

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ARM LTD., MEDIATEK INC,

Petitioners,

v.

DAEDALUS PRIME LLC,

Patent Owner.

Case No.: IPR2025-00207

U.S. Patent No. 8,984,228

PATENT OWNER'S SUR-REPLY

TABLE OF CONTENTS

I. INTRODUCTION	1
II. SCOPE AND PURPOSE OF PATENT OWNER’S SUR-REPLY	2
A. Applicable PTAB Rules Governing Sur-Replies	2
B. Petitioner Bears the Burden Throughout Trial	3
III. PETITIONER’S REPLY FAILS TO ESTABLISH A “SINGLE CACHING AGENT” UNDER AGARWAL	5
A. Petitioner Improperly Equates Distributed Coordination with a Single Caching Agent	5
B. Petitioner’s “Collective Action” Theory is Inconsistent with the Claims and Specification	6
C. Agarwal’s Directory Controllers and Independent and Address-Homed, Not Portions of a Single Agent	8
IV. PETITIONER’S REPLY FAILS TO ESTABLISH A “SINGLE CACHING AGENT” UNDER SINHAROY	10
A. Sinharoy Employs Multiple Independent Coherency Controllers Acting as Peers	10
B. Broadcast Snooping Does Not Create a Single, Logically Unified Caching Agent	11
C. Petitioner’s “Caches Acting in Concert” Theory Misstates Sinharoy’s Architecture	12
V. PETITIONER’S MISCHARACTERIZATION OF “ASSOCIATED WITH” A CORE	14
A. The Claims Require Functional Association, Not Mere Architectural Proximity	14

B. Sinharoy’s L3 Regions are Address-Homed and Accessible by All Cores.....	15
C. Agarwal and Sinharoy Both Lack the Claimed Per-Core Association	16
VI. PETITIONER’S RELIANCE ON IPR2023-01344 IS IMPROPER AND UNAVAILING	17
A. The Prior IPR Involved Different Claims and a Different Evidentiary Record.....	17
B. Petitioner Fails to Establish Any Preclusive or Persuasive Effect....	19
VII. PETITIONER’S REPLY INTRODUCES NEW AND UNSUPPORTED THEORIES REGARDING THE TABLE OF REQUESTS.....	20
A. The Claims Require a Single, Shared Table of Requests in the Caching Agent.....	20
B. Sinharoy’s RC and Write Machines are Execution Resources, Not a TOR	21
C. Agarwal’s and Sinharoy’s Per-Unit Buffers Do Not Satisfy the Claims.....	23
VIII. PETITIONER’S REPLY FAILS TO CURE DEFICIENCIES REGARDING ALLOCATION TRANSACTIONS	24
A. Sinharoy Does Not Disclose IO-Device-Originated Allocation Transactions.....	24
B. Sinharoy Lacks Forwarding by an IIO Module to a Shared Caching Agent	25
C. Agarwal’ “Cache Coherence Case” Does Not Supply the Missing Elements	27
IX. PETITIONER FAILS TO ESTABLISH “DIRECTLY STORING ... WITHOUT COMMUNICATING WITH SYSTEM MEMORY”	28

A. Petitioner Relies on Inference Rather Than Disclosure in Sinharoy	28
B. Sinharoy’s Cache Injection Involves System-Level Coherency Communication	29
X. PETITIONER’S EXPERT TESTIMONY DOES NOT REMEDY THE PETITION’S DEFICIENCIES	30
A. Expert Labels Cannot Transform Peer Controllers Into a Single Caching Agent.....	30
B. Patent Owner’s Expert Testimony Remains Unrebutted on Key Architectural Distinctions.....	31
XI. CONCLUSION.....	32

TABLE OF AUTHORITIES

CASES

<i>Arendi S.A.R.L. v. Apple Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016).....	29
<i>Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.</i> , 821 F.3d 1359 (Fed. Cir. 2016).....	4
<i>Luminex International Co., Ltd. v. Signify Holdings B.V.</i> , IPR2024-00101 (DRP PTAB Feb. 5, 2026)	4
<i>Magnum Oil Tools Int’l, Ltd. v. McClinton Energy Group, LLC</i> , 829 F.3d 1364 (Fed. Cir. 2016).....	4, 31
<i>Novartis AG v. Noven Pharmaceuticals Inc.</i> , 853 F.3d 1289 (Fed. Cir. 2017).....	18
<i>SAS Inst. Inc. v. Iancu</i> , 138 S. Ct. 1348 (2018).....	3

OTHER AUTHORITIES

Patent Trial and Appeal Board Consolidated Trial Practice Guide (Nov. 2019)	2
--	---

I. INTRODUCTION

Patent Owner submits this Sur-Reply pursuant to 37 C.F.R. § 42.23(b) and the Board’s guidance regarding the limited and focused role of sur-replies in inter partes review proceedings.

