

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ADVANCED MICRO DEVICES, INC. and PENSANDO SYSTEMS, INC.,

Petitioners,

v.

CONCURRENT VENTURES, LLC and XTREAMEDGE, INC.,

Patent Owners.

Case IPR2025-00486

U.S. Patent 10,873,753

PATENT OWNERS' RESPONSE

TABLE OF CONTENTS

I. INTRODUCTION.....	1
II. BACKGROUND.....	4
A. Overview of the ‘753 Patent	4
B. The Petition’s Relied-Upon Prior Art References.....	7
1. U.S. Patent Application Publication No. 2013/0346667 A1 (“Stroud”) (EX1004)	7
2. U.S. Patent No. 9,148,389 (“DaSilva”) (EX1005)	8
3. “Memory Systems and Pipelined Processors” (“Cragon”) (EX1006)	8
4. U.S. Patent Application Publication No. 2002/0087821 (“Saulsbury”) (EX1007)	8
C. Patent Owners’ Preliminary Response and the Board’s ID.	9
D. The IPR Petition’s Alleged Unpatentability Grounds.....	11
III. PRIORITY DATE OF THE ’753 PATENT	11
IV. LEVEL OF ORDINARY SKILL	12
V. RELATED MATTERS.....	12
VI. CLAIM CONSTRUCTION	13
A. Petitioners’ Identified Constructions.....	13
B. Patent Owners’ Constructions.....	13
1. “a module identifier”	14
2. “sandbox”	16
3. “selecting a memory superposition” and “selectable arrangement”	17
VII. PETITIONERS HAVE NOT SHOWN BY A PREPONDERANCE OF THE EVIDENCE THAT THE CHALLENGED CLAIMS ARE UNPATENTABLE AS OBVIOUS.....	19
A. Petitioners Fail to Prove their Combination of Stroud and DaSilva (Ground 1) Discloses or Renders Obvious Any Challenged Claim.	19
1. Stroud Does Not Disclose or Render Obvious a “Module Identifier” or That “Each Module . . . Can Be Addressed Through The Header” As Required By Claims 1 and 13.	19
2. Claims 1–3, 13, 14 are Patentable over Stroud and DaSilva.....	30

3. Petitioners Fail to Prove that Stroud Discloses Each Limitation of Claim 3.	30
a. Stroud Does Not Disclose “At Least One Processor Allocatable for Offloading Hardware Tasks to Software.”	31
b. Stroud Does Not Disclose a Processor “Executing in Cooperation with a Sandbox of One of the Plurality of Swappable Pods or Cards.”	39
B. Ground 2 fails because Petitioners have not shown that any of its multiple proposed combinations disclose or render or render obvious the memory limitations of [7d].....	45
1. Neither Stroud nor Cragon teaches, suggests, or renders obvious a single card with each of the four required memory arrangements.	46
2. The prior art does not teach or suggest that the memory superpositions can be <i>selected</i>	54
3. Claims 9-10 are Patentable over Stroud and Cragon.	58
C. Ground 3 Does Not Remedy the Defects in Ground 2.	58
1. Saulsbury does not teach, alone or in combination, the combination of four distinct memory arrangements on a single card required by limitation [7d].	59
2. Saulsbury does not teach, alone or in combination, the claimed “selectable arrangement.”	61
3. Claims 9-10 are Patentable over Stroud, Cragon, and Saulsbury.....	62
D. Ground 4 Does Not Remedy the Defects in Grounds 1, 2, or 3.	62
VIII. CONCLUSION.....	63
CLAIM APPENDIX	66

PATENT OWNERS' EXHIBIT LIST

Exhibit	Description
2001	Text Order Denying AMD/Pensando's Motion to Stay Case (Judge Robert Pitman) in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2002	First Amended District Court Scheduling Order (Dkt. 77) in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv- 00335 (W.D. Tex. 2024)
2003	Webpage printouts of Docket Navigator Time to Trial regarding the District Court for the Western District of Texas and Judge Alan D. Albright
2004	Webpage printout of "AMD Reports Fourth Quarter and Full Year 2024 Financial Results" dated March 4, 2025 (https://ir.amd.com/news-events/press-releases/detail/1236/amd-reports-fourth-quarter-and-full-year-2024-financial-results)
2005	AMD/Pensando's Invalidity Contentions (Cover Pleading) in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2006	AMD/Pensando's Opening Claim Construction Brief in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2007	Declaration of Dr. R. Jacob Baker submitted by AMC/Pensando in support of AMD/Pensando's Opening Claim Construction Brief in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2008	Declaration of Darrell Long submitted by AMC/Pensando in support of AMD/Pensando's Opening Claim Construction Brief in <i>Concurrent Ventures, LLC and XstreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2009	Declaration of Bill Lin submitted by AMC/Pensando in support of AMD/Pensando's Opening Claim Construction Brief

	in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2010	Declaration of Kevin Jeffay submitted by AMC/Pensando in support of AMD/Pensando's Opening Claim Construction Brief in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2011	CV/XE's Responsive Claim Construction Brief in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2012	Declaration of Dr. Paul Min submitted by CV/XE in support of CV/XE's Responsive Claim Construction Brief in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2013	AMD/Pensando's Reply Claim Construction Brief in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2014	CV/XE's Objection to the Report and Recommendation of the U.S. Magistrate Judge in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2015	AMD/Pensando's Motion for Reconsideration of Venue Order in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2016	Order denying AMD/Pensando's Motion for Reconsideration of Venue Order in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2017	September 11, 2025 Scheduling Order in <i>Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.</i> , No. 1-24-cv-00335 (W.D. Tex. 2024)
2018	Petitioners' Request For Rehearing Of Decision Denying Institution Of <i>Inter Partes</i> Review in IPR2025-00223, Paper 10.

2020	Petitioners' Opposition Brief To Patent Owners' Request For Discretionary Denial Of Institution Of Inter Partes Review in IPR2025-00486, Paper 8.
2020	Declaration of Zachary L. Jacobs in Support of Motion for Admission <i>Pro Hac Vice</i>

New Exhibits	Description
2021	Declaration of Dr. Paul S. Min in Support of Patent Owners' Response
2022	Curriculum Vitae of Dr. Paul S. Min
2023	Datasheet of ALTERA STRATIX
2024	Deposition of Dr. Alpert in IPR2025-00486, conducted December 12, 2025
2025	July 31, 2025 Memorandum
2026	Defendants Advanced Micro Devices, Inc.'s and Pensando Systems, Inc.'s Identification of Claim Terms for Construction

Concurrent Ventures, LLC and XstreamEdge, Inc. (“Patent Owners”) respectfully submit this Patent Owners’ Response (“POR”) to the Petition for *Inter Partes* Review (“IPR”) of U.S. Patent No. 10,873,753 (the “’753 patent”) filed by Advanced Micro Devices, Inc. and Pensando Systems, Inc. (“Petitioners”), and to the Board’s October 3, 2025 Institution Decision (“ID”) (Paper 11).

I. INTRODUCTION

The Petition’s grounds fail to render obvious the inventions of the ’753 patent. Petitioners rely solely on obviousness arguments, and for every ground, their references fail to teach or suggest one or more of the challenged limitations of the challenged independent claims (claims 1, 7, and 13) and many of the dependent claims (e.g., claims 3, 4, 8, and 15). For each ground, there are elements that are missing wholesale. Petitioners seek to fill these gaps by arguing they would be obvious—but without any teaching or suggestion in the art, and without any reasoned or supported expert opinion. Petitioners’ arguments that missing limitations would be obvious based only on their say so—despite their inability to find even a single reference that teaches or even suggests them—is nothing more than an invitation for the Board to engage in an improper hindsight analysis.

Claims 1 and 13 (addressed in grounds 1 and 4) claim a platform for data flow processing requiring “one or more swappable pods or cards in one or more chassis, coupled through a messaging interface network” that “support[s] messaging-based

communication using packets each having a header with . . . a module identifier . . . so that . . . each module on each board in each chassis can be addressed through the header.” EX1001, cl. 1. Petitioners do not identify a “module identifier” in any of the prior art references. Instead, Petitioners point to an “interface identifier” (*i.e.*, an identifier of interfaces with Management Switch 110) in one of their references, Stroud, and allege that it effectively functions as a module identifier. But Stroud itself is fatal to Petitioners’ argument: Stroud discloses that some modules ***do not*** have an interface with the Management Switch 110, and thus the “interface identifier” does not identify ***each*** module or make it so ***each*** module is addressable through the header. Petitioners do not argue that the claimed “module identifier” would be obvious, and thus ground 1 fails on this basis alone.

However, that is not the only flaw in ground 1. Claim 3 is a dependent claim which requires the platform of claim 1 have “at least one processor allocatable for offloading hardware tasks to software . . . and executing in cooperation with a sandbox of one of the plurality of swappable pods or cards.” *Id.*, cl. 3. Again, Petitioners use the “Stroud” reference to attempt to prove each limitation. But there is nothing in Stroud that discloses either “offloading hardware tasks to software” or that any of the “swappable pods or cards” have “sandboxes.” Worse yet, Petitioners’ theories are internally inconsistent: the claims require a single processor that is

allocatable for both “offloading” and “executing,” yet Petitioners point to different processors for their theories.

Petitioners’ arguments for independent claim 7 and dependent claims 4 and 15 (addressed in grounds 2-4) stray even further from what the claims actually require. Each of those claims require: (1) a single pod or card that has multiple types of memory that can be arranged in each of four claimed memory arrangements; and (2) that the claimed memory arrangements can be selected when the card is being used or configured. Critically, none of the references disclose or suggest a single pod or card with the four claimed memory arrangements. In fact, no reference even discloses a single pod or card with multiple memory arrangements at all. Petitioners’ own expert admits that there are hundreds of possible memory arrangements, with millions of possible combinations of four memory arrangements. Petitioners have presented no reason—beyond hindsight—why a POSITA would choose these four memory arrangements (out of the millions of options) to put on a single card. Going further, none of Petitioners’ references disclose or suggest being able to select between different memory arrangements. Again, Petitioners argue this element would have been obvious, but fail to support that contention with any evidence or reasoned expert opinion. Petitioners have thus failed to show any challenged claim is unpatentable.

II. BACKGROUND

A. Overview of the '753 Patent

The inventors of the '753 patent recognized a problem in the art relating to data flow processing architectures as of the patent's effective filing date: software-based systems offered flexibility and ease of use but were often inferior to hardware-based systems with respect to processing power. EX1001, 1:8–11. Pure hardware-based systems, in contrast, offered superior processing speeds but were not a desirable alternative because “redesigning dedicated hardware each time a new data processing application or data formatting requirement comes up is an expensive proposition.” *Id.*, 1:11–14. Further, “designing, debugging and manufacturing dedicated electronic hardware is time-consuming, especially when revision cycles are taken into account.” *Id.*, 1:14–16.

To address this problem in the art, the inventors of the '753 patent developed an innovative hybrid data-processing platform that offers the processing speed advantage of hardware while retaining much of the flexibility of software-based systems. EX2021, ¶47. Specifically, the inventors conceived of an architecture that utilizes one or more swappable pods or cards that include a portion for user-definable hardware modules or user-definable software modules, with the pods or cards communicatively linked via a messaging interface network. *Id.* The goal of the system was maximum flexibility. *Id.*

One approach, which is reflected in independent claims 1 and 13 and dependent claim 8, is a “messaging-based communication” system in which packets have headers with five specific identifiers: “a chassis identifier, a board identifier, a module identifier, an instance identifier, and a type identifier.” EX1001, cl. 1, 13. These headers are shown below in Figure 6A. *Id.*, Fig. 6A. The claims specify that the headers exist “so that each type of module, each instance of a type of module, and each module on each board in each chassis can be addressed through the header.” *Id.*, cl. 1, 13.

