

PETITIONERS' TRIAL HEARING DEMONSTRATIVES

ARM LTD, MEDIATEK INC.,
Petitioner,
v.
DAEDALUS PRIME LLC,
Patent Owner.

Inter Partes Review No.: IPR2025-00207
U.S. Patent No. 8,984,228

TRIAL HEARING
JUNE 25, 2026

DUANE MORRIS, LLP

Overview of Grounds

Ground	Basis	Cited Art	'228 Claims
1	103	Agarwal	1-2, 4-7, 11-14
2A	103	Sinharoy	1, 4, 6, 7, 11-14
2B	103	Sinharoy in view of Blumrich	2, 3, 16-18
2C	103	Sinharoy in view Sistla	5
2D	103	Sinharoy in view of Sistla in further view of Blumrich	19

Overview of Dispute

Dispute	Basis	Claims
1	Whether Agarwal discloses “ <i>a single caching agent...</i> ”	1-2, 4-7, 11-14
2	Whether Agarwal discloses “ <i>a table of requests...</i> ”	2
3	Whether Agarwal discloses “ <i>wherein the caching agent is to receive an allocation from the IIO module....</i> ”	6
4	Whether Agarwal discloses “ <i>directly storing data...without communicating with the system memory</i> ”	14
5	Whether Sinharoy discloses “ <i>a single caching agent....</i> ”	1, 4, 6, 7, 11-14
6	Whether Sistla discloses “ <i>wherein the plurality of distributed portions of the caching agent are coupled via a ring interconnect.</i> ”	5, 19

Not in Dispute

- Ground 1: PO does not argue separate patentability of Claims 4, 5, 7, 12 and 13