Petitioner’s Reply confirms, rather than cures, the fundamental shortcomings of the Petition. Confronted with the architectural gaps identified in Patent Owner’s Response, the Reply does not point to new disclosures in the asserted prior art. Instead, it relies on re-labeling, inference, and expanded theories introduced for the first time at the reply stage.

As explained below, neither Agarwal nor Sinharoy discloses or renders obvious the claims as written. Both references describe systems built around multiple independent coherency controllers operating under protocol rules—not a single caching agent with distributed portions associated with corresponding cores. Petitioner’s attempt to equate distributed coordination, broadcast snooping, or components “acting in concert” with a single caching agent is inconsistent with the claim language and the specification and would effectively read the word “single” out of the claims.

The Reply similarly fails to establish the remaining disputed limitations. Petitioner's new theories regarding a purported table of requests, allocation transactions, and direct storage without communicating with system memory are unsupported by the disclosures of Agarwal or Sinharoy and rest on inference rather than teaching. Reliance on a prior IPR involving different claims and a different record does not relieve Petitioner of its burden here.

In short, the Reply underscores that Petitioner cannot meet its burden through characterization alone. Absent disclosures showing the claimed architecture and transaction semantics, Petitioner has not established unpatentability by a preponderance of the evidence, and the Petition should be denied.

II. SCOPE AND PURPOSE OF PATENT OWNER'S SUR-REPLY

A. Applicable PTAB Rules Governing Sur-Replies

The Board has made clear that the purpose of a sur-reply is to ensure a fair opportunity to respond to reply-stage argument, particularly where the reply introduces new theories, materially reframes prior arguments, or mischaracterizes the record. *See, e.g.*, PTAB Consol. Trial Practice Guide (Nov. 2019) at 73–74 (explaining that sur-replies are intended to address

reply arguments and to assist the Board in evaluating issues raised late in trial).

Consistent with these limits, Patent Owner's Sur-Reply is confined to addressing arguments and characterizations advanced or materially expanded for the first time in Petitioner's Reply, including revised theories concerning Agarwal and Sinharoy, new attempts to rely on prior IPR outcomes, and newly articulated interpretations of claim limitations such as the "single caching agent," the "table of requests," and allocation-transaction semantics. Patent Owner does not introduce new evidence and does not seek to relitigate issues fully presented in its Patent Owner Response.

B. Petitioner Bears the Burden Throughout Trial

It is well established that the burden of persuasion remains with the petitioner at all times in an inter partes review. The Supreme Court has made clear that the petitioner bears the burden of proving that the patent is invalid. *SAS Inst. Inc. v. Iancu*, 138 S. Ct. 1348, 1358 (2018). That burden does not shift to the patent owner at any stage of the proceeding.

Additionally, the Federal Circuit has emphasized that a petitioner must present a complete and supported case of unpatentability in the petition

itself and may not use a reply to fill gaps or remedy deficiencies. See *Magnum Oil Tools Int’l, Ltd. v. McClinton Energy Group, LLC*, 829 F.3d 1364, 1380–81 (Fed. Cir. 2016) (it is improper to place the burden on the patent owner to disprove unpatentability theories not adequately articulated in the petition); *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1369–70 (Fed. Cir. 2016) (arguments raised for the first time in reply cannot supply missing elements of a prima facie case); *and see Luminex International Co., Ltd. v. Signify Holdings B.V.*, IPR2024-00101, Paper 57, slip op. at 5-6 (DRP PTAB Feb. 5, 2026) (a petitioner’s reply cannot provide new theories after a patent owner has pointed out flaws in its petition).

Accordingly, arguments advanced for the first time or materially altered in Petitioner’s Reply, whether framed as clarifications, extensions of prior positions, or reliance on outcomes in other proceedings, cannot cure a deficient Petition. The Sur-Reply that follows is directed to demonstrating that Petitioner’s Reply does not satisfy its continuing burden to establish unpatentability of the challenged claims by a preponderance of the evidence.

III. PETITIONER’S REPLY FAILS TO ESTABLISH A “SINGLE CACHING AGENT” UNDER AGARWAL

A. Petitioner Improperly Equates Distributed Coordination with a Single Caching Agent

In its Reply, Petitioner attempts to satisfy the “single caching agent” limitation by characterizing Agarwal’s distributed directory-based coherence protocol as a form of distributed coordination that allegedly constitutes a single agent. Reply at 6-9. This reframing does not withstand scrutiny. The claims are not satisfied by a collection of independent controllers that exchange messages under a distributed protocol. They require a single, logically unified caching agent that performs coherency operations for all cores (and the IIO module), even if that agent is physically partitioned into distributed portions. Ex. 2006 at ¶¶ 33, 39–44.

That distinction is material. Distributed coordination among independent controllers, each making coherence decisions for different memory regions, does not create a single caching agent. Rather, it reflects a conventional distributed directory protocol, in which coherence emerges from protocol-driven interactions among multiple agents. *Id.* at ¶¶ 39–44. Nothing in the claims or specification equates protocol-level coordination with the existence of a single caching agent.