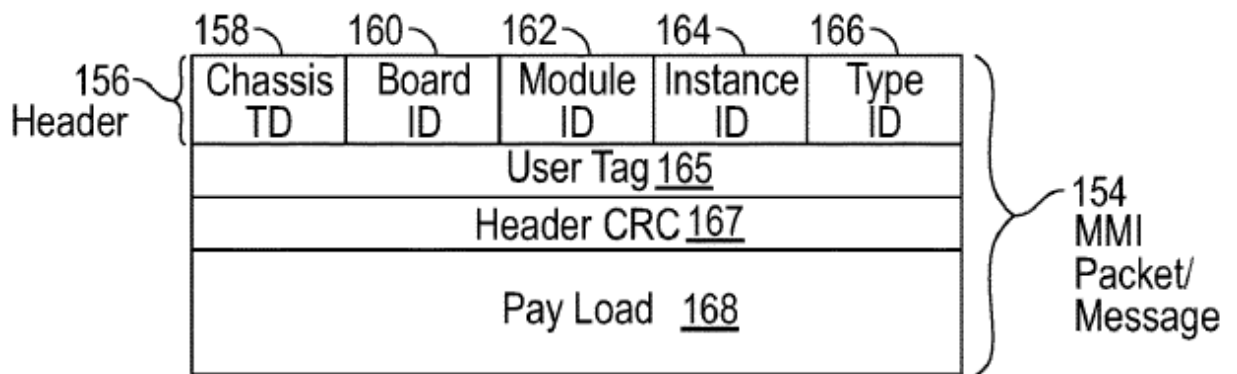


FIG. 6A

Id., Fig. 6A.

As noted above, the goal of the system was maximum flexibility. EX2021, ¶47. To accomplish that, the patented header provides multiple ways to identify each module. *Id.* For example, the module identifier allows each module to be identifiable and addressable through a unique value assigned to that module. *Id.* Additionally, the instance identifier also allowed each module to be identifiable and addressable,

this time through a unique value assigned to that instance out of modules of that type. *Id.* The specification provides an illustrative example of this concept: “[A]n MMI packet 154 could be destined to chassis number two, board number three in that chassis, module number seven on that board, instance number three of that module on that board, and the module being a hardware type.” EX1001, 8:61-65. In this example, the module is the seventh module on the board (because it has a module identifier value of “seven”) and it is also the third hardware module on the board (because it has an instance identifier value of “three” and a type identifier value of “hardware”). *Id.*; *see also* EX2024, 66:4-71:4. In this way, the patented innovation ensures that packets can be routed and addressed with maximum flexibility and accuracy. EX2021, ¶¶39-40.

Another approach, which is reflected in independent claim 7 and dependent claims 4 and 15, is for the claimed card or pod to have multiple types of memory that are arranged in various memory arrangements. As the name conveys, memory arrangements are different ways of arranging memory types for various benefits. The specification explains that “[a]nalogous to the overlapping of waves to form a final observed wave in wave superposition, where each wave contributes to the whole, in this case different types of memory are being overlapped, each with benefits and drawbacks, to form a final memory system which performs optimally in a given video stream or other data flow/data processing task.” EX1001, 7:40-46. The

patented invention is not, however, just to use memory arrangements, but to use four specific memory arrangements and make them all available and selectable on a single pod or card: “multilevel cache, interleaved memory, swapped memory over a specified address range, and memory allocated to specified address ranges according to parameterized task needs.” *Id.*, claims 7, 4, 15, 6:65-8:46. The inventors of the ’753 patent determined that it was particularly advantageous to combine these four memory arrangements on a single pod or card and allow them to be selected. *Id.*; EX2021, ¶51.

B. The Petition’s Relied-Upon Prior Art References

1. U.S. Patent Application Publication No. 2013/0346667 A1 (“Stroud”) (EX1004)

Stroud is a patent application directed to “systems and methods for testing communications networks, services, and devices, e.g., testing the traffic-handling performance and/or security of the network, network accessible devices, cloud services, and data center services.” EX1004, ¶0001. Stroud discusses an increased cyber security threat environment and the importance of network security testing, *id.*, ¶¶2–3, and notes the importance of “content-aware and software-driven packet inspection,” *id.*, ¶5. To address these issues, Stroud proposes a “network testing system 16” that “may be configured to test the performance (e.g., traffic-handling performance) of devices 14, the security of a test system 18 (e.g., from security attacks), or both the performance and security of a test system 18.” *Id.*, ¶81. Stroud

does not discuss or contemplate large-scale data storage or suggest that its teachings would be relevant to data center design. As Petitioners' expert acknowledged, Stroud concerns "testing communications networks, services and devices." EX2024, 34:12-20. Stroud does not concern "building or designing or developing those communication networks themselves" or "building or designing or developing data centers" except for "the testing aspect" within data centers. *Id.* 34:21-35:18.

2. U.S. Patent No. 9,148,389 ("DaSilva") (EX1005)

The Petition relies on DaSilva for its teachings regarding a "chassis ID." Pet., 19 (citing EX1005, Abstract).

3. "Memory Systems and Pipelined Processors" ("Cragon") (EX1006)

Cragon is a 1996 computer textbook that provides an overview of a variety of general concepts and considerations relating to memory systems as of that date. In particular, as Dr. Alpert testified, Cragon discusses "issues that a system designer would face related to the performance and other characteristics of disks." EX2024, 99:16-22. Cragon discusses the advantages and disadvantages of some memory configurations but does not disclose using the configuration together in a selectable arrangement. EX2021, ¶¶90-94.

4. U.S. Patent Application Publication No. 2002/0087821 ("Saulsbury") (EX1007)

Petitioner relies on Saulsbury in Ground 3 to supply the "selectable arrangement" of "types of memory" required by claim limitation [7d]. Pet., 84-85.

As discussed below, however, Saulsbury does not teach any capability to select memory arrangements, much less selecting between the specific memory arrangements required by the claims and does not resolve the defects in the Stroud/Cragon combination.

C. Patent Owners' Preliminary Response and the Board's ID.

Patent Owners raised four arguments in their preliminary response ("POPR"). Paper 7. First, Patent Owners argued that Stroud does not disclose the "chassis identifier," *Id.*, 11-19. Second, Patent Owners argued that Stroud does not disclose the "module identifier" because, even accepting Petitioners' allegations, the alleged module identifier—Stroud's "interface identifier"—only identifies "each processor, NP, and CLD" and does not identify the "modules [that] may comprise multiple CLDs." *Id.*, 20-22. Third, Patent Owners argued that neither Cragon nor Saulsbury disclose the "selectable arrangements." *Id.*, 22-24, 25-26. Fourth, Patent Owners argued that neither Cragon nor Saulsbury discloses the "memory allocated to specified address ranges according to parameterized task needs." *Id.*, 24-26.

Of these, Patent Owners are not disputing the "chassis identifier" (the first argument) or the "memory allocated to specified address ranges according to parameterized task needs" (the fourth argument). Patent Owners are disputing the "module identifier," but based on a different argument than presented in the POPR. Patent Owners are maintaining the third argument—that neither Cragon nor

Saulsbury disclose the “selectable arrangements”—albeit on significantly different bases.

The Board preliminarily rejected Patent Owners’ arguments and instituted trial. Paper 11. The Board based its decision regarding the “module identifier” on Petitioners’ unchallenged assertion that “the identified CLD and processor (e.g., NP) modules [are] distinct devices *with a distinct connection to Ethernet switch 110 interface with a corresponding address for each connected device.*” *Id.*, 17 (emphasis added). Patent Owners now challenge that underlying assertion in this Response and show it is incorrect—the CLDs and NPs do not have “distinct connection to Ethernet switch 110 interface”—undermining the basis of Petitioners’ argument. *See* Section VII.A.1.

The Board based its decision regarding the “selectable arrangement” on its preliminary finding that each of the four claimed memory arrangements were themselves “well-known” and thus could have been selected by a designer. Paper 11, 36-38. Patent Owners now newly argue that the claims require that the selecting occurs not during the design of the platform, but during its configuration or use. *See* Section VI.B.3. Moreover, Patent Owners are newly arguing that the prior art does not render obvious a single card which has all four of the claimed memory arrangements, providing further evidence that the references do not render obvious the further step of making the four claimed memory arrangements individually

selectable. This new argument is particularly relevant in light of Petitioners’ expert’s admission that there are hundreds of memory arrangements—making millions of possible combinations of four memory arrangements.

Patent Owners’ arguments regarding claim 3’s “offloading” and “executing” limitations (Section VII.A.3) and claim 7’s limitation that a single card must have each of the four required memory arrangements (Sections VII.B.1 and VII.C.1) were not raised in the POPR and, correspondingly, were not addressed in the Board’s ID.

D. The IPR Petition’s Alleged Unpatentability Grounds

Grounds	Claims	Statutory Basis	Prior Art
1	1–3, 13, 14	§103	Stroud in view of DaSilva
2	7, 9-10	§103	Stroud in view of Cragon
3	7, 9-10	§103	Stroud in view of Cragon and Saulsbury
4	4, 8, 15	§103	Stroud in View of DaSilva, Cragon, and Saulsbury

III. PRIORITY DATE OF THE ’753 PATENT

Petitioners claim that the ’753 patent has a priority date of July 15, 2018. Patent Owners do not dispute, for purposes of this Response, that all of Petitioners’ prior art pre-dates the priority date of the ’753 patent.

IV. LEVEL OF ORDINARY SKILL

Petitioners identify their proposed level of ordinary skill. Pet., 13. For purposes of this proceeding, Patent Owners do not dispute Petitioners' proposed level of ordinary skill, nor do Patent Owners dispute that Petitioners' expert is a POSITA.

Here, Patent Owners' expert, Dr. Paul Min, is also a POSITA because he has a Ph.D., M.S., and B.S. in Electrical Engineering. EX2021, ¶¶9-10 (citing EX2022). Dr. Min is currently a Senior Professor in the Department of Electrical Systems Engineering at Washington University, St. Louis, and has worked in the field of electrical engineering and data processing, streaming architecture, and data networks for more than thirty years. *Id.*

V. RELATED MATTERS

Patent Owners, CV/XE, are presently asserting the '753 patent in the following district court action: *Concurrent Ventures, LLC and XtreamEdge, Inc. v. Advanced Micro Devices, Inc. and Pensando Systems, Inc.*, No. 4:25-cv-09567-JST (N.D. Cal. 2025)

Petitioners have two other instituted *inter partes* reviews on two other patents asserted in the district court action: IPR2025-00223 and IPR2025-00478.

VI. CLAIM CONSTRUCTION

A. Petitioners' Identified Constructions

Petitioners identify constructions that “were proposed in district court” for four claim terms, but go on to state that “[t]he prior art teaches these terms irrespective of whether these constructions are adopted.” Pet., 14–15. Patent Owners do not agree that these constructions are correct, in part because they are inconsistent with the plain meaning of each claim term, but Patent Owners do not object to adoption of Petitioners’ construction of these four terms in the proceeding for the limited purpose of evaluating the unpatentability theories set forth in the Petition. As explained below, Petitioners’ patentability theories fail even if their own proffered construction is adopted. *Accord Promptu Sys. Corp. v. Comcast Corp.*, 92 F.4th 1372, 1380 (Fed. Cir. 2024) (“[O]nly those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy.” (quoting *Vivid Technologies, Inc. v. American Science & Engineering, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999))).