Overview of Ground 1 Agarwal

Overview of '282 Patent

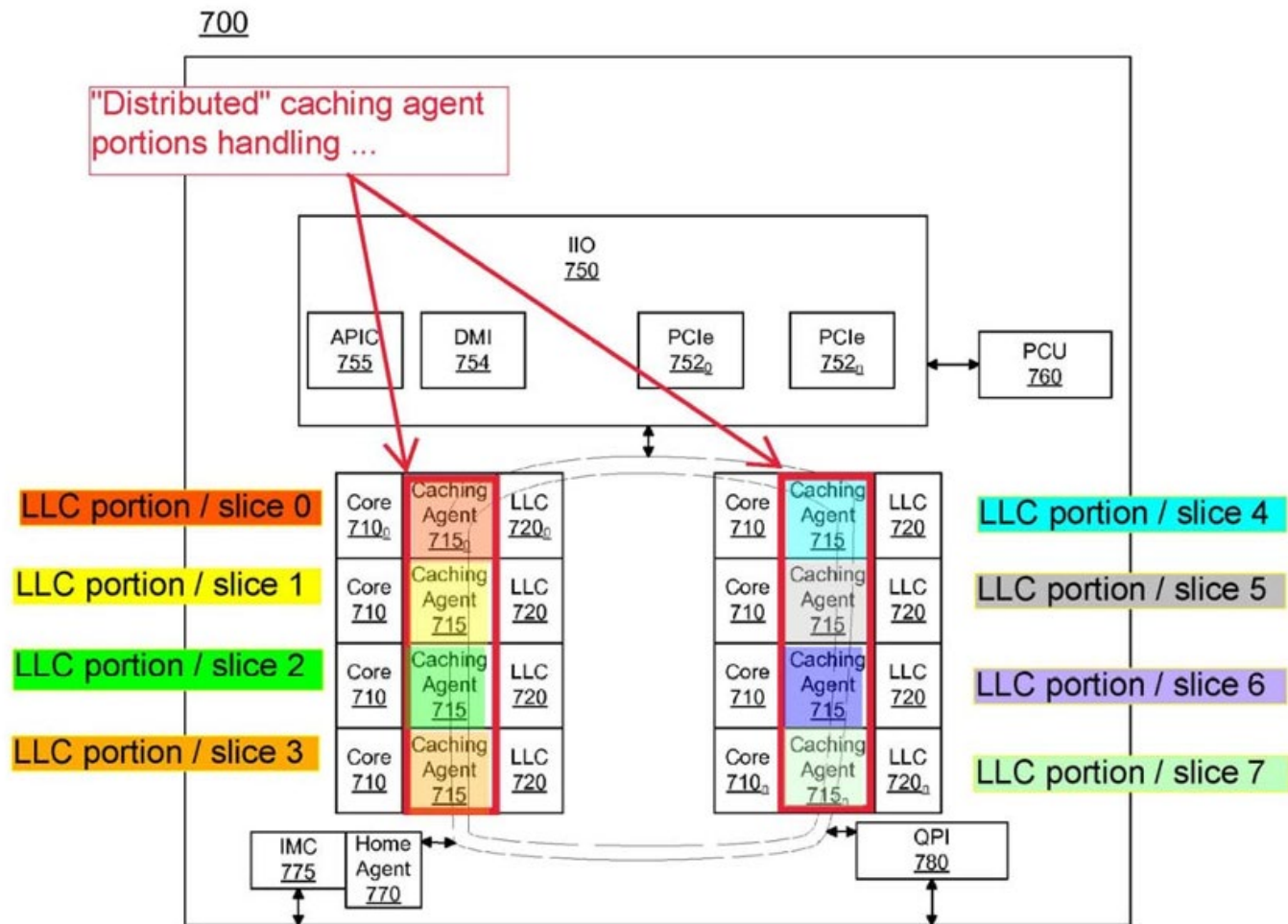


FIG. 6

Overview of Agarwal

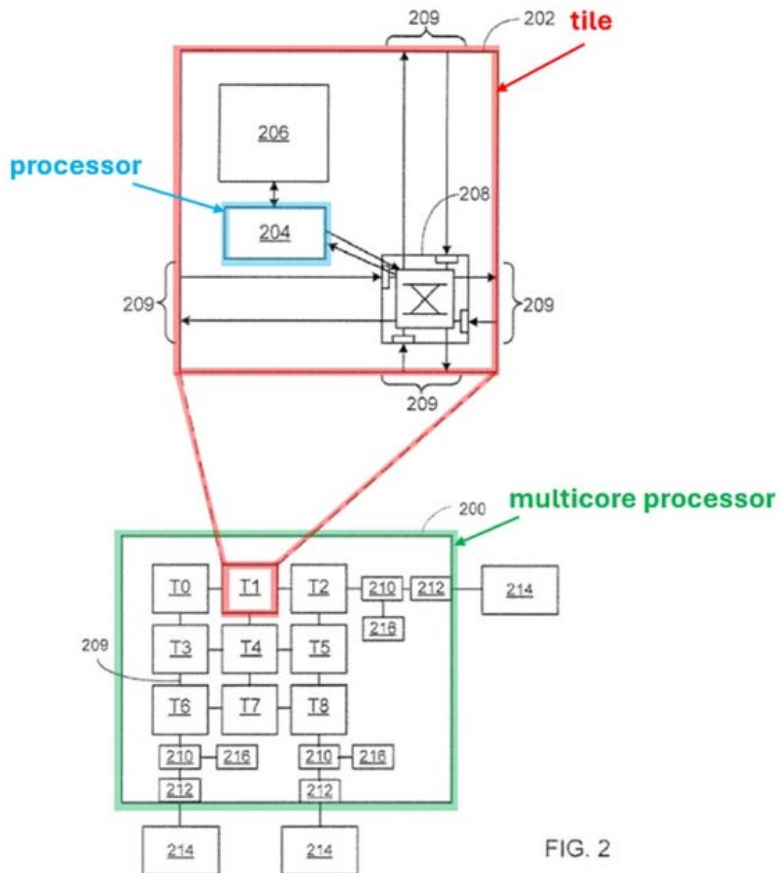


FIG. 2

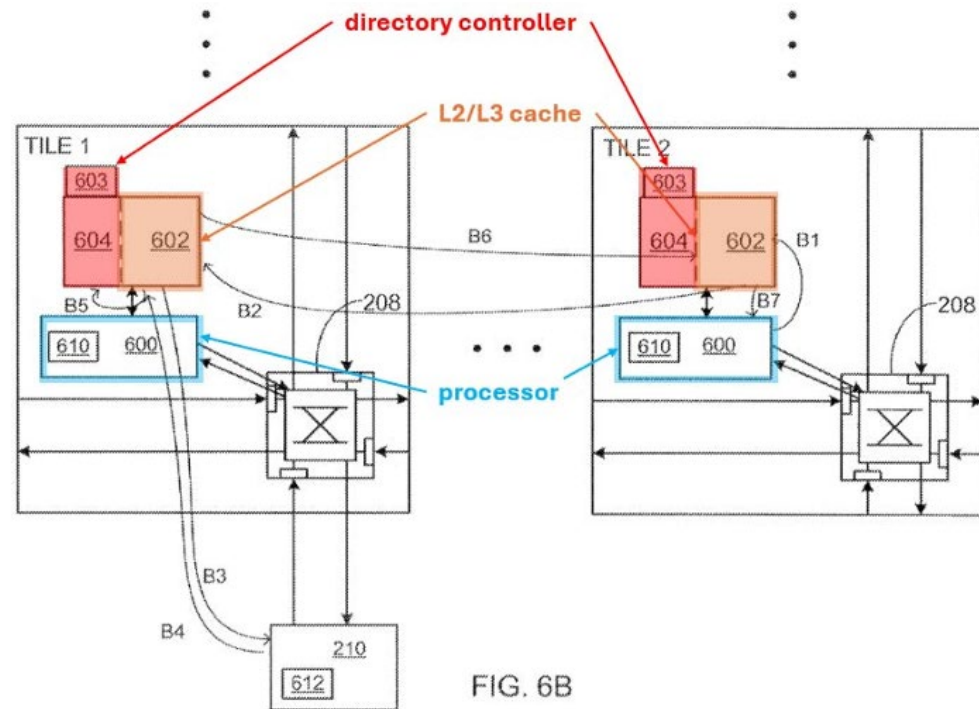


FIG. 6B

Dispute 1: Claim 1 Whether Agarwal discloses “*a single caching agent*”

Claim 1

1. *An apparatus comprising:
a multicore processor including 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, **the caching
agent a single caching agent for the multi core processor
and including a plurality of distributed portions each
associated with a corresponding one of the plurality of
cores.***

PO's Arguments are Not Factually Supported