In Agarwal, each directory controller independently receives requests, determines coherency actions for its assigned address range, and issues coherency messages without reference to any shared control structure or global decision-making entity. *Id.* at ¶¶ 35-38, 41-43. Unsurprisingly then, Petitioner’s Reply does not identify any disclosure in Agarwal describing a unified caching agent that maintains or enforces cache coherence for the processor as a whole. Instead, Petitioner relies on the fact that Agarwal’s directory controllers exchange messages and collectively achieve coherence. Reply at 7-9. That result, however, is inherent to any directory-based coherence protocol and does not satisfy the claim requirement that coherency operations be performed by a single caching agent.

B. Petitioner’s “Collective Action” Theory is Inconsistent with the Claims and Specification

Petitioner’s Reply asserts that Agarwal’s directory controllers “act collectively” and that this collective behavior satisfies the “single caching agent” limitation. Reply at 6-9. This theory is inconsistent with both the claim language and the intrinsic record.

The claims require **one caching agent** that performs cache coherency operations for both the plurality of cores and the IIO module. While that

caching agent may include distributed portions, the specification makes clear that those portions together *form* the single caching agent and operate as a unified entity. Ex. 2006 at ¶¶ 29–33. The specification contrasts this architecture with systems that rely on multiple caching agents, explaining that the invention avoids the need for separate or independent agents by providing a single caching agent shared by the cores and the IIO module. *Id.*

By contrast, Agarwal distributes coherency responsibility across autonomous directory controllers that operate independently and coordinate only through protocol-defined message exchanges. In other words, Agarwal does not disclose the required structure. Its directory controllers do not function as distributed portions of a single agent. Rather, each directory controller independently determines coherence actions for its assigned memory region. The fact that these controllers follow a common protocol and exchange messages does not transform them into portions of a single caching agent. Protocol compliance is not logical unity. *Id.* at ¶¶ 41-44, 63-66.

Petitioner’s “collective action” theory effectively reads the word “single” out of the claims. Under Petitioner’s approach, any distributed directory-based coherence system would necessarily include a “single

caching agent,” a result that is incompatible with the specification’s express distinction between the claimed architecture and conventional multi-agent coherence systems.

C. Agarwal’s Directory Controllers and Independent and Address-Homed, Not Portions of a Single Agent

Agarwal’s architecture further confirms that it does not disclose the claimed single caching agent. In Agarwal, cache coherence responsibility is assigned on an address-homed basis. For any given memory address, a specific tile is designated as the “home tile,” and the directory controller on that tile alone is responsible for coherence decisions for that address. Ex. 2007 at 52:5 – 54:2, 66:19 – 67:3. That controller performs coherence operations regardless of which core initiated the request. This address-based assignment of coherency authority means no controller maintains, or even has visibility into, coherency state outside its own address range. Ex. 2006 at ¶¶ 35-38, 46-48.

As a result, Agarwal’s directory controllers are not associated with corresponding cores in the manner required by the claims. A directory controller routinely performs coherence operations on behalf of other cores, depending solely on address mapping. This is fundamentally different from

the claimed architecture, in which each distributed portion of the caching agent is associated with a corresponding core as part of a single, unified agent. *Id.* at ¶¶ 45-48.

Moreover, Agarwal's directory controllers do not share a global view of cache state across all memory regions. Each controller maintains state only for its assigned addresses and operates autonomously. System-wide coherence is achieved through message passing between independent controllers, not through a unified caching agent with distributed portions. *Id.* at ¶¶ 41-43, 59-61.

Petitioner's Reply does not identify any disclosure in Agarwal suggesting that these independent, address-homed directory controllers are unified into a single caching agent or operate under centralized or unified control. Absent such a disclosure, Agarwal cannot satisfy the "single caching agent" limitation, regardless of how Petitioner characterizes the protocol-level interactions among controllers.

IV. PETITIONER’S REPLY FAILS TO ESTABLISH A “SINGLE CACHING AGENT” UNDER SINHAROY

A. Sinharoy Employs Multiple Independent Coherency Controllers Acting as Peers

Petitioner’s Reply asserts that Sinharoy discloses the claimed “single caching agent” because multiple cache controllers within the POWER7 architecture cooperate to maintain cache coherence. Reply at 23-24. That assertion mischaracterizes Sinharoy’s architecture. Sinharoy expressly describes a system in which multiple coherency controllers operate as peers on a coherence fabric, each with its own control logic and state. Ex. 2006 at ¶¶ 108-111; *and see* Ex. 2007 at 82:18 – 83:15.