B. Patent Owners' Constructions

Petitioners’ arguments ignore the correct understanding that a POSITA would attribute to the below terms, making claim construction necessary.

1. “a module identifier”

The term “a module identifier” requires an identifier that can identify *each* module.¹ Claims 1 and 13 require that “each module on each board in each chassis” must be addressable through the header, and a POSITA would understand from the surrounding claim language and specification that it is the module identifier that allows this addressability. This is apparent from the structure of the claims. *See Endo Pharms., Inc. v. Actavis LLC*, 922 F.3d 1365, 1370 (Fed. Cir. 2019) (“[C]laim construction must begin with the words of the claims themselves.”). Claims 1 and 13 require a “type identifier,” “instance identifier,” “module identifier,” “board identifier,” and “chassis identifier.” Correspondingly, the claims require that these identifiers are provided “so that each type of module, each instance of a type of module, and each module on each board in each chassis can be addressed through the header.” Thus, it is clear from the claim language itself that the “module identifier” must enable “each module” to “be address[able]” by identifying each module—just as the “instance identifier” must enable “each instance” to “be address[able]” by identifying each instance.

¹ The Board appears to have the same understanding as Patent Owners, finding in the ID that claim 1[e] requires a “corresponding address for *each* connected device” (*i.e.*, module). Paper 11 at 17 (emphasis added).

Despite this unambiguity, Petitioners' arguments appear to rest on an implicit assumption that a module identifier need only identify some (but not all) modules. However, Petitioners' own expert, Dr. Alpert, agreed with Patent Owners' construction during his deposition, testifying that the module identifier must identify each hardware or software module on the pod or card. EX2024, 66:4-20 ("[E]ach module on the board has a number, and that corresponds to the module ID in the claims."); 60:6-17 ("Oh, it identifies the specific one . . . I mean that the module identifier, *a different value for each of the hardware or software modules.*" (emphasis added)); 60:20–22 ("My understanding is that the module identifier would -- would identify a specific one of the hardware or software modules."); 64:13-16 ("Q What would the module identifier identify? A *The module identifier would identify the module for all of the hardware and software modules* regardless of the type." (emphasis added)); 70:8–10 (Q. "A module identifier has to identify a particular module, correct?" A. "Yes, it should. Yeah, it does"). Dr. Alpert explained that it is the module identifier that identifies each module and thus allows each module to be addressable. *Id.*, 61:22–62:3 ("And the module ID in the header would identify a module and in particular so that the -- the module can be addressed through the header").

Finally, that "a module identifier" must identify each module is further supported by the specification. The '753 patent explains that "[a] header on Module

Message Interface packets specifies chassis, board, module, instance and type identifiers, so that packets in the MMI network are routed to a hardware or software module in a pod or card in the chassis or one of multiple chassis.” EX1001, 3:4–8. The inventors further described that the module identifier in each packet “could be static (typically, for hardware), or dynamic (typically, for software),” indicating that the module identifier is specific to each module. *Id.*, 9:19–20. If the module identifier did not identify each module, then packets could not be addressable to each module in the system. *See also* EX2021, ¶¶39-40 (explaining that this understanding of “module identifier” is consistent with the purpose of the invention to allow users maximum flexibility in configuring its components).

2. “sandbox”

Petitioners do not explicitly propose a construction for the term “sandbox;” nevertheless the Petition and Petitioners’ expert *sub silentio* adopt one: “portion designated for programmable electronic circuits.” Pet., 60 n.7; EX1003, ¶161 n.7. Moreover, Petitioners interpret their construction to mean that a sandbox can be **anything** so long as it includes “programmable electronic circuits.” As explained below in Section VII.A.3.b, Petitioners’ overbroad understanding is unsupported and should be rejected.

3. “selecting a memory superposition” and “selectable arrangement”

The terms “selecting a memory superposition” and “selectable arrangement” should be interpreted to require that selection occur during the “method for data flow processing in the platform.” That understanding is consistent with the plain meaning of the terms in the context of the claims and specification. Construction is necessary because Petitioners’ positions and the testimony of its expert suggest that it is applying an overbroad construction that would encompass selection by a system designer before the claimed method even begins.

The claim language at issue appears in limitation [7d]. Claim 7 is, by its own terms, a “method for data flow processing in a platform” and includes, for example, the step of “communicating a data flow” and “selecting a memory superposition.” EX1001, cl. 7. The plain language of the claim thus shows it is directed to the *use*, not original design, of a data processing “platform.” EX2021, ¶¶80-81. Further, with respect to limitation [7d] specifically, the specification makes clear that memory superposition selection is a dynamic process that occurs during system configuration or operation. *See, e.g.*, EX1001, 8:22-46 (explaining that memory superposition permits memory structures to be “reconfigured for subsequent tasks”), 7:21-56 (explaining that in memory superposition “different types of memory are being overlapped, each with benefits and drawbacks, to form a final memory system which performs optimally in a given video stream or other data flow/data processing task”).

In short, a POSITA would understand the claims to require selection of a memory superposition to occur not during the design of the platform, but “during runtime” (configuration and operation). EX2021, ¶81.

Petitioners, however, appear to rely on a broader and unreasonable interpretation of the claims under which the “selecting” step is satisfied merely by a system designer’s choice of a particular memory arrangement during the original design of the platform. For example, Petitioners’ Ground 2 hinges on the contention that Cragon teaches the claimed “selecting,” even though Cragon merely discloses a variety of disk attributes and arrangements that designers can consider during system design. *See, e.g.*, Pet., 73-74; EX1003, ¶¶217-218. Further, during his deposition Dr. Alpert was unwilling to exclude selection during the design phase from the scope of the claims. EX2024, 88:19-89:22. And during redirect he expressly agreed that “Claim 7 allow[s] the claim selection to be performed when the user designs software and CLD functionality.” *Id.*, 149:13–21.

This broad interpretation of the claims is inconsistent with the claim language and specification, as discussed above, and also defies common sense. If “selecting” can occur during the design phase, and if a “selectable arrangement” is merely an arrangement that one *could* select during system design, then the terms effectively have no meaning—all existing memory systems would be “selectable” since they are technologically achievable and thus are, in a trite sense, available to “select”

during system design. EX2021, ¶82. And any memory arrangement present in the system would have been “selected” since at some point it was selected by a designer to be included in the platform. This unreasonably broad interpretation effectively renders the terms superfluous. *See Power Mosfet Techs., L.L.C. v. Siemens AG*, 378 F.3d 1396, 1410 (Fed. Cir. 2004) (“interpretations that render some portion of the claim language superfluous are disfavored”).

VII. PETITIONERS HAVE NOT SHOWN BY A PREPONDERANCE OF THE EVIDENCE THAT THE CHALLENGED CLAIMS ARE UNPATENTABLE AS OBVIOUS

A. Petitioners Fail to Prove their Combination of Stroud and DaSilva (Ground 1) Discloses or Renders Obvious Any Challenged Claim.

1. Stroud Does Not Disclose or Render Obvious a “Module Identifier” or That “Each Module . . . Can Be Addressed Through The Header” As Required By Claims 1 and 13.

Petitioners have not shown that Stroud discloses a “module identifier” or that “each module . . . can be addressed through the header” as required by claim elements 1[e] and 13[e].² Petitioners allege that Stroud’s “6th Byte” of the MAC address—the Management Switch 110 “interface identifier”—is a “module identifier.” Pet., 42–43, 46 (citing EX1004, ¶372). Petitioners acknowledge that

² Petitioners only allege that Stroud discloses this element, not that DaSilva discloses it or that Stroud and/or DaSilva render it obvious. *See* Pet., 42–43, 46 (“Stroud teaches . . .”).

Stroud’s Management Switch 110 interface identifier does not directly identify modules but argues that it “identifies modules by their interface to the [M]anagement [S]witch [110].” Pet., 42; EX1004, ¶372. That is, Petitioners’ argument is that each module has a unique interface to the Management Switch 110 and therefore the interface identifier effectively identifies each module as well. But Petitioners’ fundamental premise—that each module has a unique interface to the Management Switch 110—is not only unsupported by Stroud; it is explicitly contradicted. Stroud discloses that each module **does not** have an interface with the Management Switch 110. EX2021, ¶99. Some modules do not interface directly with the Management Switch 110 but instead connect through other modules. *Id.*, ¶97. Thus, Stroud’s Management Switch 110 “interface identifier” is not a module identifier, because it cannot identify **each** module, and does not make **each** module addressable, as required by the claims.³ *Id.*

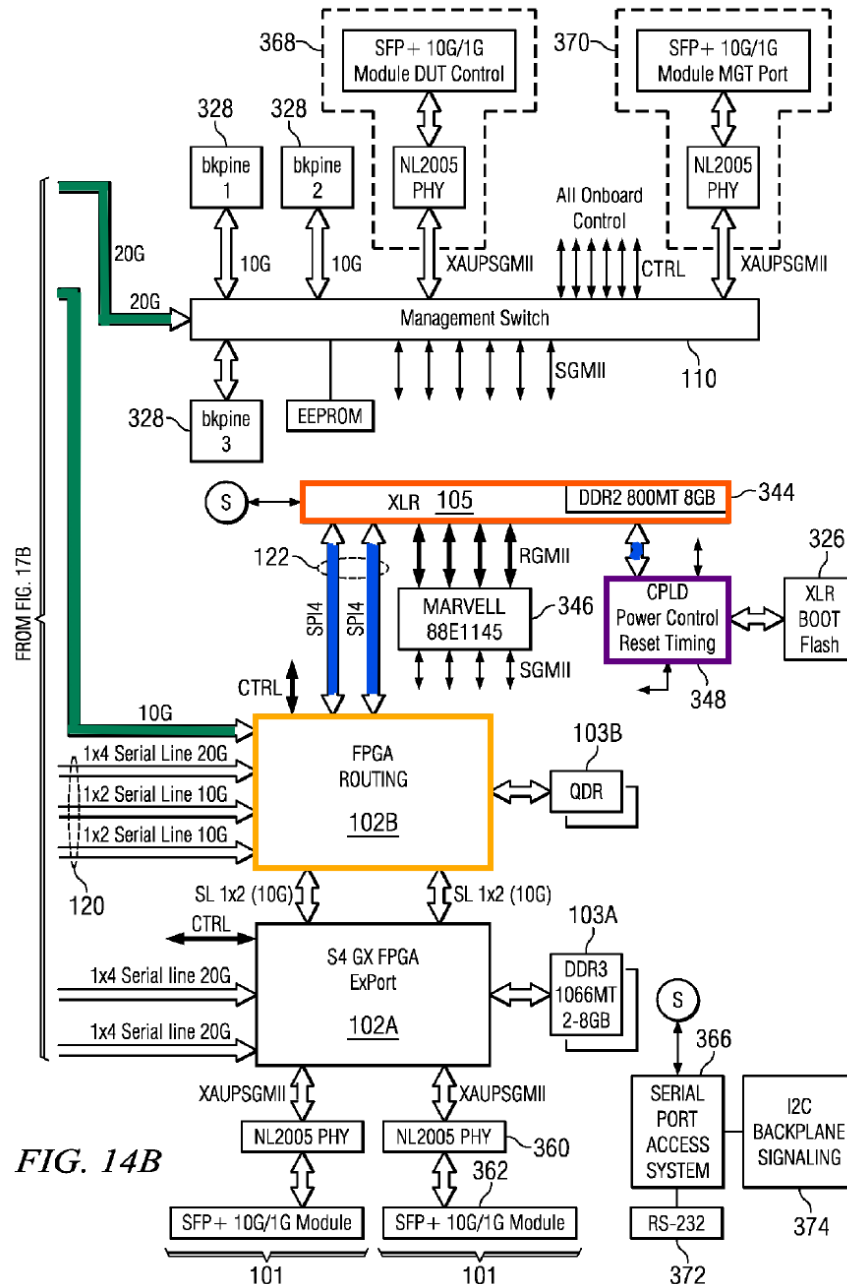
³ As explained above, the Board did not have occasion to consider this argument at institution. Section II.D. At institution, Patent Owners argued that, even if Petitioners were correct that the interface identifier uniquely identified each particular Controller, CLD, and NP module, that was insufficient because Stroud disclosed that the Controller, CLD, and NP were sub-modules that could be combined into larger modules, and the larger modules were not themselves identified by the interface