- “The single caching agent should be understood as a logically unified entity, that is, a cohesive caching agent where all distributed portions are interconnected, consistent, and function together in a rational or systematic manner, not a collection of independent caching agents that merely communicate with one another.” Ex. 2006, ¶ 33 (citing Ex. 1001, 2:18-20, 58-60; 6:25-29; 7:5-7).
- PO repeatedly asserts the claims require “centralized control.”

POR 6, 12, 18.

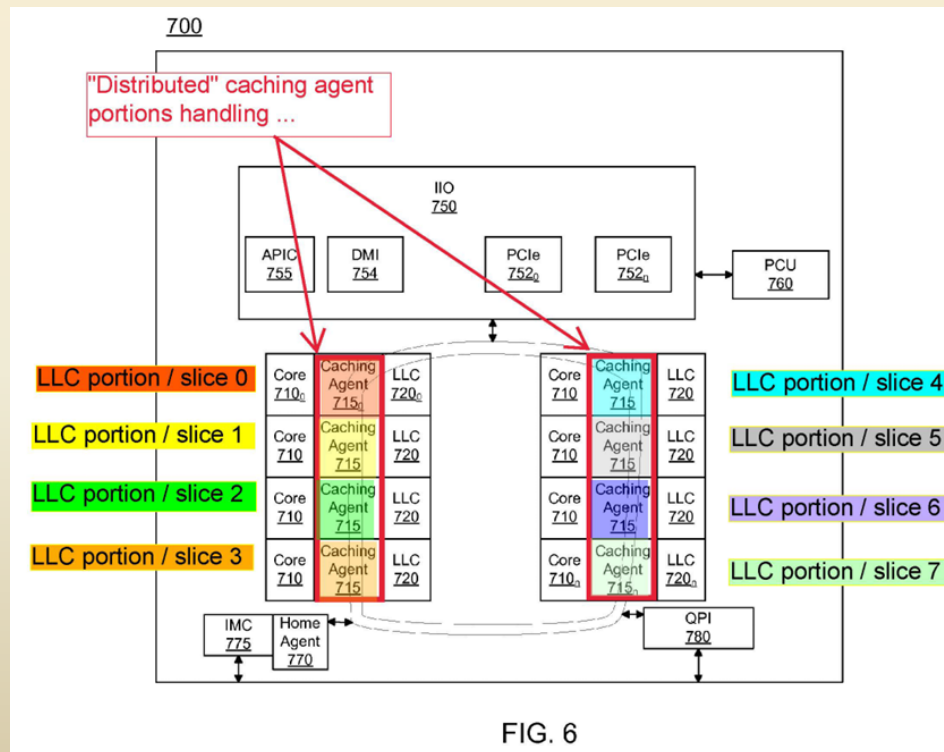
The Claims Recite A Distributed Cache

- The claims recite a “shared cache” among cores for which “cache coherency operations” are performed by the distributed caching agent portions.