Sinharoy’s POWER7 design expressly favors distributed, peer-level control to improve scalability, rather than consolidating coherency decision-making into a unified controller. In particular, Sinharoy discloses separate controllers associated with different cache levels and cache regions, including multiple L3 cache region controllers. Each L3 region includes its own controller resources and participates in coherence operations independently. Ex. 2007 at 72:1 – 74:20. Coherency is maintained through interactions among these peer controllers, not through a single caching agent that performs coherency operations for the processor as a whole. Ex. 2006 at ¶¶ 108-111, 114-117.

Nothing in Sinharoy describes a unified caching agent—logically or architecturally—that subsumes these controllers into a single entity. To the contrary, Sinharoy emphasizes scalability and performance through distributed control, with multiple controllers operating concurrently rather than through centralized or unified control. The presence of multiple peer controllers is fundamentally inconsistent with the claim requirement of a single caching agent, even if those controllers communicate with one another. *Id.* at ¶¶ 108-111, 115-117.

B. Broadcast Snooping Does Not Create a Single, Logically Unified Caching Agent

Petitioner argues the Board already rejected PO's snoop/peer-controller objections and held that caches act in concert as a group. See Reply at 24. This argument conflates a communication mechanism with agent identity.

Broadcast snooping is a protocol technique by which multiple independent agents observe coherence traffic and respond according to protocol rules. It does not consolidate those agents into a single caching agent. Ex. 2006 at ¶¶ 109-111, 115-117. Under Sinharoy's broadcast-based coherence scheme, each controller independently evaluates transactions and

independently takes action when required. No controller exercises centralized authority over coherence operations for all caches and cores. *Id.*

The claims, however, do not recite a system in which multiple agents independently snoop and react. They recite a single caching agent that performs cache coherency operations for the plurality of cores and the IIO module. Sinharoy's broadcast snooping mechanism distributes responsibility across multiple peer controllers; it does not create a logically unified agent with distributed portions operating as one. *Id.*

Accepting Petitioner's position would render the "single" limitation meaningless, as virtually any snooping-based coherence system would qualify as having a single caching agent simply because multiple controllers observe the same traffic. The claims and specification do not support such a reading.

C. Petitioner's "Caches Acting in Concert" Theory Misstates Sinharoy's Architecture

To the extent the Reply actively asserts that Sinharoy's caches "act in concert" and therefore collectively form the claimed single caching agent, this characterization substitutes outcome for structure. That multiple caches

collectively maintain coherence does not mean they constitute a single caching agent.

In Sinharoy, the L3 cache is physically partitioned into multiple regions, each with its own controller resources, dispatch ports, and internal control logic. These regions are accessible by all cores, and coherence is enforced through peer-to-peer interactions among controllers over the coherence fabric. The architecture is expressly designed so that no single controller maintains global coherence state or directs coherence operations across all regions. *Id.* at ¶¶ 108-111, 114.

The claims, by contrast, require that distributed portions form a single caching agent. *Id.* at ¶ 33. That requirement is not satisfied by multiple autonomous controllers that happen to interact through a protocol. Sinharoy's controllers do not operate as portions of a single agent; they operate as independent agents following common rules. *Id.* at ¶¶ 115-117.

Petitioner's "acting in concert" theory therefore misstates Sinharoy's architecture and improperly collapses a multi-agent coherence system into a single agent by semantic relabeling. The Reply does not identify any disclosure in Sinharoy that would support treating these independent peer controllers as distributed portions of a single caching agent, nor does it

explain how such a transformation would be consistent with Sinharoy's design principles.

**V. PETITIONER'S MISCHARACTERIZATION OF
"ASSOCIATED WITH" A CORE**

**A. The Claims Require Functional Association, Not Mere
Architectural Proximity**

Petitioner's Reply asserts, with respect to Agarwal, that the "associated with" limitation is satisfied by a directory controller's architectural association with a core. Reply at 11–13. Petitioner further rejects Patent Owner's functional interpretation of "associated with" in the context of Sinharoy. Reply at 24–26. Petitioner's positions are inconsistent with the claim language and the specification.

The claims require that each distributed portion of the caching agent be associated with a corresponding one of the plurality of cores. Ex. 2006 at ¶¶ 45-47. Read in context, this association is functional, not merely architectural. The specification explains that the distributed portions together form a single caching agent and that each portion is tied to a corresponding core as part of a unified coherency mechanism. This reflects a design in which responsibility for coherency is partitioned on a per-core basis within a

single agent, not one in which controllers happen to be co-located with cores but operate independently of core identity. *Id.* at ¶¶ 29-31, 45-47.

Functionally associating a caching-agent portion with a corresponding core requires that the portion primarily services, and is responsible for coherency on behalf of, that specific core at runtime. Petitioner's construction collapses this distinction. If mere physical proximity or implementation within a tile were sufficient, the "associated with" limitation would add nothing to the claims. The intrinsic record does not support such a reading, and the Reply does not identify any disclosure that treats architectural proximity alone as satisfying the claimed association.