Stroud includes only a single sentence regarding “interface identifier”: “6th Byte: indicates processor interface (each interface to the management switch has its own MAC address).” EX1004, ¶372. Of course, this sentence does not itself disclose that the interface identifier identifies each module, or even that each module has a unique interface to the Management Switch 110. EX2021, ¶101. EX1001, 8:65–67. Similarly, looking at Stroud as a whole, nothing discloses or suggests that each module has its own interface to Management Switch 110 such that the interface identifier could function as a module identifier. EX2021, ¶101. That fact by itself ends the inquiry. Petitioners’ theory for “module identifier” relies on their assertion that each module has a unique interface to Management Switch 110, yet they have pointed to nothing in Stroud to support that assertion. *Harmonic Inc. v. Avid Tech.*,

identifier. *See* POPR (Paper 7), 20–22; ID (Paper 11), 17. The Board rejected that argument based on Petitioners’ unchallenged assertion that “the identified CLD and processor (e.g., NP) modules [are] distinct devices *with a distinct connection to Ethernet switch 110 interface with a corresponding address for each connected device.*” Paper 11, 17 (emphasis added). As shown herein, Petitioners’ underlying assertion is factually wrong—each CLD and NP does *not* have “a distinct connection to Ethernet switch 110 interface with a corresponding address for each connected device.”

Inc., 815 F.3d 1356, 1363 (Fed. Cir. 2016) (burden on Petitioner to show “with particularity” that claims are unpatentable).

But Stroud is not silent on this issue; it discloses that some modules *do not* interface directly with Management Switch 110 and therefore each module in Stroud *does not* have a unique Management Switch 110 interface. In particular, Stroud discloses several unique modules that sit behind other modules, meaning that those modules do not interface with Management Switch 110 but instead connect through other modules. For example, XLR 105 (shown in red below), CPLD Power Control Reset Timing 348 (shown in purple below), and FPGA Routing 102B (shown in orange below) are all modules. EX2021, ¶¶106-11; EX1004, FIG. 14B; EX2024, 83:17–20; 75:7–11. However, while FPGA Routing 102B is directly connected with the Management Switch 110 (route is shown in green below), XLR 105 and CPLD Power Control Reset Timing 348 *are not*. XLR 105 and CPLD Power Control Reset Timing 348 have no interface to the Management Switch 110; XLR 105 is only connected via FPGA Routing 102B and CPLD Power Control Reset Timing 348 is connected through both XLR 105 and FPGA Routing 102B (route shown in blue below).



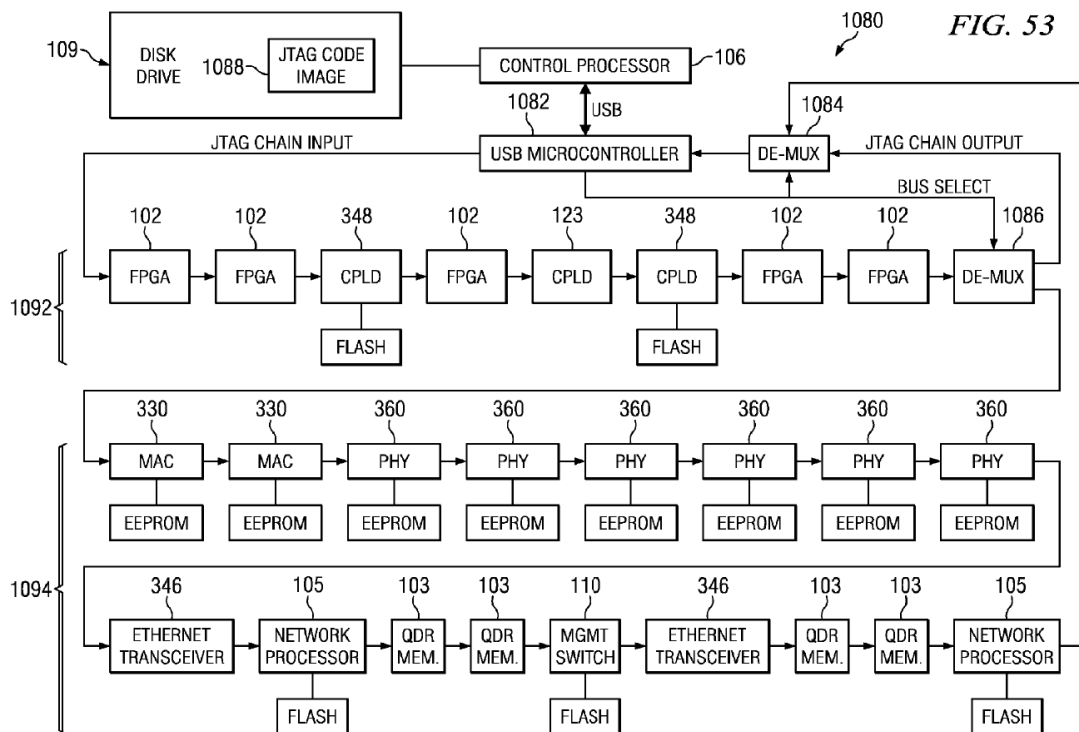
EX1004, FIG. 14B. In practical terms, while there might be an “interface identifier” that corresponds to FPGA Routing 102B (as it appears to have an interface to Management Switch 110), there cannot be an “interface identifier” for either CPLD Power Control Reset Timing 348 and XLR 105 (as they do not have such an interface). EX2021, ¶106. Expanding beyond Figure 14B, this exposes the

fundamental flaw with Petitioners' theory: the interface identifier does not identify *each* module or make *each* module addressable. To the extent it appears to identify *any* module, it is coincidence (*i.e.*, that module happens to have an interface with Management Switch 110). Stroud does not disclose that the interface identifier is a module identifier.

Petitioners and their expert both ignore Stroud's disclosure that some modules do not have an interface with Management Switch 110 and thus are not addressable or identifiable by the interface identifier. Worse yet, Petitioners' expert admitted at his deposition that he did not even perform the necessary analysis to support Petitioners' theory. When asked at his deposition whether the CPLD Power Control Reset Timing (in purple above) is a module, he said "it's not something that I've considered . . . I'm not prepared to offer an opinion about 348." EX2024, 76:4-21. Yet Dr. Alpert acknowledge that the claims require that *each* module must be addressable. EX2024, 78:9-12. Dr. Alpert cannot persuasively opine that the interface identifier allows *each* module to be addressable and identified without identifying what the modules are. It appears that Dr. Alpert only reached his opinion by analyzing the few modules that happen to correspond to the interface identifier and ignoring the rest. Without expert opinion that the interface identifier identifies and can address *each* module (not just Petitioners' hand-selected modules),

Petitioners cannot carry their clear and convincing burden as a matter of law. *See TQ Delta, LLC v. Cisco Sys.*, 942 F.3d 1352, 1358 (Fed. Cir. 2019).

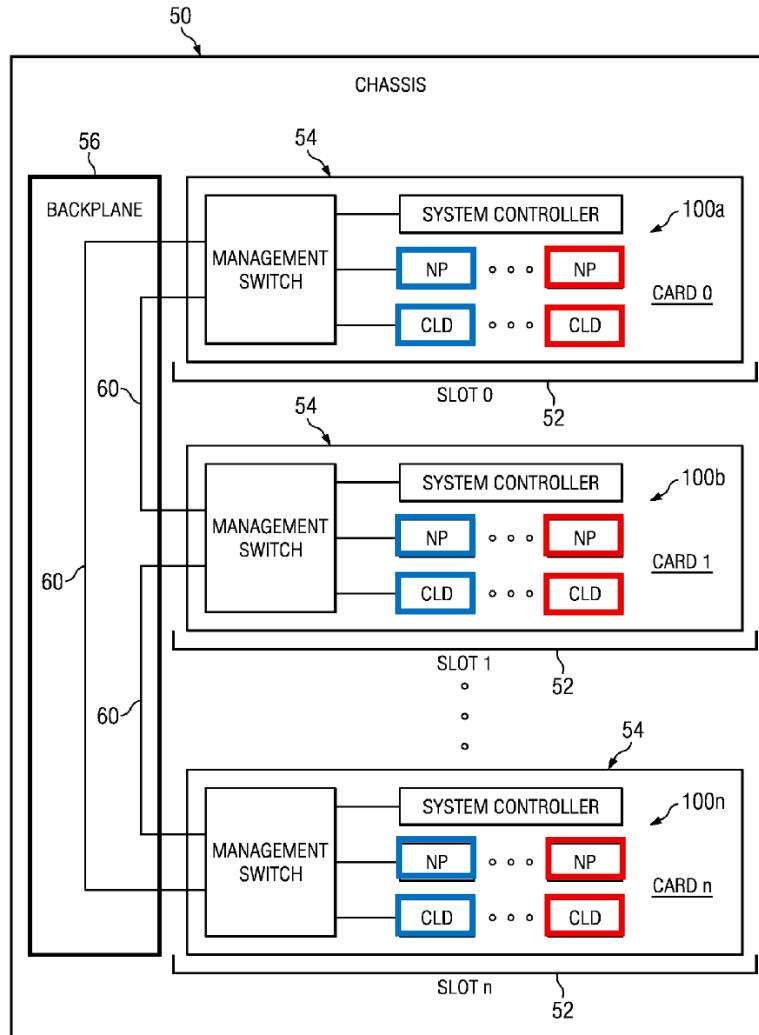
Nor is Figure 14B the only example in Stroud that show modules that do not have interfaces with the Management Switch 110, and thus are not addressable or identifiable by the interface identifier. In Figure 53, there are numerous modules chained together sitting between each other without a direct connection to Management Switch 110:



EX1004, FIG. 53, ¶309 (“CPLD 348 may provide glue logic between network processor 205 and boot flash 326 (e.g., because the network processor may be capable of interfacing flash memory directly); EX2021, ¶108. CPLD 348 may also provide reset and power sequencing for network processor 105.”); *see also id.*, Fig.

23 (showing the Capture/Offload CLD 102A, Routing CLD 102B, Network Processor 106 and Control Processor 105—all of which Petitioners identify as modules—between the Test Ports 101 and the disk drive 109, without interfaces to the Management Switch 110).

Stroud is replete with such evidence. Figure 3 shows that while one layer of NPs and CLDs is directly connected to the Management Switch 110 (shown in blue below), there more than NPs and CLDs chained behind the top-level NP and CLD (shown in red below):



EX1004, FIG. 3; EX2021, ¶110. Similarly, in Figure 15 the Nehalem Control is a module that does not have an interface with the Management Switch 110. Pet., 26 (identifying “controller 106”⁴ as a software module); EX2021, ¶110.

⁴ The Nehalem Control refers to the Controller 106. See EX1004, Fig. 4; EX2021, ¶113.