Pet. Reply 2-3.

The Specification of the '228 Discloses a Distributed Cache

- Figure 6 of the '228 Patent depicts the claimed “single caching agent” with **distributed** portions, and cores and distributed slices / partitions of the shared cache / LLCs (memory regions) all associated with each other.



Agarwal Discloses Cache Coherency

- Agarwal Discloses:
 - “Cache coherence is a useful mechanism in multiple processor systems” because “[w]hen multiple processors cache a given shared memory location, a cache coherence problem may arise because a copy of the same memory location exists in multiple caches.” EX-1005 1:34-45.
 - “Directory based protocols are one way of maintaining cache coherence” using a “directory state and directory controllers.” EX-1005 1:46-2:42.
 - “Proxy cache coherence” protocol (“PCC”) teaches the directory controller and directory storage can be distributed across “tiles” (or “nodes”) in a tiled multicore processor. EX-1005 11:22-26; Ex-1003 ¶¶68-69.

Pet. 7.

Agarwal Discloses Cache Coherency Contemplated by the '228 Patent

- Agarwal discloses distributed directory coherence protocols that are contemplated by the '228 Patent:
 - Ex-1001, 4:1-4 (“Conventional cache coherency protocol operations and compliance with ordering rules can be performed when accessing this data to maintain coherency.”)
 - 3:61-63 (“data can be stored coherently in accordance with a given cache coherency protocol.”).

Pet. Reply 7-8.

PO's Arguments Have Been Previously Considered and Rejected by the PTAB

- In IPR2023-01344 the Board considered and rejected PO's arguments:

The claim term is “single caching agent,” not a single cache controller, and the “single caching agent” “includes a plurality of distributed portions associated with a corresponding one of the plurality of cores” see Ex. 1001, 3:6–10 (a single caching agent can be distributed so each different portion of the caching agent is associated with a corresponding core and last level cache). FWD at 27-28.

Pet. Reply 10.

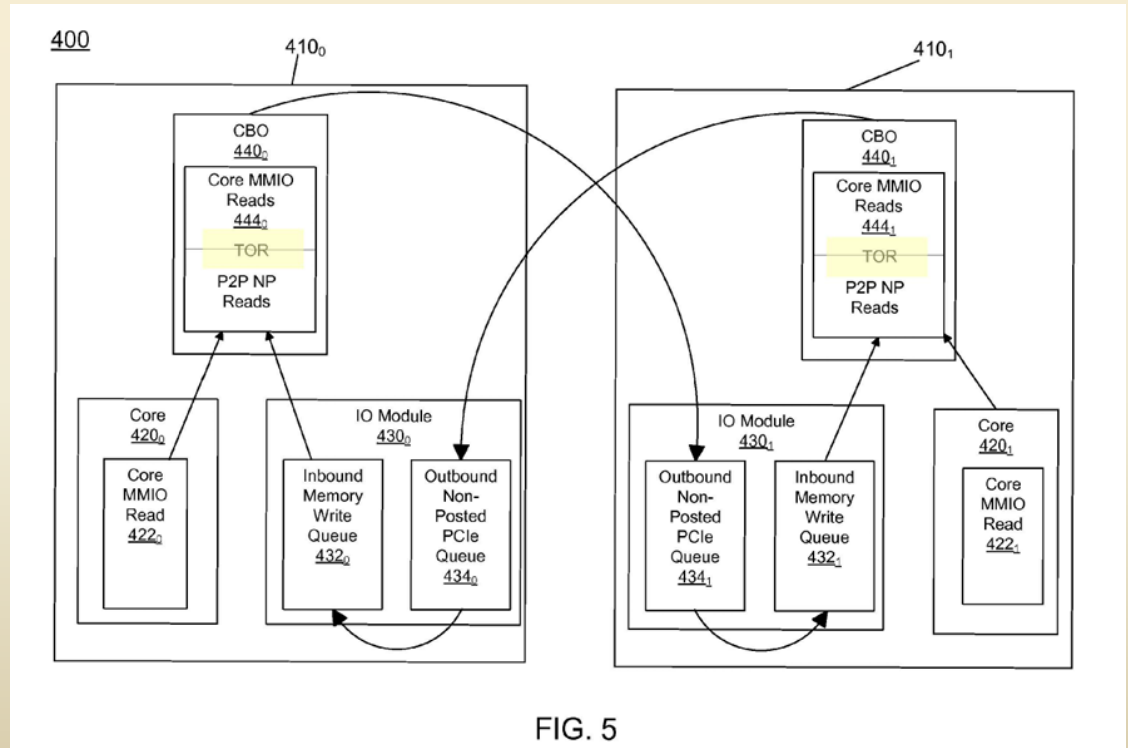
Dispute 2: Claim 2- Whether Agarwal discloses “*a table of requests....*”

