
2.0-ns latency, 256 GB/s private

4 MB partial victim local L3 region
~6.0-ns latency, 128 GB/s private

32 MB adaptive victim L3 cache
~30-ns latency, 512 GB/s shared by 8

L2 cache

The private 256-KB POWER7 L2 cache utilizes 128 byte lines and is eight-way set associative. It is comprised of a single centralized controller, with two address-hashed cache data arrays.

L3 cache

The shared 32-MB POWER7 L3 cache is composed of eight 4-MB L3 regions. Each L3 region utilizes 128-byte lines and is eight-way set associative. A given L3 region comprises a single centralized controller, with four address hashed eDRAM cache data banks and four address hashed SRAM directory banks. The local L3 region is tightly

Sinharoy at p. 16
POR at 14-15

Sinharoy

Horst Testimony

9 So Sinharoy actually says the L2 cache is
10 comprised of a "single centralized controller,
11 correct"?

12 A That's one sentence from the -- from the
13 reference.

14 Q Right.

15 And that single centralized controller's
16 what I'm asking about. Is that one controller for
17 all of the L2 caches?

18 A That's -- they're structurally
19 individualized cache controllers for each L2
20 cache.

Horst depo, Ex. 2003, 15:15-20
PO Sur-reply at 5

1 UNITED STATES PATENT AND TRADEMARK OFFICE
2 * * *
3 BEFORE THE PATENT TRIAL AND APPEAL BOARD
4 * * *
5
6 Mercedes-Benz USA, LLC,
7 Petitioner,
8 v.
9 Daedalus Prime LLC,
10 Patent Owner.
11 * * *
12 Case IPR2023-01344
13 Patent 9,575,895
14 * * *
15
16 VIDEOCONFERENCED (BY ZOOM) DEPOSITION OF
17 ROBERT W. HORST, PH.D.
18 Bellevue, Washington
19 Wednesday, October 2, 2024
20 8:57 a.m. Pacific Daylight Time
21
22

Sinharoy

IBM POWER[®] multicore server processor

The IBM POWER[®] processor is the dominant reduced instruction set computing microprocessor in the world today, with a rich history of implementation and innovation over the last 20 years. In this paper, we describe the key features of the POWER7[®] processor chip. On the chip is an eight-core processor, with each core capable of four-way simultaneous multithreaded operation. Fabricated in IBM's 45-nm silicon-on-insulator (SOI) technology with 11 levels of metal, the chip contains more than one billion transistors. The processor core and caches are significantly enhanced to boost the performance of both single-threaded response-time-oriented, as well as multithreaded, throughput-oriented applications. The memory subsystem contains three levels of on-chip cache, with SOI embedded dynamic random access memory (DRAM) devices used as the last level of cache. A new memory interface using buffered double-data-rate-three DRAM and improvements in reliability, availability, and serviceability are discussed.

B. Sinharoy
R. Kalla
W. J. Starke
H. Q. Le
R. Cargnoni
J. A. Van Norstrand
B. J. Ronchetti
J. Stuechli
J. Leenstra
G. L. Guthrie
D. Q. Nguyen
B. Blener
C. F. Marino
E. Rietter
P. Williams

Introduction

Over the years, IBM POWER[®] processors [1–4] have introduced reduced instruction set computing architecture, advanced branch prediction, out-of-order execution, data prefetching, multithreading, dual-core chip, core accelerators [5], on-chip high-bandwidth memory controller, and highly scalable symmetric multiprocessing (SMP) interconnect. In this seventh-generation POWER processor [6–9], IBM introduces a balanced, eight-core multichip design, with large on-chip embedded dynamic random access memory (eDRAM) caches, and high-performance four-way multithreaded cores, implementing IBM PowerPC[®] architecture version 2.06 [10].

Starting in 2001 with POWER4[®] in 180-nm technology up to POWER6[®] in 65-nm, spanning four technology generations, all POWER processors were designed with two cores on the processor chip. The emphasis had been to stay with two cores per chip and significantly improve per-core performance by improving the core and cache microarchitectures, as well as exploiting the socket resources by only two cores on the chip. The socket resources that have an impact on performance include chip power, various off-chip bandwidths such as input/output (I/O), memory, and SMP bandwidths, as well as socket-level cache and

memory capacity. As we add more cores to share the existing socket resources in a given technology generation, per-core performance can be negatively affected. Unlike previous generation POWER processors, in POWER7[®], four times as many cores share the socket resources. Along with significant slowdown in the performance improvements from the silicon technology, this makes it more challenging to significantly improve the per-core performance of POWER7 over POWER6.

The goal of POWER7 was to improve the socket-level, core-level, and thread-level performances significantly over POWER6 while achieving all of the following in one technology generation.