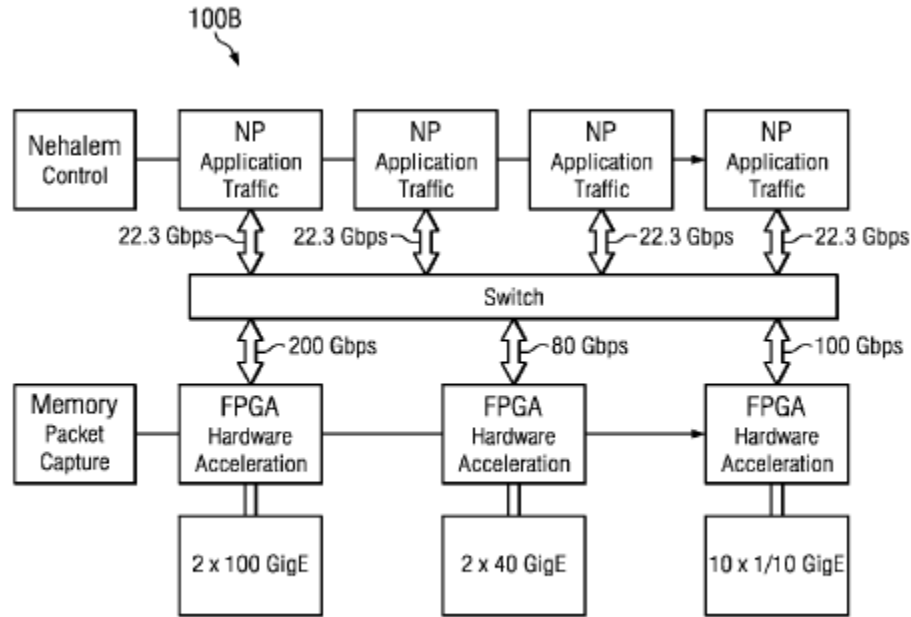


FIG. 15

Id., FIG. 15; *see also id.*, Fig. 20 (showing that the “network processor 105”—which Petitioners identify as a module—does not interface with the Management Switch 110 but goes through desk drive 109 and in particular, DHCP server 300); EX2021, ¶113.

Petitioners provide a litany of citations to Stroud, but none support their fundamental assertion that each module “is connected to the management switch” such that “each NP/CLD ha[s] a dedicated connection to the management switch,” and thus the “interface identifier” “is unique to the module within each card.” Pet., 42-44 (citing EX1004, ¶100, ¶101, ¶103, ¶113, ¶¶117-118, ¶134, ¶¶146-147, ¶¶156-157, ¶167, ¶¶174-176, ¶189, ¶310, ¶¶314-319, ¶354, ¶366, Figs. 3-5, 6A-9A); EX2021, ¶112. For example, ¶¶100, 101, 103, and 113 merely list different components that may be included in the system without regard to whether they have

their own interfaces and can be addressed by the interface identifier. Paragraphs 117–118 do the same; they merely discuss that there are multiple CLDs within the system which have relationships to other components, but do not discuss anything about interface connections. EX1004, ¶¶117–118. In fact, this is true with every citation listed above: Stroud discloses that software and hardware elements may connect to allow for packet processing and configuration, but not a single citation demonstrates how the “interface identifier” is unique to each module in the system. EX2021, ¶112. The interface identifier does not identify *each* module (*i.e.*, it does not identify modules with shared and non-unique connections to Management Switch 110). *Id.*

Moreover, as Dr. Min explains, Stroud’s interface identifier fails to achieve the purpose of the claimed module identifier, because it, at best, can only identify modules that are directly connected to the Management Switch 110. EX2021, ¶113, ¶104 (“An interface identifier, by contrast, only identifies a point of attachment to the switch, which requires additional logic to determine which downstream module should receive the message and limits flexibility as modules are added, removed, or reconfigured.”). Stroud’s interface identifier cannot identify any modules that do not have a direct interface to the management switch.

Fundamentally, Petitioners attempt to fit a square peg in a round hole. In order to bridge the gap between what the claims require (a module identifier) and what

Stroud actually discloses (an interface identifier), Petitioners equate modules and Management Switch 110 interfaces. But modules do not correspond one-to-one to interfaces. Petitioners’ assertion that the interface identifier is “unique to the module within each card,” Pet., 42–43 (citing EX1004, ¶372), is unsupported and untrue. While the interface identifier might inadvertently correspond to some modules (those that interface directly with Management Switch 110), it definitionally does not identify any modules that do not have a direct interface to Management Switch 110—and thus does not identify *each* module or make *each* module addressable. Petitioners have failed to show that Stroud discloses the claimed “module identifier.”

2. Claims 1–3, 13, 14 are Patentable over Stroud and DaSilva.

Because independent claims 1 and 13 are patentable over Stroud and DaSilva, the more specific dependent claims are also patentable for the same reasons. EX2021, ¶115.

3. Petitioners Fail to Prove that Stroud Discloses Each Limitation of Claim 3.

Petitioners also fail to show that Stroud discloses each limitation in claim 3. Claim 3 requires that one of the swappable pods or cards includes a single processor that must be allocatable for four distinct functions: (1) “offloading hardware tasks to software,” (2) “orchestrating data flows,” (3) “orchestrating configurations of hardware modules,” and (4) “executing in cooperation with a sandbox of one of the plurality of swappable pods or cards.” EX1001, cl. 3. The Petition points to the

“controller 106 which comprises a ‘management processor’” as the claimed “processor,” but fails to show that it meets each claim 3 requirement. *First*, Stroud does not disclose a “processor” allocatable for “offloading hardware tasks to software.” *Second*, Stroud does not disclose a processor “executing in cooperation with a sandbox of one of the plurality of swappable pods or cards.”

a. Stroud Does Not Disclose “At Least One Processor Allocatable for Offloading Hardware Tasks to Software.”

Claim 3 of the ‘753 patent requires that the claimed “processor” is “allocatable for offloading hardware tasks to software.” EX1001, cl. 3. This claim element allows the claimed “processor” to optimize packet processing by offloading packets that are originally assigned to hardware units to be processed by software instead. *Id.*, 2:67–3:3. As the specification explains, “the workload can be offloaded from a programmable logic device [which is hardware] to a processor [running software] for tasks that can be done serially and/or slower.” EX1001, 2:67–3:3; 10:1–3. Petitioners fail to demonstrate that Stroud renders obvious that the claimed “processor” is allocatable for the claimed “offloading.”⁵

⁵ Petitioners only allege that Stroud discloses or renders obvious this element by itself, not that DaSilva discloses or renders it obvious. *See* Pet., 57.

i. The Petition Relies On A Different Alleged “Processor” For “Offloading” Than It Does For The Other Three Claimed Functions.

The Petition’s arguments regarding the “offloading” element are internally—and fatally—inconsistent. The Petition alleges that the claimed “processor” in Stroud is the “controller 106 which comprises a ‘management processor.’” Pet., 57-59. But when alleging that Stroud discloses the claimed “offloading,” the Petition points to actions performed by a *different* device altogether: CLD 102B, also referred to as the “routing CLD.” Petitioners’ claim 3 allegations do not present a consistent invalidity theory, and thus fail as a matter of law. *See Mianyang BOE Optoelectronics Technology Co., Ltd. et al v. Samsung Display Co., Ltd.*, IPR2024-00620, Paper 38 at 30 (PTAB Sep. 17, 2025 (“We agree with Patent Owner that Petitioner’s contentions for this ground are fatally flawed because they are internally inconsistent.”)).

While Petitioners attempt to obfuscate this inconsistency, it is apparent from a comparison of the Petition and the text of Stroud. The Petition alleges that Stroud’s disclosure of “[i]ncoming packet routing and load balancing” meets the claimed “offloading.” Pet., 57 (citing EX1004, ¶¶245-248, ¶321, Fig. 15). But Stroud clearly discloses that the identified “routing and load balancing” is performed by CLD 102B—not the “controller 106 which comprises a ‘management processor’” that

Petitioners map to the claimed “processor.”⁶ *See, e.g.*, EX1004, ¶241, ¶¶245–248, ¶¶410–11, ¶459; *compare* ¶157 (discussing the “controller 106” *without* stating that it performs the routing and load balancing) *with* ¶¶128–143 (explaining that is it the “Router CLDs 102B” that exclusively perform the routing and load balancing); EX2024, 110:12-111:6; EX2021, ¶111. CLD 102B is the “*router* CLD”—it is this “router CLD,” not “controller 106” or any subparts, that perform the routing and load balancing. EX1004, ¶241, ¶¶245–248, ¶¶410–11, ¶459.

Petitioners’ own expert admitted that “routing and load balancing is done *by that routing CLD 102B*.” EX2024, 115:15-116:13. When asked at his deposition, Dr. Alpert testified that CLD 102B is solely responsible for the identified routing and load balancing—both in the specific paragraphs cited by Petitioners and in Stroud more generally. *Id.* (Q. “Stroud generally discloses that its routing and load balancing is done by that routing CLD 102B, right?” A. “Yeah, these paragraphs of Stroud do say that the CLD 102B is doing the load -- the routing and load balancing.” Q. “And that’s true for Stroud generally, right?” A. “Yes. That’s true for Stroud, and

⁶ Confusingly, Petitioners argue “it was obvious to offload hardware tasks to . . . the controller.” Pet., 57. Claim 3 requires that the “processor” offloads hardware tasks to software, not that another device can offload hardware tasks *to* the claimed processor.

there’s just a number of different places that 102B is identified in the figures.”), 116:14–117:2. Petitioners’ alleged “offloading”—routing and load balancing—is not performed by the alleged “processor,” but by CLD 102, a different device entirely.

Petitioners’ argument for claim 3 falls apart in the face of this inconsistency. Claim 3 requires a *single* processor that is allocatable for all four claimed functions; alleging that one device performs three functions and another device performs the fourth function is insufficient to show the claim is obvious, even if those allegations were correct (they are not, *see infra*).

**ii. Stroud Does Not Disclose The Claimed
“Offloading Hardware Tasks To Software.”**

In addition to Petitioners’ theory being internally inconsistent, it also fails because Stroud does not disclose the claimed “offloading.” Pet., 57. Stroud only discloses routing and load balancing *software* tasks, not *offloading hardware* tasks *to* software.

Petitioners’ argument begins with—and fundamentally relies on—a mischaracterization of Stroud. Petitioners allege that “the platform practices ‘[i]ncoming packet routing and load balancing’ to distribute packets among NPs and CLDs (FPGAs).” Pet., 57. Petitioners careful quoting is illuminating. The quote in full is: “[i]ncoming packet routing and load balancing enable multiple *network processors 105* to be used efficiently in a single system.” EX1004, ¶245 (emphasis added). Stroud’s “network processors 105” are software. Pet., 29. Petitioners’

citation thus only shows routing tasks *between* software elements—Stroud’s “[i]ncoming packet routing and load balancing” has nothing to do with offloading from hardware to software devices. EX1004, ¶245.

It is undisputed that routing between software elements does not disclose offloading hardware tasks from hardware elements to software. Both experts agree that “offloading hardware tasks to software” “means transferring a task that’s been previously assigned to hardware to be performed by software instead.” EX2024, 108:2–12 (hardware tasks are tasks that have been assigned to a hardware element for execution); EX2021, ¶133. The claims are not met by routing and load balancing (1) tasks that have never been assigned to hardware, or (2) tasks between different software elements.