B. Sinharoy's L3 Regions are Address-Homed and Accessible by All Cores

The Reply's attempt to satisfy the "associated with" limitation under Sinharoy, see Reply at 24-25, 30, further illustrates this error. Sinharoy's L3 cache is physically divided into multiple regions, but those regions are address-homed, not core-homed. Any core may access any L3 region, and coherence responsibility is determined by address mapping, not by which core initiates a request. Ex. 2006 at ¶¶ 118-122.

As a result, no L3 region controller in Sinharoy is functionally associated with a corresponding core. A controller routinely services requests from multiple cores, depending on the accessed address. That architecture is the opposite of the claimed per-core association, which requires that each distributed portion of the caching agent be tied to a specific core as part of a single, unified agent. *Id.*

Petitioner's Reply does not grapple with this distinction. Instead, it relies on the fact that L3 regions exist and participate in coherence. But participation in coherence does not establish per-core association, particularly where the architecture expressly enables all cores to access all regions and assigns control based on address rather than core identity.

C. Agarwal and Sinharoy Both Lack the Claimed Per-Core Association

The same defect is present in Petitioner's Agarwal and Sinharoy theories. In both references, coherence responsibility is assigned independently of core identity. In Agarwal, directory controllers are address-homed and perform coherency operations for whichever core accesses a given address. *Id.* at ¶¶ 35-38. In Sinharoy, L3 region controllers

act as peers and respond to coherence traffic based on address ownership and protocol rules. *Id.* at ¶¶ 109-111, 114.

In neither system is there a one-to-one functional association between a caching-agent portion and a corresponding core, as required by the claims. *Id.* at ¶¶ 45-48, 118-122. Instead, both references describe architectures in which controllers regularly act on behalf of multiple cores. That design choice is incompatible with the claimed architecture, in which per-core association is a defining feature of the distributed portions of a single caching agent.

Petitioner's Reply does not identify any disclosure in either Agarwal or Sinharoy that remedies this mismatch. Recasting address-homed or peer-based controllers as being "associated with" cores by virtue of proximity or participation in coherence improperly broadens the claim language and ignores the functional nature of the required association.

VI. PETITIONER'S RELIANCE ON IPR2023-01344 IS IMPROPER AND UNAVAILING

A. The Prior IPR Involved Different Claims and a Different Evidentiary Record

Petitioner's Reply looks to the decision in IPR2023-01344 as purported support for its arguments in this proceeding. Reply at 1, 10-11, 23-

24. That reliance is misplaced. The prior IPR involved different patent claims, different asserted grounds, and a different evidentiary record. As a result, the outcome of that proceeding cannot substitute for Petitioner's obligation to establish unpatentability here based on the claims and record now before the Board.

Although the '895 patent at issue in IPR2023-01344 is related to the '228 patent, claim language must be analyzed in the context of the specific claims under review and the evidence presented in the particular proceeding. *See, e.g., Novartis AG v. Noven Pharmaceuticals Inc.*, 853 F.3d 1289, 1294 (Fed. Cir. 2017). The Petition in this case challenges different claims and relies on different combinations and theories, including a substantially expanded reliance on Agarwal and Sinharoy. Those differences matter, and Petitioner cannot avoid addressing them by repeatedly citing the result of a prior proceeding.

Moreover, the evidentiary record in this proceeding includes expert testimony and argument directed to the specific architectural distinctions at issue here, particularly with respect to Sinharoy and the "associated with" limitation, that were not before the Board in IPR2023-01344. Petitioner's

Reply does not explain how the prior decision resolves these issues on the present record.

B. Petitioner Fails to Establish Any Preclusive or Persuasive Effect

Petitioner does not, and cannot, establish that IPR2023-01344 has any preclusive effect in this proceeding. Nor does Petitioner meaningfully explain why the reasoning of that decision should be treated as persuasive here, beyond repeatedly asserting that the Board “already rejected” similar arguments.

That assertion overstates the significance of the prior decision. Even where related claims share similar language, the Board evaluates unpatentability on a claim-by-claim basis and on the record presented. Petitioner’s Reply does not identify any holding in IPR2023-01344 that directly resolves the specific issues presented here, including the deficiencies in Petitioner’s Agarwal- and Sinharoy-based theories addressed in Patent Owner’s Response.

Importantly, reliance on a prior final written decision cannot cure defects in a Petition. Arguments raised for the first time in Reply, whether framed as extensions of a prior decision or otherwise, do not relieve

Petitioner of its burden to demonstrate unpatentability by a preponderance of the evidence in this proceeding. To the extent Petitioner seeks to use IPR2023-01344 as a substitute for analysis of the claims and references at issue here, that effort is improper and should be rejected.