- Reduce the core area and power to such an extent that four times as many cores can be placed on the processor chip in the same power envelope as that of POWER6. To reduce power, processor frequency is reduced in POWER7, while higher performance is achieved through much more emphasis on microarchitecture improvements, such as aggressive out-of-order execution, advanced four-way simultaneous multithreading (SMT), advanced branch prediction, advanced prefetching, and cache and memory latency reduction and bandwidth improvement.
- Fit the POWER7 chip in the same socket as POWER6 and utilize the same SMP and I/O buses as in POWER6

Digital Object Identifier: 10.1145/191201.1.212730

©Copyright 2011 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the journal reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied by any means or distributed royalty free without further permission by computer-based and other information-service systems. Permission to republish any other portion of this paper must be obtained from the editor.

0018-8646/11/05.00 © 2011 IBM

L3 cache

Storage accesses that miss both the L2 cache and the 4-MB local L3 region are broadcast to the coherence fabric and snooped by the memory controller, other L2 caches, possibly other L3 caches, and by the seven remote 4-MB L3 regions that comprise the remaining 28 MB of the on-chip L3 cache.

Sinharoy at p. 19
PO Sur-reply at 6

MB USA v. Daedalus Prime IPR2023-01344
Patent Owner's Demonstrative Exhibits -- Not Evidence
Daedalus Ex. 2004

Sinharoy

Horst Testimony

19 Q Yeah. My question was that if a cache
20 access misses, the cache controller doesn't have
21 any of this cache state information in Table 2,
22 because that cache line is not present in the

1 cache; isn't that correct?

2 A When the memory reference misses, it
3 doesn't have the information for that cache line,
4 but it does have information for the other cache
5 lines that own -- that are held in the location
6 that needs to be replaced with the correct cache
7 line.

Horst depo, Ex. 2003, 33:19 – 34:7
PO Sur-reply at 7

1 UNITED STATES PATENT AND TRADEMARK OFFICE
2 * * *
3 BEFORE THE PATENT TRIAL AND APPEAL BOARD
4 * * *
5
6 Mercedes-Benz USA, LLC,
7 Petitioner,
8 v.
9 Daedalus Prime LLC,
10 Patent Owner.
11 * * *
12 Case IPR2023-01344
13 Patent 9,575,895
14 * * *
15
16 VIDEOCONFERENCED (BY ZOOM) DEPOSITION OF
17 ROBERT W. HORST, PH.D.
18 Bellevue, Washington
19 Wednesday, October 2, 2024
20 8:57 a.m. Pacific Daylight Time
21
22

Sinharoy

IBM POWER7 multicore server processor

The IBM POWER[®] processor is the dominant reduced instruction set computing microprocessor in the world today, with a rich history of implementation and innovation over the last 20 years. In this paper, we describe the key features of the POWER7[®] processor chip. On the chip is an eight-core processor, with each core capable of four-way simultaneous multithreaded operation. Fabricated in IBM's 45-nm silicon-on-insulator (SOI) technology with 11 levels of metal, the chip contains more than one billion transistors. The processor core and caches are significantly enhanced to boost the performance of both single-threaded response-time-oriented, as well as multithreaded, throughput-oriented applications. The memory subsystem contains three levels of on-chip cache, with SOI embedded dynamic random access memory (DRAM) devices used as the last level of cache. A new memory interface using buffered double-data-rate-three DRAM and improvements in reliability, availability, and serviceability are discussed.

Introduction

Over the years, IBM POWER[®] processors [1–4] have introduced reduced instruction set computing architecture, advanced branch prediction, out-of-order execution, data prefetching, multithreading, dual-core chip, core accelerators [5], on-chip high-bandwidth memory controller, and highly scalable symmetric multiprocessing (SMP) interconnect.

In this seventh-generation POWER processor [6–9], IBM introduces a balanced, eight-core multichip design, with large on-chip embedded dynamic random access memory (eDRAM) caches, and high-performance four-way multithreaded cores, implementing IBM PowerPC[®] architecture version 2.06 [10].

Starting in 2001 with POWER4[®] in 180-nm technology up to POWER6[®] in 65-nm, spanning four technology generations, all POWER processors were designed with two cores on the processor chip. The emphasis had been to stay with two cores per chip and significantly improve per-core performance by improving the core and cache microarchitectures, as well as exploiting the socket resources by only two cores on the chip. The socket resources that have an impact on performance include chip power, various off-chip bandwidths such as input/output (I/O), memory, and SMP bandwidths, as well as socket-level cache and

memory capacity. As we add more cores to share the existing socket resources in a given technology generation, per-core performance can be negatively affected. Unlike previous generation POWER processors, in POWER7[®], four times as many cores share the socket resources. Along with significant slowdown in the performance improvements from the silicon technology, this makes it more challenging to significantly improve the per-core performance of POWER7 over POWER6.

The goal of POWER7 was to improve the socket-level, core-level, and thread-level performances significantly over POWER6 while achieving all of the following in one technology generation.