Claim 2

*2. The apparatus of claim 1, wherein the caching agent includes a **table of requests** to store pending requests for the plurality of cores and the IIO module.*

The '228 Patent Shows Each Caching Agent Has its Own TOR

- PO alleges “Figure 2 shows the TOR is a single, unified table within the caching agent.”
POR 22.
 - However, Figure 2 of the '228 Patent shows only one caching agent amongst a plurality of caching agents, and Figure 5 makes clear that each caching agent has its own TOR:



Dispute 3: Claim 6- Whether Agarwal discloses “*wherein the caching agent is to receive an allocation from the IIO module...*”

Claim 6

6. *The apparatus of claim 1, wherein the caching agent is to receive an allocation transaction from the IIO module, the allocation transaction received in the IIO module from an IO device coupled to the multicore processor.*

Agarwal Shows Transactions Initiated by External I/O Device

- PO asserts that Agarwal's "cache coherence case" concerns core-initiated direct memory access (DMA) transactions, not transactions initiated by external IO device. POR 28.
 - However, Agarwal discloses "An I/O module is also able to ... store data directly into the cache of a given tile...." EX-1005 28:23-32.
 - Agarwal states that the **IO devices** are using the "cache coherence case." The situation that occurs when the I/O module 912 writes directly into a tile is case 2: the "cache coherence case." EX-1005 28:31-32.
 - Agarwal states "a[n] I/O module is also able to process a DMA transaction or otherwise store data directly into the cache of a given tile....The situation that occurs when the **I/O module** 912 writes directly into a tile is case 2: the 'cache coherence case.'" EX-1005 28:23-32.
- PO argues that Agarwal does not disclose Claim 11 using the same arguments for Claim 1 and 6. POR 29-32.

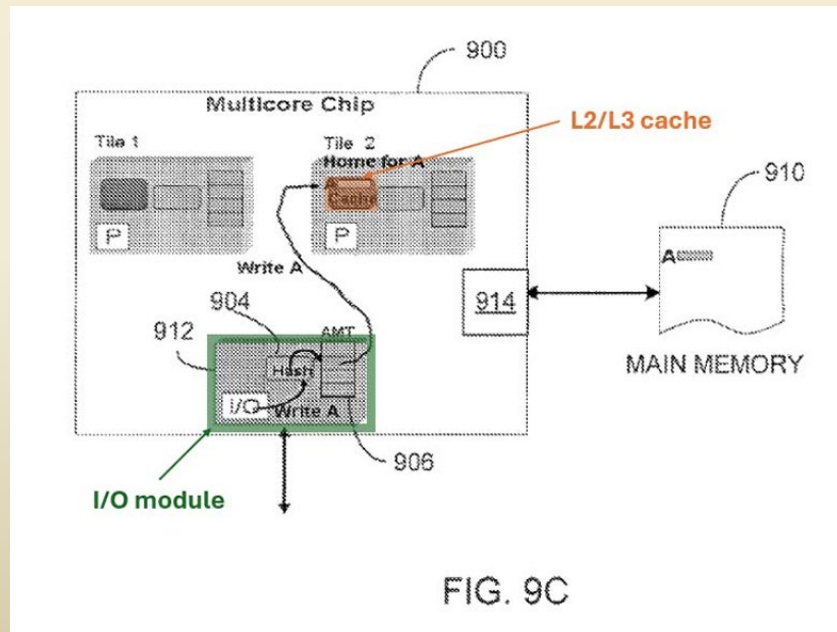
Dispute 4: Claim 14- Whether Agarwal discloses “*directly storing data ...with communication with the system memory.*”

Claim 14

14. *The method of claim 12, further comprising directly storing the data of the allocation transaction into the shared cache memory without communicating with the system memory.*

Agarwal Discloses A Direct Memory Transaction to Cache

- PO asserts that Agarwal is silent. POR 33.
 - However, Figure 9C discloses a “cache coherence case” where the directory controller and directory storage do not initiate a memory transaction to a main memory (i.e., “system memory”). EX-1005 29:5-30:11.