Implicitly recognizing this flaw, Petitioners next claim the limitation would be obvious, arguing “when were [sic] CLDs near capacity, it was obvious to offload hardware tasks to a NP, including the controller which acts ‘as an auxiliary network processor.’ Pet., 57 (citing EX1004, ¶154). But there is no discussion in Petitioners’ cited paragraph about offloading from hardware to software, instead it only explains that controller 106 can act as an “auxiliary network processor.” In other words, controller 106 can act as backup for other “network processors”—which are software—to perform overflow software tasks. EX2021, ¶¶124-25, 134. There is no indication or suggestion that the controller 106 can perform any hardware tasks. *Id.*

Further, the surrounding portions of Stroud list the tasks the control processor 106 can perform, none of which include offloading hardware tasks to itself or to other network processors. EX1004, ¶¶154–163.

Petitioners next allege that “[i]t was common to offload complicated tasks from hardware to avoid clogging the pipeline,” Pet., 57, but they do little to support their bald assertion. Petitioners’ only citation is to Dr. Alpert’s declaration, but that itself lacks any support or reasoning. Dr. Alpert’s opinion largely parrots the petition, adding nothing in support of Petitioners’ crucial obviousness argument. *See, e.g., Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016) (“[Petitioner’s] expert testimony in support of this theory is equally unavailing, adding nothing beyond the conclusory statements in [the] petition.”).

Dr. Alpert’s only arguably non-parroted opinion is wholly unsupported. Dr. opines that it “was a common practice in network processing, where time-consuming or complicated tasks were offloaded from hardware to CPUs to avoid clogging up the packet pipeline.” EX1003, ¶151. However, his citations to Stroud do not disclose anything of substance. Paragraphs 245–48 only show how control processor 106 configures CLDs and NPs during system design time and serves as the user interface for managing the system; nothing related to offloading from hardware to software. EX1004, ¶¶245–248. The same is true for Figure 15 and ¶321, which merely show that the system contains a multitude of different CLDs and NPs. EX1004, FIG. 15,

¶321. Given this lack of evidence, Dr. Alpert’s opinions are conclusory and unsupported and should be given little weight. *Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9, 15 (“Thus, the cited declaration testimony is conclusory and unsupported, adds little to the conclusory assertion for which it is offered to support, and is entitled to little weight.”). Petitioner bears the burden of demonstrating unpatentability and that a POSITA would have had a reasonable expectation of success in arriving at the invention—they have not met that burden. *See Eli Lilly & Co. v. Teva Pharms. Int’l GmbH*, 8 F.4th 1331, 1348 (Fed. Cir. 2021).

Finally, Petitioners argue that Stroud “suggests workloads are portable between NPs/CLDs,” but neither Petitioners nor their cited paragraphs tie this portability back to the claimed offloading or explain how it relates at all. Pet., 57. For example, Stroud teaches that “the capture CLD and L2/L3 CLD may be configured to offload a portion of traffic, e.g., 25%, from the network processors,” but this does not disclose or suggest offloading hardware tasks to software. EX1004, ¶296. If anything, it shows the complete opposite: a portion of traffic is offloaded **to hardware**. Nor do the litany of paragraphs describing Figure 12 suggest otherwise as they all identify that “[i]ncoming packets, also called ingress packets, arriving on external interfaces may be processed by one or more of several core functional modules in high-speed configurable logic devices,” and their output “may be fed into one or more network processors along with the ingress packet.” *Id.*, ¶¶282–94.

Again, no suggestion that hardware tasks can be offloaded to software. Claim 3 requires that a processor offloads hardware tasks to software to remove the burden from hardware, not simply that tasks can move between modules on a board. EX1001, cl. 1.

As Dr. Min explains, it would not be obvious to a POSITA to offload from hardware to software. As Dr. Min explains, systems typically offload packets in the opposite direction—from software to hardware—to conserve software resources for more complex computational tasks. EX2021, ¶142. The inventors of the '753 patent, however, built a modular, swappable, flexible dataflow system that depended on each processor operating at optimal efficiency. *Id.* While typical systems seek to offload software tasks to hardware to free up software resources, the '753 patent understood that its system would benefit from tasks flowing in each direction. *Id.* This is why the '753 patent discloses a system of “deeply embedding processing elements in hardware.” EX1001, 6:28–30. Offloading hardware tasks to software was not routine or conventional and a POSITA would not find it obvious to modify Stroud to add it in wholesale.

Petitioners' argument boils down to an attempt to insert the claimed “offloading” into Stroud without reasoned analysis and evidentiary support.

Petitioners’ attempt should be rejected.⁷ *See Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1362 (Fed. Cir. 2016) (“[O]ur cases repeatedly warn that references to ‘common sense’—whether to supply a motivation to combine or a missing limitation—cannot be used as a wholesale substitute for reasoned analysis and evidentiary support, especially when dealing with a limitation missing from the prior art references specified.”).

b. Stroud Does Not Disclose a Processor “Executing in Cooperation with a Sandbox of One of the Plurality of Swappable Pods or Cards.”

Claim 3 requires that the claimed “processor” is also allocatable for “executing in cooperation with a sandbox of one of the plurality of swappable pods or cards.” EX1001, cl. 3. Stroud does not disclose that any of the “swappable pods or cards” have “a sandbox,” and thus does not disclose this limitation.⁸

⁷ Petitioners’ argument should also be rejected because its wholesale creation of a limitation—offloading hardware tasks to software—that is neither taught nor suggested by either of the references plainly violates 37 C.F.R. § 42.104(b)(4) and Federal Circuit precedent and thus fails as a matter of law. *See* Section VII.B.1.

⁸ Petitioners only allege that Stroud discloses this element, not that DaSilva discloses it or that Stroud and/or DaSilva render it obvious. *See* Pet., 60.

Petitioners argue that “Stroud’s CLDs *include* sandboxes” because “they comprise PLDs/FPGAs with portions for user-definable hardware” Pet., 60 (emphasis added, internal citations omitted).⁹ But Stroud does not disclose “sandboxes.” Petitioners’ citations do not mention sandboxes—they only disclose that CLDs can comprise FPGAs and PLDs. EX1004, ¶107, ¶¶109–12, ¶117 ¶¶311–13 (each disclosing only that Stroud’s CLDs can be PLDs/FPGAs, but not mentioning sandboxes), ¶462; EX1003, ¶161 (discussing the operation of CLDs without mentioning sandboxes). While Stroud discusses that the system may contain a CPU, GPU, NP, or ASIC, it is silent on whether there are sandboxes within these devices. Pet., 60 (citing EX1004, ¶¶110–112). And while Stroud defines CLDs as a “device that can be modified to simulate many types of hardware,” it only states that CLDs are reprogrammable to support testing and simulation architecture—not that they include sandboxes. *Id.* (citing EX1004, ¶117). In sum, Stroud does not mention sandboxes at all—much less specify that Stroud’s CLDs “include sandboxes,” as Petitioners claim without support.

What’s more, paragraphs 311–313 demonstrate affirmatively that Stroud’s CLD’s do *not* contain sandboxes. Pet., 60. In those paragraphs, Stroud explicitly

⁹ Petitioners do not otherwise argue that it would obvious—they only rely on Stroud to demonstrate the claimed sandbox.

discloses exemplary specific FPGAs and PLDs, and none of them include a sandbox.

Id. For example, Stroud references the ALTERA STRATIX 4. *Id.* But while the ALTERA STRATIX 4 was a widely available FPGA, it ***did not*** come programmed with a sandbox region accessible to the control processor. EX2021, ¶150. Dr. Min discusses that if a user sought to use ALTERA STRATIX 4 (or one of the other identified devices) including a sandbox, they would need to reconfigure the system themselves to create a sandbox out of whole cloth. *Id.* This is reflected in an exemplary datasheet that Dr. Min provided with his report. EX2023.

In short, there is no evidence that “Stroud’s CLDs include sandboxes.” In the face of this silence, Petitioners rely on an overly broad—and unsupported—definition of “sandbox” to argue that although Stroud does not mention sandboxes, CLDs *inherently* include them. First, Petitioners assert that a sandbox means a “portion designated for programmable electronic circuits.”¹⁰ Pet., 60. Next, Petitioners’ expert opines that the wording of this construction means that a sandbox can be ***anything*** that includes “programmable electronic circuits.” EX1003, ¶161. Finally, Petitioners argue that since CLDs (*e.g.*, PLDs/FPGAs) include at least some “programmable electronic circuits,” CLDs must inherently include sandboxes. *Id.*;

¹⁰ As explained above, Petitioners do not explicitly propose a construction for the term “sandbox” but instead adopt one implicitly. Pet., 60 n.7; EX1003, ¶161 n.7.

EX2024, 132:1-12. Petitioners' argument is contrary to the plain and ordinary meaning of "sandbox" and effectively eliminates the limitation altogether.

Shockingly, Petitioners and Dr. Alpert offer no support for their "construction" or its overly broad interpretation. Pet, 60; EX1003, ¶161. At his deposition, Dr. Alpert testified that he only applied Petitioners' construction because he understood (wrongly) that it was agreed to by the parties.¹¹ EX2024, 124:13–17;

¹¹ In the parallel district court litigation, Petitioners proposed that sandbox be construed. EX2026. In response, Patent Owners proposed "sandbox" should be given its plain and ordinary meaning, explaining that they understood that meaning to be "portion designated for programmable electronic circuits." EX1036. Petitioners dropped their proposal to construe sandbox, and it was thus never construed. As is clear from this history, Patent Owners never adopted this construction or, most importantly, Petitioners' overly broad interpretation of it. The fact that Patent Owners once suggested a possible understanding of sandbox's plain and ordinary meaning provides no support or basis for adopting that suggestion as a construction. *In re Viadyanathan*, No. 2009-1404, 2010 WL 2000682, at *989 (Fed. Cir. May 19, 2010); *Microsoft Corp. v. Proxyconn, Inc.*, IPR 2012-00026, Decision re Motion for Rehearing (Feb. 25, 2013) at p. 5 (broad claim constructions must be supported by evidence in the record, namely the specification).

130:3–6 (Dr. Alpert explaining that he did not provide any analysis of the meaning of this term in his declaration). And Dr. Alpert explained that his opinion that a sandbox can be *anything* with “programmable electronic circuits” is based only on Petitioners’ construction—which again, he adopted without analysis. EX2024, 129:5-8. In short, Petitioners’ broad understanding of “sandbox” is based only on the wording of their own construction, which they adopted without evidence, argument, or analysis.¹² Petitioners’ sandbox arguments should be rejected for that reason alone. *See TQ Delta*, 942 F.3d at 1358 (“Conclusory expert testimony does not qualify as substantial evidence.”).

Petitioners’ broad understanding of “sandbox” is wrong. As Dr. Min explains, a POSITA would understand that a sandbox is a *partitioned* portion designated for programmable electronic circuits. EX2021, ¶146, ¶72-73 (“As a POSITA, I believe

¹² Further, Petitioners’ interpretation of their own construction itself does not make sense. Petitioners’ assert that anything that contains programmable electronic circuits is, or at least includes, a sandbox. But that interpretation ignores that Petitioners’ own construction is limited to a “*portion* designated for programmable electronic circuits.” That is, even under Petitioners’ own construction, a sandbox requires that the “programmable electronic circuits” be within a particular “portion”—not just the existence of “programmable electronic circuits.”

that “sandbox” as a term of art with a settled meaning, particularly when viewed in light of the intrinsic record. The plain language of the claims makes clear that it denotes a customizable region inside of a configurable circuit that is different than a general FPGA.”). Although a PLD/FPGA is a configurable device, a POSITA would understand that it does not contain a sandbox unless it contains a partitioned region within it for configurable circuitry. *Id.*, ¶¶78-79, ¶146.