VII. PETITIONER’S REPLY INTRODUCES NEW AND UNSUPPORTED THEORIES REGARDING THE TABLE OF REQUESTS

A. The Claims Require a Single, Shared Table of Requests in the Caching Agent

Petitioner’s Reply advances expanded theories regarding the “table of requests” (“TOR”), including arguments based on per-unit buffers and RC/write machines. Reply at 13-18, 27-29. The claims require that the caching agent includes a table of requests to store pending requests for the plurality of cores and the IIO module. Read in light of the specification, this table of requests is a single, shared structure within the caching agent, not a collection of independent buffers dispersed across separate controllers. Ex. 2006 at ¶¶ 74–76, 79–81.

The specification describes the TOR as a centralized mechanism that arbitrates and stores pending requests from multiple sources, including both cores and the IIO module, and that includes dedicated resources for certain

transaction types. This description makes clear that the TOR is part of the caching agent's unified control logic. It is not merely any temporary storage that happens to track outstanding transactions, nor is it replicated independently across multiple agents. *Id.*

As described in the '228 patent, the table of requests operates as a front-end control structure that receives, stores, and arbitrates pending transactions before they are executed, rather than as an execution pipeline that performs coherency actions. *Id.* at ¶¶ 74–76. Petitioner's Reply departs from this requirement by suggesting that any set of distributed buffers or queues that collectively track transactions can satisfy the TOR limitation. That reading improperly broadens the claims and disregards the requirement that the table of requests be included in the caching agent as a shared structure. The Reply does not identify any disclosure in the asserted references describing such a single, shared table of requests.

B. Sinharoy's RC and Write Machines are Execution Resources, Not a TOR

With respect to Sinharoy, Petitioner's Reply newly asserts that the POWER7 architecture's read/claim ("RC") machines and write machines

collectively satisfy the TOR limitation. Reply at 27-29. This theory mischaracterizes Sinharoy's disclosure.

Sinharoy describes RC and write machines as execution resources—state machines that process coherence transactions once they are scheduled. They are not described as a table, buffer, or centralized structure that stores pending requests from multiple sources. Nor are they described as arbitrating or managing the ordering of incoming requests from both cores and IIO modules in the manner required by the claims. Execution resources act on transactions after acceptance and scheduling; they do not function as an ingress structure that stores and arbitrates pending requests from multiple sources. Ex. 2006 at ¶¶ 131-133.

Further, execution resources that carry out coherence operations are conceptually and architecturally distinct from a table of requests that stores and arbitrates pending transactions before execution. Id. at ¶¶ 74-76, 131-133. Petitioner's Reply does not identify any disclosure in Sinharoy indicating that RC or write machines function as a shared ingress structure for pending requests, much less a structure that corresponds to the claimed TOR. Recasting execution pipelines as a table of requests is unsupported by the reference and inconsistent with the claim language.

C. Agarwal's and Sinharoy's Per-Unit Buffers Do Not Satisfy the Claims

Petitioner's Reply further suggests that per-unit buffers in Agarwal and local tracking structures in Sinharoy, such as per-tile buffers and RC/write machines, satisfy the TOR limitation. Reply at 13-18, 27-29. This argument fails for the same reason under both references.

In Agarwal, any buffers used to track outstanding transactions are local to individual directory controllers and operate independently for address-homed regions. Ex. 2006 at ¶¶ 35–38, 77–78. In Sinharoy, any local tracking of transactions is similarly tied to individual controllers or execution units. *Id.* at ¶¶ 131-133. In neither reference is there a disclosure of a single, shared table of requests that stores pending requests for the plurality of cores and the IIO module as part of a unified caching agent.

The claims do not encompass a system in which multiple independent buffers, each associated with a different controller, collectively perform the function of a table of requests. Such an interpretation would eliminate the structural distinction the claims draw between a shared table within the caching agent and the distributed components that perform coherence operations. The Reply does not explain how independent, per-unit buffers

could be combined, without architectural modification, to form the claimed TOR, nor does it identify any teaching or suggestion to do so.

Accordingly, Petitioner’s newly articulated TOR theories do not cure the deficiencies identified in the Petition and remain inconsistent with both the claim language and the disclosures of Agarwal and Sinharoy.

VIII. PETITIONER’S REPLY FAILS TO CURE DEFICIENCIES REGARDING ALLOCATION TRANSACTIONS

A. Sinharoy Does Not Disclose IO-Device-Originated Allocation Transactions

Petitioner’s Reply asserts that Sinharoy discloses the claimed “allocation transaction” because the POWER7 architecture supports cache injection. Reply at 26-27. That assertion does not address the specific transaction flow required by the claims. The claims require an allocation transaction that originates from an IO device, is received by an IIO module, and results in direct storage of data into shared cache memory. The claimed allocation transaction thus requires a specific sequence—IO device origination, receipt by an IIO module, and forwarding to a shared caching agent—that is distinct from internal cache-management operations. Ex. 2006 at ¶¶ 91-95, 99-101.