- Reduce the core area and power to such an extent that four times as many cores can be placed on the processor chip in the same power envelope as that of POWER6. To reduce power, processor frequency is reduced in POWER7, while higher performance is achieved through much more emphasis on microarchitecture improvements, such as aggressive out-of-order execution, advanced four-way simultaneous multithreading (SMT), advanced branch prediction, advanced prefetching, and cache and memory latency reduction and bandwidth improvement.
- Fit the POWER7 chip in the same socket as POWER6 and utilize the same SMP and I/O buses as in POWER6

B. Sinharoy
R. Kalla
W. J. Starke
H. Q. Le
R. Cargnoni
J. A. Van Norstrand
B. J. Ronchetti
J. Stuecheli
J. Leenstra
G. L. Outhrie
D. Q. Nguyen
B. Elsner
C. F. Marino
E. Rietter
P. Williams

Otherwise, a set of heuristics weigh the usage pattern (both capacity usage and traffic volume) of the targeted region to determine whether to accept the lateral cast-out.

Each core's L3 region serves as a victim cache for its associated L2 cache and for the other seven 4-Mbyte regions.

Heuristics strike a balance between these roles by monitoring and managing cache line allocation. When one L3 region evicts a cache line, it first decides whether to route the line to memory or to another L3 region. If another region is selected and cloned copies exist, the line is merged with the clones. If no clones exist, the L3 region determines which of the seven other regions is the best candidate to store the cache line. ***But, the targeted local L3 region must then decide whether to accept or reject the cache line.***

The need for such decision-making by the candidate localized L3 region demonstrates that ***there is no single cache controller operating***. It is up to the candidate L3 region to decide whether or not to accept the proposed cache line.

Sinharoy at p. 20
POR at 20
Brogioli Dec. at ¶ 38.

MB USA v. Daedalus Prime IPR2023-01344
Patent Owner's Demonstrative Exhibits -- Not Evidence
Daedalus Ex. 2004

Digital Object Identifier: 10.1145/1002301.1.212730

©Copyright 2011 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the journal reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied by any means or distributed royalty free without further permission by computer-based and other information-service systems. Permission to republish any other portion of this paper must be obtained from the Editor.

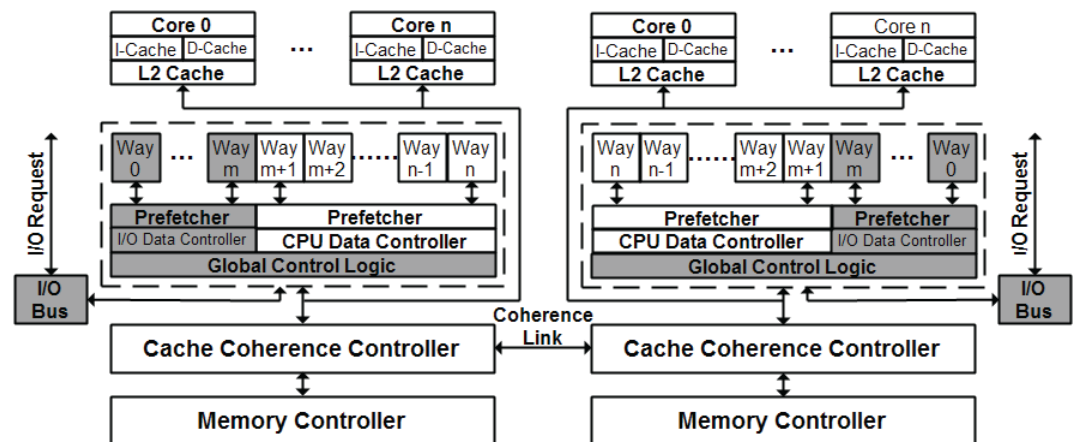
0010-8649/11/05.00 © 2011 IBM

Tang

Transcript of Robert W. Horst, Ph.D.
Conducted on October 2, 2024

47

1 the I/O data controller and the global control
2 logic and the box below it, the cache coherence
3 controller.
4 BY MR. FAHMI:
5 Q And ways M + 1 through way N are
6 controlled by the CPU data controller, the global
7 control logic, and the cash coherence controller;
8 is that right?
9 A That's right.
10 Q And there's only one cache coherence
11 controller per processor; is that right?
12 A That's what it shows here, yes.
13 Q And that one cache coherence controller
14 communicates with each of the cores; is that
15 right?
16 A Yes, it communicates with the cores.
17 Q And the one last level cache -- and all of
18 its associated ways and controllers and
19 pre-fetchers -- communicates with each of the
20 cores; is that right?
21 A The sum of all that control logic
22 communicates with each of the cores.



Tang: Fig. 10.
POR: 27-28

Conclusion

The Petition has failed to prove the unpatentability of any of the challenged claims.