Pet. Reply 22.

Dispute 5: Ground 2- Whether Sinharoy discloses Claim 1

Overview of Sinharoy

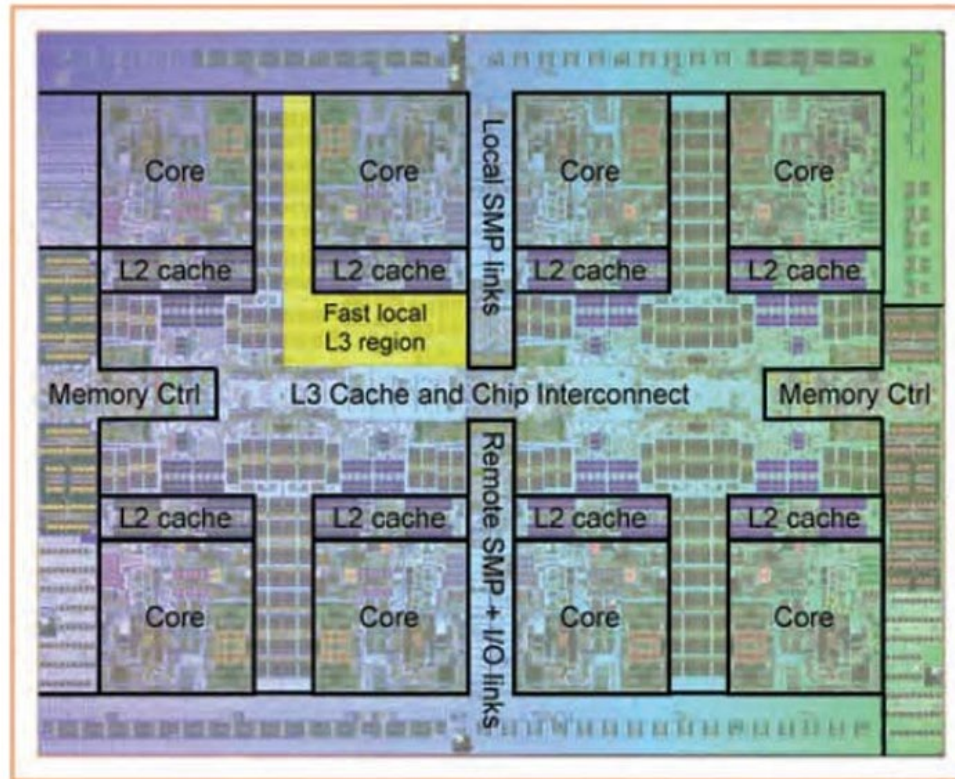


Figure 1

Die photo of the IBM POWER7 chip.