Sinharoy does not disclose such an IO-device-originated transaction. As described in Sinharoy, cache injection is performed by integrated controllers within the processor complex as part of DMA handling and coherence operations. *Id.* at ¶¶ 108-110. The Reply does not identify any disclosure in Sinharoy in which an external IO device generates a transaction of the type claimed, nor does it explain how Sinharoy's internal cache injection mechanism corresponds to the claimed allocation transaction semantics.

Recharacterizing cache injection as an allocation transaction does not supply the missing elements. The claims are directed to a specific transaction flow and origin, not merely to the end result of data appearing in cache. Sinharoy's disclosure does not satisfy that requirement.

B. Sinharoy Lacks Forwarding by an IIO Module to a Shared Caching Agent

The Reply further fails to address the requirement that the allocation transaction be forwarded by an IIO module to a caching agent shared by the cores and the IIO module. This forwarding step is explicit in the claims and reflects the architecture described in the specification, in which the IIO

module proxies IO-originated transactions through the shared caching agent.

Id. at ¶¶ 29-31, 99-101.

Sinharoy does not disclose such forwarding. That omission reflects Sinharoy's peer-based coherence model, in which IO-related cache activity is handled internally rather than being proxied through a shared caching agent. In Sinharoy, coherence and cache injection are handled by peer controllers operating on a coherence fabric. There is no disclosure of an IIO module receiving an IO-originated allocation transaction and forwarding that transaction to a shared caching agent for processing. Instead, cache injection occurs within the processor's coherence system without the intermediary forwarding step required by the claims. *Id.* at ¶¶ 108-111, 123-127.

The Reply does not identify any mechanism in Sinharoy that performs this claimed forwarding, nor does it explain how Sinharoy's peer-based coherence architecture could be modified to do so without fundamental architectural change. As a result, the Reply does not cure the deficiencies identified in Patent Owner's Response.

C. Agarwal’ “Cache Coherence Case” Does Not Supply the Missing Elements

Petitioner’s Reply also relies on Agarwal’s “cache coherence case” as allegedly supplying the missing allocation-transaction elements. Reply at 19-20. That reliance is misplaced. As addressed in Patent Owner’s Response, Agarwal’s cache coherence case describes how coherence is maintained when data is written into cache, not how an IO-device-originated allocation transaction is generated, received by an IIO module, and forwarded to a shared caching agent. Ex. 2006 at ¶¶ 93-95.

The Reply does not identify any disclosure in Agarwal describing the claimed transaction origin or forwarding sequence. Nor does it explain how Agarwal’s address-homed directory controllers, which independently manage coherence for assigned memory regions, could satisfy the requirement that a shared caching agent receive and process an IO-originated allocation transaction.

Accordingly, neither Sinharoy nor Agarwal supplies the missing allocation-transaction elements, and Petitioner’s Reply does not remedy the deficiencies identified in the Petition.

IX. PETITIONER FAILS TO ESTABLISH “DIRECTLY STORING ... WITHOUT COMMUNICATING WITH SYSTEM MEMORY”

A. Petitioner Relies on Inference Rather Than Disclosure in Sinharoy

Petitioner’s Reply asserts that Sinharoy satisfies the limitation of “directly storing ... without communicating with system memory” based on the existence of cache injection. Reply at 21-23. That assertion relies on inference, not disclosure. The claims require an affirmative showing that data is stored directly into cache without communicating with system memory, not merely that data appears in cache at some point during operation.

Sinharoy does not expressly disclose the absence of system-memory communication during cache injection. Ex. 2006 at ¶¶ 104-107. The Reply does not identify any passage in Sinharoy stating that cache injection occurs without system-level communication, nor does it explain how such an absence would be understood by a person of ordinary skill in the art. Instead, Petitioner assumes that because data is written into cache, system memory must not be involved. That assumption is insufficient.

Limitations of this type must be supported by disclosure or necessary implication, not by silence. *See, e.g., Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d

1355, 1362 (Fed. Cir. 2016). The Reply’s reasoning impermissibly treats the lack of an explicit memory-access description as evidence that no such communication occurs. Sinharoy does not support that inference, and Petitioner’s Reply does not bridge the gap.

B. Sinharoy’s Cache Injection Involves System-Level Coherency Communication

Even aside from Petitioner’s reliance on inference, Sinharoy’s architecture affirmatively undermines Petitioner’s position. Cache injection in Sinharoy is performed within a system-level coherence framework, involving broadcast snooping, directory state, and coordinated coherency actions among peer controllers. Those coherency mechanisms are designed to maintain consistency across caches and memory and necessarily operate at the system level. *Id.* at ¶¶ 108-111, 114. Agarwal likewise relies on directory-based coherence mechanisms that involve system-level communication, *id.* at ¶¶ 35-38, 59-61, and Petitioner does not contend that Agarwal avoids such communication.