This understanding is also consistent with the ’753 patent specification. The ’753 patent discloses a **PLD** for data flow processing in a user-configurable server that contains a second **region** with “one or more sandboxes with user-definable electronic circuits.” EX1001, 1:36–44. Claim 3 makes clear that one or more of the swappable pods within the system contain a sandbox. *Id.*, cl. 3. The ’753 patent explains that, in some embodiments, the programable device is an FPGA, and the MMI network connection links to sandboxes **within** the FPGA. *Id.*, 4:62–65.

In sum, a sandbox must be a **partitioned** portion designated for programmable electronic circuits. Notably, Petitioners have offered no evidence, expert opinion, or analysis in support of their broader definition, under which any programmable device would inherently “include a sandbox.” And without Petitioners’ overly broad understanding of sandbox, Petitioners have presented no evidence that Stroud discloses or renders obvious this element.

B. Ground 2 fails because Petitioners have not shown that any of its multiple proposed combinations disclose or render obvious the memory limitations of [7d].

Limitation [7d] has several requirements. The claim first requires “selecting a memory superposition in at least one of the plurality of swappable pods or cards.” EX1001, 18:64-65. Next, the swappable pod or card at issue must contain a “plurality of types of memory with selectable arrangement of the plurality of types of memory.” *Id.*, 18:66-67. Finally, the single swappable pod or card must have four specific “selectable arrangement[s]”: (1) “multilevel cache;” (2) “interleaved memory;” (3) “swapped memory over a specified address range;” and (4) “memory allocated to specified address ranges according to parameterized task needs” (collectively, the “claimed memory arrangement”). In other words, Limitation [7d] requires a single pod or card that has multiple types of memory that can be arranged in each of the four claimed memory arrangements, such that the claimed memory arrangements can be selected.

The prior art does not render obvious Limitation [7d] for at least two reasons. *First*, the prior art does not disclose or render obvious a card¹³ that has each of the four claimed memory arrangements. No prior art of record teaches or even suggests

¹³ The Petition uses “card” to refer to both “cards” and “pods.” Pet., 23. Patent Owners do the same herein.

combining these four memory arrangements in particular on a single card. Indeed, no prior art of record teaches combining *any* number of memory arrangements on a single card. And Petitioners do not even meaningfully address, much less establish, why a POSITA would be motivated to choose these four specific memory arrangements over the almost four million alternative configurations that were available. *Second*, the prior art does not disclose or render obvious a card having memory *arrangements* that are *selectable*. Instead, the Petition conjures the claimed selectable arrangement from thin air, relying on conclusory expert testimony as to the alleged knowledge and skill of a POSITA to close gaps in its combinations. As explained below, this is contrary to long-standing Federal Circuit precedent and PTAB regulations.

1. Neither Stroud nor Cragon teaches, suggests, or renders obvious a single card with each of the four required memory arrangements.

It is undisputed that Stroud does not disclose the four specific memory arrangements required by limitation [7d]. *See* EX2024, 92:13-16 (“Stroud does not disclose all four of those memory arrangements.”). Indeed, Petitioners do not rely solely on Stroud alone to disclose *any* of the four required memory arrangements. *See* Pet., 70-78. This is because it does not disclose any memory arrangements. EX2021, ¶161. It is likewise undisputed that Cragon does not disclose the four claimed memory arrangements on a single card—Petitioners’ expert admitted to

precisely that. EX2024, 104:10-14 (“Cragon does not disclose all four arrangements being used in one card.”). Thus *neither* of the two references Petitioners rely on for Ground 2 disclose a key feature of limitation [7d]—the requirement of a single card that has each of the four distinct memory arrangements available. EX2021, ¶¶161-64. Thus Petitioners’ proposed obviousness combination assumes a POSITA would be motivated modify Stroud in multiple ways: by adding the four specific memory arrangements to Stroud (which does not disclose any memory arrangements) and choosing to put them together on a single card (which Cragon does not teach). But neither prior art reference Petitioners rely on even disclose the use of more than one memory arrangement on a card, much less four of them, and the four specific claimed ones at that. Petitioner is thus essentially manufacturing this limitation through its expert, in violation of 37 C.F.R. § 42.104(b)(4)’s requirements that all limitations be found in patents or printed publications. This is reason alone to reject Ground 2. *See* 37 C.F.R. § 42.104(b)(4); *Qualcomm Inc. v. Apple Inc.*, 134 F.4th 1355, 1364-65 (Fed. Cir. 2025); Section VII.B.2.

Petitioners never meaningfully address this core defect in Ground 2. Instead, Petitioners focus on cherry-picking assorted disparate disclosures from Cragon’s over 1200 pages of text¹⁴ to supply the four required memory arrangements. *See* Pet.,

¹⁴ All references to pages in Cragon are to Petitioners’ stamped pages.

74-77; EX1003, ¶¶220-233. But even assuming *arguendo* that Cragon discloses the four claimed memory arrangements (which Patent Owners do not concede), Ground 2 still fails as a matter of law because Petitioners’ have failed to provide any credible reason *why* a POSITA would pick those particular memory arrangements and combine them on a single card—a configuration that neither Stroud, Cragon, or any of the other prior art of record discloses. *See Adidas AG v. Nike, Inc.*, 963 F.3d 1355, 1359 (Fed. Cir. 2020) (“The obviousness inquiry does not merely ask whether a skilled artisan could combine the references, but instead asks whether they would have been motivated to do so.”); *Virtek Vision Int’l ULC v. Assembly Guidance Sys., Inc.*, 97 F.4th 882, 887 (Fed. Cir. 2024) (“The mere fact that these possible arrangements existed in the prior art does not provide a reason that a skilled artisan would have substituted the[m].”).

Here, Petitioners’ proposed motivations are vague, conclusory, and deeply infected with hindsight. EX2021, ¶¶164-70. Petitioners contend that “a POSITA would have been motivated to provide flexible memory arrangements to support Stroud’s flexible architecture and the variety of potential functions on Stroud’s cards,” and consequently “would have been motivated to apply Cragon’s memory arrangements for their known benefits.” Pet., 78-79; EX1003, ¶¶234-38. As Dr. Min opines, Cragon’s dated disclosure is among the last places a POSITA in 2018 would look for flexible memory arrangements. EX2021, ¶167. But even if a POSITA were

to look to Cragon, neither Petitioners nor their expert explain why a POSITA would have been motivated to select these particular four memory arrangements and use them together on a single card. EX2021, ¶168.

The testimony of Petitioners' expert underscores why this omission is such a grave defect in Petitioners' theory. In his deposition, Dr. Alpert emphasized that there were "a very large variety of ways that memories have been arranged in order to achieve performance and cost goals for different types of systems" and ultimately agreed that "there were *hundreds of memory arrangements out there as of the priority date.*" EX2024, 103:17-104:6. Even conservatively reducing this to 100 possible memory arrangements, as Dr. Min explains that means that there were *almost four million possible sets of four memory arrangements* available to POSITA. EX2021, ¶165. And there is nothing in the record—other than the challenged claims themselves—teaching or suggesting that sets of memory arrangements (whether consisting of four arrangements or otherwise) were desirable. If larger set sizes are considered the quantity of potential permutations reaches the billions. *Id.* The fact of the matter is that the inventors of the '753 patent discovered that it was particularly advantageous for a single card to have the four particular claimed memory arrangements, and Petitioners have failed to identify anything predating the patent that even suggests that innovation. EX2021, ¶51.

Nothing in the Petition addresses head on why a POSITA would choose the particular set of four claimed arrangements over the almost innumerable alternatives—much less why a POSITA would choose to use them together on a single card. Petitioners’ vague suggestion that Cragon teaches “the details for using different types of memory together” in memory configurations (Pet., 79) does nothing to explain why a POSITA would be motivated to combine different memory configurations on a single card (much less the four claimed arrangements in particular).¹⁵ As Dr. Min explains, the cited portions of Cragon merely disclose advantages and disadvantages of a variety of different memory configurations that disk designers can consider when designing memory systems. EX2021, ¶¶90-94, ¶¶168, ¶¶173; EX1006, 000536-548; *see also* EX2024, 99:19-22 (Dr. Alpert agreeing that Section 5.2.1 of Cragon “does talk about the -- some of the issues that a system designer would face related to the performance and other characteristics of disks”). Nothing in Cragon discloses or even suggests that *these distinct memory arrangements can or should be used together in same card*. EX2021, ¶¶163. To the

¹⁵ “Memory types” refer to different forms of memory, such as RAM and ROM. EX2021, ¶¶50. “Memory arrangements” are different ways of arranging those forms of memory, such as “interleaving” the RAM and ROM together. EX2021, ¶¶51; Section II.A.

contrary, Cragon’s decades-old disclosure, which focuses on dated memory hardware, teaches away from such a configuration. EX2021, ¶164.

Indeed, Petitioners’ expert admitted that he offered no opinion in his declaration on whether the four distinct memory arrangements would even be compatible if used together on a single card. EX2024, 105:2-6 (“Q. Okay. Can you combine all types of memory arrangements together successfully? A. It’s not something I thought about. *I don’t think I have an opinion whether all types can be combined.*”).¹⁶ And Petitioners ignore that Cragon itself states that the core portion of its disclosure upon which Petitioners rely relates to “disks” that “have

¹⁶ Recognizing how problematic this testimony was for Petitioners’ obviousness theories, on redirect Petitioners’ counsel attempted to elicit new opinions from Dr. Alpert on compatibility. See EX2024, 142:4-19. Apart from being conclusory and unsupported, these opinions are plainly improper, and Patent Owners reserve the right to move to exclude them. Regardless, even if Petitioners had shown that the four memory arrangements were compatible, it does not cure Petitioners’ failure to come forward with any reasoned basis for a POSITA to be motivated to create such a configuration. A reasonable expectation of success (“could combine references”) does not establish a motivation to combine (“would have been motivated to do so”). *Adidas*, 963 F.3d at 1359.

been discussed in terms of a peripheral device.” EX1006, 000558. This is worlds away from the implementation on a single card that claim 7 requires. EX2021, ¶51.

Rather than explaining why a POSITA would be motivated to stitch together the disparate disclosures within Cragon upon which Petitioners rely, Petitioners treat the 1200-page textbook as a single monolithic embodiment, all parts of which can be freely combined with Stroud for the same conclusory reasons. This is contrary to law. The obligation to provide a motivation to combine applies not just to combinations of references but to combinations of disclosures *within a single reference*. *In re Stepan Co.*, 868 F.3d 1342, 1346 n.1 (Fed. Cir. 2017) (“Whether a rejection is based on combining disclosures from multiple references, combining multiple embodiments from a single reference, or selecting from large lists of elements in a single reference, there must be a motivation to make the combination and a reasonable expectation that such a combination would be successful . . .”).

Here, Petitioners and their expert do not even attempt to explain why a POSITA would be motivated to combine Cragon’s distinct disclosures of different memory arrangements *with each other*. Nor do they explain why a POSITA would pick these particular memory arrangements from Cragon even though, as Dr. Min explains, Cragon describes many others. EX2021, ¶168-69. Instead, Petitioners effectively argue that a POSITA would be motivated to apply any and all of Cragon’s teachings regarding memory arrangements to Stroud, in any and all possible

combinations. This is legally insufficient. *See TQ Delta, LLC v. CISCO Sys., Inc.*, 942 F.3d 1352, 1360 (Fed. Cir. 2019) (rejecting expert testimony “because it failed to explain why a person of ordinary skill in the art would have combined elements from specific references *in the way the claimed invention does*” (emphasis in original)); *Amerigen Pharms. Ltd. v. UCB Pharma GmbH*, 913 F.3d 1076, 1089 (Fed. Cir. 2019) (rejecting “general motivation” and emphasizing petitioner’s burden “to show that the prior art would have suggested making the specific molecular modifications necessary to achieve the claimed invention” (citation omitted)).