Overview of Sinhharoy

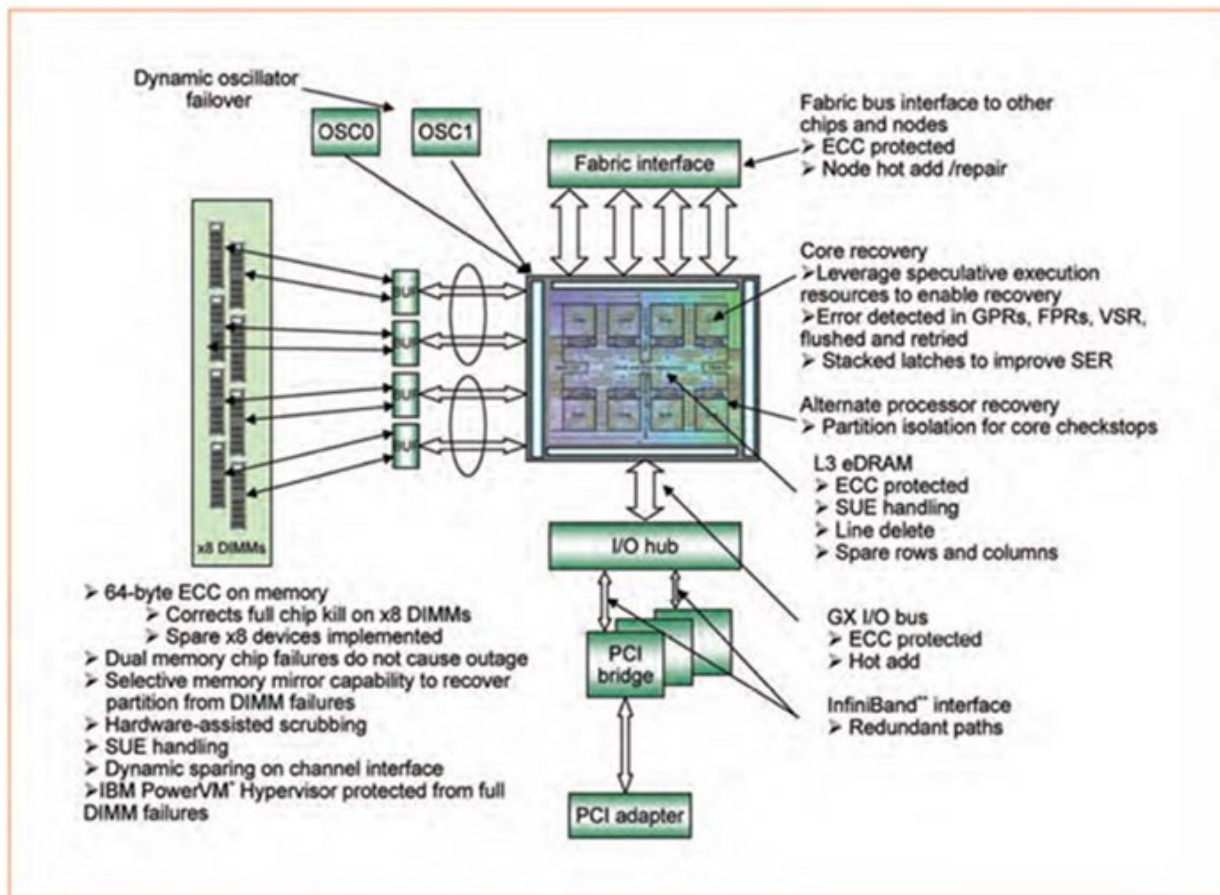


Figure 14

IBM POWER7 reliability and availability features.

Overview of Sinharoy

'228 Patent,
Fig. 6

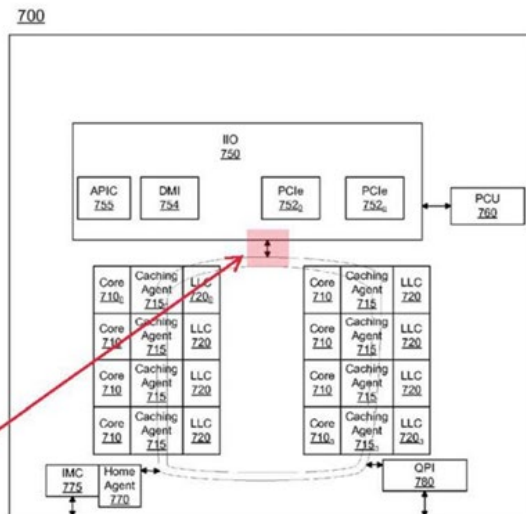
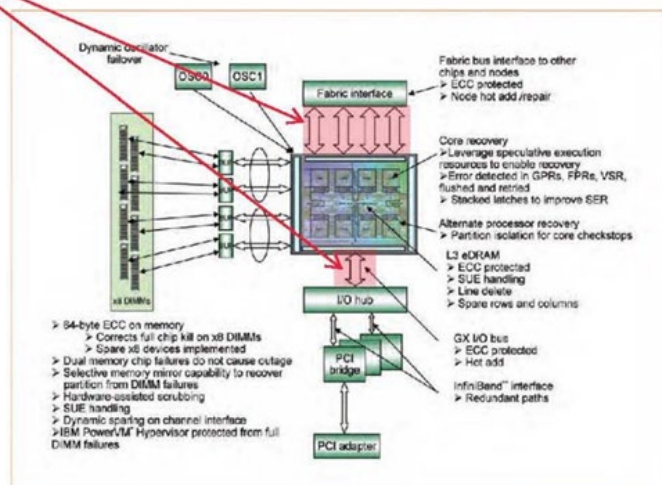