The claims do not merely require that data be stored in cache; they require that such storage occur without communicating with system memory. *Id.* at ¶¶ 102-107. System-level coherency communication,

particularly in a broadcast-based architecture like Sinharoy's, is fundamentally different from the absence of memory communication contemplated by the claims. *Id.* at ¶¶ 104-107, 108-111. The Reply does not explain how Sinharoy's coherence fabric, which enforces consistency across caches and memory, could operate without system-level communication during cache injection.

Accordingly, Petitioner has not established that Sinharoy discloses or renders obvious the claimed "directly storing ... without communicating with system memory" limitation. The Reply does not cure the deficiencies identified in the Petition and does not meet Petitioner's burden on this element.

X. PETITIONER'S EXPERT TESTIMONY DOES NOT REMEDY THE PETITION'S DEFICIENCIES

A. Expert Labels Cannot Transform Peer Controllers Into a Single Caching Agent

Petitioner's Reply relies heavily on expert characterizations, particularly references to components "acting collectively," to argue that the asserted prior art discloses a "single caching agent." Reply at 6-9. Such labels do not supply the missing claim elements.

An expert's use of descriptive terminology cannot substitute for disclosure in the prior art itself. *See, e.g., Magnum Oil Tools Int'l*, 829 F.3d at 1380. Whether a system includes a "single caching agent" is a question of what the reference teaches, not how an expert chooses to describe the outcome of protocol-driven interactions. As explained above, both Agarwal and Sinharoy disclose architectures in which multiple independent controllers operate as peers under coherence protocols. Describing those controllers as acting "collectively" does not transform them into distributed portions of a single caching agent.

The Reply does not identify any expert testimony explaining how independent, address-homed or peer-based controllers are unified into a single logical agent, nor does it reconcile such a theory with the express architectural disclosures of the references. To the extent the expert testimony merely restates Petitioner's legal conclusions using different terminology, it does not remedy the Petition's deficiencies.

B. Patent Owner's Expert Testimony Remains Unrebutted on Key Architectural Distinctions

Patent Owner's expert testimony identified specific architectural distinctions that Petitioner's Reply does not rebut. These include, among

other things, the absence of a unified caching agent in Agarwal and Sinharoy, the address-homed and peer-based nature of their coherence controllers, the lack of per-core functional association, and the absence of a single, shared table of requests within a caching agent.

Petitioner's Reply does not meaningfully address these points. Instead, it relies on generalized statements that coherence is maintained or that components interact under protocol rules. That level of response does not rebut expert testimony grounded in the actual structure and operation of the disclosed systems. Nor does it explain how the asserted references could be modified—without fundamental architectural change—to meet the claimed limitations.

Accordingly, Petitioner's expert testimony does not cure the Petition's failures. The Reply does not undermine Patent Owner's expert analysis of the asserted references, and it does not supply the missing disclosures required to establish unpatentability by a preponderance of the evidence.

XI. CONCLUSION

For the reasons set forth above, Petitioner's Reply does not cure the deficiencies identified in the Petition or meaningfully rebut Patent Owner's

Response. Accordingly, the Board should deny the relief requested in the Petition.

Respectfully submitted,

Dated: April 22, 2026

Ascenda Law Group, PC
2150 N First St., Suite 420
San Jose, CA 95131
Tel: 866-877-4883
Email: tarek.fahmi@ascendalaw.com

/Tarek N. Fahmi/

Tarek N. Fahmi
Reg. No. 41,402

WORD-COUNT CERTIFICATE

I certify that the foregoing document complies with the type-volume limitation of 37 C.F.R. § 42.24(a) and (b) and contains 5,582 words (including words in drawings created by the Patent Owner or as annotations added by the Patent Owner to images reproduced in this document) based on the word count indicated by Microsoft® Word for Mac v. 16.107.2, excluding those portions exempted by the rule.

/Tarek N. Fahmi/

Tarek N. Fahmi

Reg. No. 41,402

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of the foregoing:

PATENT OWNER'S SUR-REPLY

was served on 4/22/2026, by filing this document through the PTAB P-TACTS System as well as by delivering a copy via email directed to the attorneys of record for the Petitioner at the following addresses:

Kevin Anderson
Patrick D. McPherson
DUANE MORRIS LLP
901 New York Ave. NW
Suite 700 East
Washington, D.C. 20001

Email:
KPAAnderson@duanemorris.com
PDMcPherson@duanemorris.com

Cory C. Bell
J. Preston Long
Alexander M. Boyer

Finnegan, Henderson, Farabow,
Garrett & Dunner, LLP
Two Seaport Lane, 6th Floor
Boston, MA 02210-2001

Email:
cory.bell@finnegan.com
jp.long@finnegan.com
alexander.boyer@finnegan.com

The parties have agreed to electronic service in this proceeding.

Respectfully submitted,

Dated: April 22, 2026

Ascenda Law Group, PC
2150 N First St., Suite 420
San Jose, CA 95131
Tel: 866-877-4883
Email: patents@ascendallaw.com

/Tarek N. Fahmi/
Tarek N. Fahmi
Reg. No. 41,402