FIG. 6

I/O interface
to cache is a
single access

Sinharoy,
Fig. 14



Overview of Sinharoy

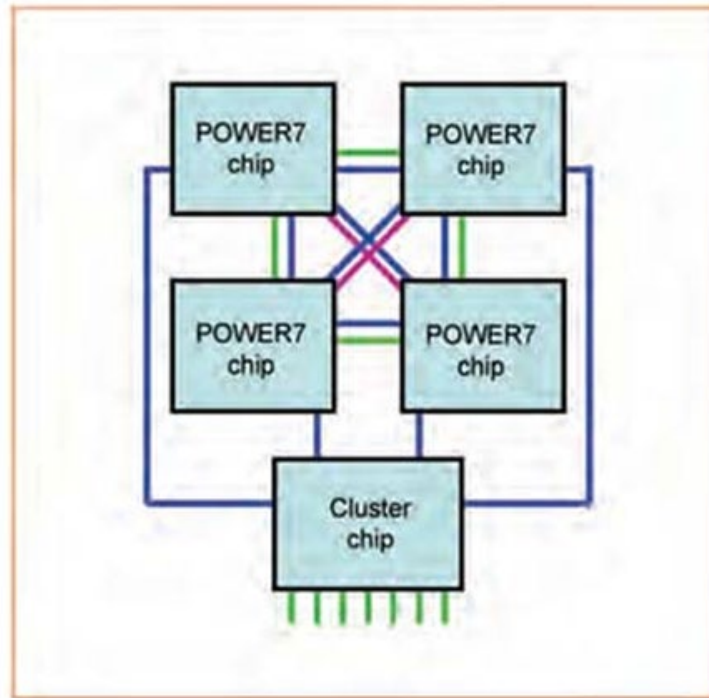


Figure 13

Cluster system first-level coherent nodal topology.

PO's Same Arguments Were Rejected in IPR2023-01344

- Claim 1 is substantially identical to Claim 1 in child '895 Patent
- Claim 2 is substantially identical to Claim 3 in child '895 Patent
- Claim 3 is substantially identical to Claim 4 in child '895 Patent
- Claim 4 is substantially identical to Claim 5 in child '895 Patent
- Claim 6 recites “allocation transaction” recited Claim 1 in child '895 Patent

Dispute 6: Ground 2- Whether Sistla discloses “*wherein the plurality of distributed portions of the caching agent are coupled via a ring interconnect.*”

Overview of Sistla

- Sistla discloses:
 - “a processor” comprising “a memory,”
 - “a plurality of processor cores in communication with the cache memory,”
 - “a scalability agent unit that operates as an interface between an on-die interconnect and both multiple processor cores and memory” where the “on-die interconnect 320 is configured as a ring interconnect.”

Pet. 75-76 (citing EX-1008 Abstract, [0030]-[0031]).

Claim 5 and 19

5. *The apparatus of claim 4, wherein the plurality of distributed portions of the caching agent are coupled via a ring interconnect.*

19. *The system of claim 16, 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, 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, and wherein the plurality of distributed portions of the first caching agent are coupled via a ring interconnect.*

PO's Argument is Not Factually Supported

- PO's asserts that "Sistla's 'ring' couples clusters, not distributed portions of a caching agent" and that the ring interconnect can connect only the caching agent – and nothing else. POR 50-51.
- However, Claims 5 and 19 do not recite any such restriction.
 - The '228 patent's disclosure of "ring interconnect" confirms that it connects all portions of the cores (just like Sistla):

“As seen, a ring interconnect 730 thus couples the cores together, and provides interconnection between the cores, caching agent 715 and I/O module 750. Ex-1001, 6:36-39; Fig. 6.”

The Petition Sets Forth the Motivation to Use Sistla's Ring Interconnect with Sinahoy's Caching Agents

- Using a ring interconnect would have provided a high-performance, scalable interconnect topology that would have been expected improve Sinharoy's bus-style interconnect architecture with a centralized arbitration logic. (Ex-1003 ¶¶514-515)
- A ring interconnect provides a simple, fast, and efficient interconnection between cores and caches in a multicore processor with higher bandwidth. (Marty, 24, 45; Chrysos ¶[0018] Blake, Tbl. 2).
- A POSITA also would have understood that “[r]ings offer simple, distributed arbitration and access methods” (Marty, 48) which had advantages over a bus interconnect, like Sinharoy's. (Ex-1003 ¶515; Marty, 46.)
- A ring provides other advantages over Sinharoy's interconnect including scalability and reduced complexity. (Ex-1003 ¶¶515-516) (Blake, Tbl. 2.) 76 Sistla's teachings would have been a simple substi