Ultimately, even if a POSITA were motivated to “provide flexible memory arrangements” to improve Stroud as Petitioners contend (Pet., 78), Petitioners have come forward with no credible basis for concluding that a POSITA would be motivated to create such flexible arrangements by choosing the four specific memory types required by limitation [7b] and combining them on a single card. EX2021, ¶¶168-69. The inescapable conclusion is that the only reason to select this particular configuration is impermissible hindsight. *See In re Fritch*, 972 F.2d 1260, 1266 (Fed. Cir. 1992) (“It is impermissible to use the claimed invention as an instruction manual or ‘template’ to piece together the teachings of the prior art so that the claimed invention is rendered obvious.”); *American Airlines, Inc. and Southwest Airlines Co., v. Intellectual Ventures II LLC*, IPR2025-0093, Paper 11, 47

(PTAB November 21, 2025) (denying institution in part because “[w]ithout the challenged claims as a guide, we are not persuaded an ordinarily skill artisan at the critical time would have combined the prior art as suggested by Petitioner”). Ground 2 fails for this independent reason.

2. The prior art does not teach or suggest that the memory superpositions can be *selected*.

Ground 2 also fails because, contrary to Petitioners’ attempt to pull the claimed “selectable arrangement” from thin air, nothing in either reference (Stroud or Cragon) comes close to disclosing this key limitation. EX2021, ¶171. Instead, as discussed above, Cragon at most discloses various configurations that disk designers could consider during the design phase of the memory. Nothing Petitioners point to in Cragon discloses a *selectable* arrangement of memory types, much less a *selectable* arrangement of the four specific memory arrangements required by the claims.

Petitioners attempt to conjure a disclosure of selectability from Cragon by asserting that Cragon teaches “selecting the arrangement of memory based on workload parameters.” Pet., 73. The cited portions of Cragon, however, do not support their assertion. Petitioners’ first citation, to pages 00413-414, deals with techniques for protecting memory systems from unauthorized access, not selection of different memory arrangements. EX2021, ¶172. The cited pages in particular deal with the use of “bound registers.” But as Dr. Min explains, “bound registers” are

part of a processor and not part of the memory on a card. EX2021, ¶172. A POSITA would not understand Cragon’s high-level discussion of “bound registers” to be a teaching of a selectable arrangement. *Id.* Further, the portion of Dr. Alpert’s declaration that Petitioners cite in support largely parrots the Petition and does not explain how Cragon’s description of bound registers could constitute a teaching of selectability. EX1003, ¶217; *Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9 at 15-17 (PTAB Aug. 24, 2022) (designated precedential) (according “little weight” to declaration testimony that contains a verbatim restatement of a petition’s conclusory assertions without additional supporting evidence or reasoning).

Petitioners’ only other citation in support of their assertion that selectability would be obvious is to pages 000536-548 of Cragon—which Petitioners extensively rely on throughout Ground 2. But as Dr. Min explains, these pages are directed to general features of disks that designers take into account when *designing* disks—they do not teach or suggest any type of arrangement in which multiple memory types are present on a card and are available for selection **during system configuration or operation**.¹⁷ EX2021, ¶173. Nor do they disclose how such a

¹⁷ To the extent Petitioners theory relies on an interpretation of “selectable arrangement” that would permit selection to occur during the design phase of an information processing system—as some of Dr. Alpert’s testimony suggests—it is

system could be made or used. *Id.*, ¶174. And Cragon’s statement that “[t]here is very little solid information published on the question of the disk bandwidth and storage requirements based upon workloads” affirmatively undermines Petitioners’ implicit contention that a POSITA would know how to create such a selectable arrangement or have a reasonable expectation of success in doing so. EX1006, 000536; EX2021, ¶173.

At bottom, Petitioners’ Ground 2 hinges on an unexplained and unjustified logical leap based on vague, abstract characterizations of the prior art and conclusory expert testimony. Petitioners claim that because Stroud allegedly “aims to provide a flexible platform with user-definable modules configured to perform a wide variety of applications, it was obvious to select between common memory arrangements to suit the particular application.” Pet., 73-74 (citation omitted). This is not a reasoned obviousness theory—it is pure unsupported *ipse dixit* combined with rhetorical sleight of hand. Petitioners take abstract concepts like flexibility and scalability and manufacture a teaching or suggestion of *selectability* among memory arrangements—even though Stroud does not even disclose memory arrangements. EX2021, ¶167. Next, Petitioners point to Cragon, asserting that “Cragon teaches

unreasonably broad and fails as a matter of law for the reasons discussed above in the claim construction analysis. *Supra* Section VI.B.3.

several conventional memory arrangements for different purposes, *e.g.*, depending on parameters” and concluding that “[t]herefore, the combination teaches or suggests a selectable arrangement of the plurality of types of memory on Stroud’s card based on the needs of the application running on the card.” Pet., 74. Once again, this is pure *ipse dixit*, supported only by an equally conclusory and almost identical paragraph of Dr. Alpert’s declaration. *Id.*, citing EX1003, ¶218.

The simple fact is that neither Stroud nor Cragon disclose any selectable memory arrangements, much less the specific selectable arrangement required by limitation [7d], and the Petition does not contend otherwise. Instead, it asks the Board to read the “selectable arrangement” requirement into the prior art based on conclusory expert testimony and the general knowledge of a POSITA. This is precisely the type of obviousness theory the Office has identified as inconsistent with both the plain language of PTAB regulations and binding Federal Circuit caselaw. *See* EX2025 (July 31, 2025 Memorandum clarifying that “applicant admitted prior art (AAPA), expert testimony, common sense, and other evidence that is not ‘prior art consisting of patents or printed publications’ (collectively, ‘general knowledge’) may not be used to supply a missing claim limitation,” and citing the text of 37 C.F.R. § 42.104(b)(4) and *Qualcomm Inc. v. Apple Inc.*, 134 F.4th 1355,

1364-65 (Fed. Cir. 2025)).¹⁸ Ground 2 fails this threshold, black letter requirement that each element of each challenged claim must be present in a prior art patent or publication. Indeed, although ostensibly based on a combination of Stroud and Cragon, Ground 2 in fact relies on the wholesale creation of a structural limitation—a selectable memory arrangement—that is neither taught nor suggested by either of the references relied upon. This plainly violates 37 C.F.R. § 42.104(b)(4) and Federal Circuit precedent and thus fails as a matter of law.

3. Claims 9-10 are Patentable over Stroud and Cragon.

Because independent claim 7 is patentable over Stroud and Cragon, the more specific dependent claims are also patentable for the same reasons. EX2021, ¶175.

C. Ground 3 Does Not Remedy the Defects in Ground 2.

Implicitly recognizing that Ground 2 includes no disclosure of a “selectable arrangement,” Petitioners add ground 3, which is a backup ground that relies on Saulsbury purely for its alleged “additional teachings of selecting between memory modes.” Pet., 82. This ground fails as well, for multiple reasons.

¹⁸ Although the July 31, 2025 memorandum itself technically applies to petitions filed on or after September 1, 2025, it reflects the Office’s interpretation of the plain text of 37 C.F.R. § 42.104(b)(4) and precedential case law and those interpretations should apply here.

1. Saulsbury does not teach, alone or in combination, the combination of four distinct memory arrangements on a single card required by limitation [7d].

Tellingly, even with the benefit of a third reference, Petitioners have still failed to identify any prior art that discloses a single card that has the four claimed memory arrangements. Indeed, it is undisputed that that “Saulsbury doesn’t disclose all four of the claimed memory arrangements,” either on a single card or otherwise. EX2024, 93:11-20. In fact, Saulsbury does not identify *any* memory arrangements (only different memory types).¹⁹ EX2021, ¶177. Petitioners rely on Saulsbury solely for its disclosure of “*switch[ing] the memory mode between physical memory and cache memory [memory types], depending on the particular application of the processor and memory chip.*” Pet., 85 (emphasis in original, brackets added; citing EX1007 ¶¶3, 65, Fig. 1). But this disclosure of switching between memory types does nothing to remedy the core defects in the Stroud-Cragon combination outlined above because it does not disclose using the four claimed memory arrangements on a single card.

Instead, Ground 3 continues to rely on disparate disclosures in Cragon without providing an obviousness analysis for why a POSITA would combine them and, like

¹⁹ Again, “memory types” refer to different forms of memory, while “memory arrangements” are different ways of arranging those types of memory.

Ground 2, Ground 3 fails to provide any reason that a POSITA would choose the four enumerated memory types from among the hundreds of arrangements (and millions of sets of four arrangements) available. *See* Pet., 84-93. Ground 3 thus lacks a credible motivation to combine for all of the reasons stated above.

Further, Petitioner fails to show how Saulsbury's switching mechanism would or even could be operable on a single card containing four distinct memory arrangements. Dr. Alpert admitted that Saulsbury teaches only "switching between cache memory and physical memory." EX2024, 95:16-96:2. Further he acknowledged that "Saulsbury teaches using a control bit to select the arrangement of the memory" (EX2024, 96:13-14) and that "[a] bit can have two values." *Id.*, 98:6-19; *see also* Pet., 85 ("The arrangement is selected using a control bit."). Nothing in the Petition or Dr. Alpert's declaration explains how this limited disclosure of a binary control bit that can switch between two options could be modified by a POSITA to achieve the required four-way selectability of limitation [7d]. *See* Pet., 92. And in fact, as Dr. Min explains, Saulsbury's limited control bit could not be used to select among four different memory arrangements on a single card, as required by limitation [7d]. EX2021, ¶179. Thus, if anything, Saulsbury teaches away from having four distinct memory arrangements on a single card.

2. Saulsbury does not teach, alone or in combination, the claimed “selectable arrangement.”

Despite adding Saulsbury-based grounds to supply the “selectable arrangement” limitation, Petitioners’ proposed combinations still fail to do so. First, it is undisputed that the only switching Saulsbury teaches is “between physical memory and cache memory.” Pet., 85. But that is a disclosure of switching between *memory types, not memory arrangements*. As Dr. Min explains, memory types are simply different categories of memory structures, while memory arrangements are the wide variety of methods for integrating and utilizing various types of memory to achieve optimal system performance depending on design objectives. EX2021, ¶181; EX1001, 7:40-46 (“Analogous to the overlapping of waves to form a final observed wave in wave superposition, where each wave contributes to the whole, in this case different types of memory are being overlapped, each with benefits and drawbacks, to form a final memory system which performs optimally in a given video stream or other data flow/data processing task.”).

Petitioners never address this distinction. Instead, they treat Saulsbury’s disclosure of switching between memory types as tantamount to a general suggestion that anything related to memory can or should be selectable. *See, e.g.*, Pet., 85, 90-91. But, as Dr. Min explains, a POSITA would understand Saulsbury’s switching between memory types as a fundamentally different concept from the four-way, selectable memory *arrangements* required by the claims. EX2021, ¶182.

Further, as noted above, the Petition and Dr. Alpert's declaration are silent as to how Saulsbury's control bit could be used to control selection of the four arrangements required by the claims. Thus Ground 3's addition of Saulsbury does nothing more than add the general concept of selectability—it does not disclose a card with the four required memory arrangements selection among memory arrangements, or even a method that would work to select between four memory arrangements. Petitioners' attempt to limitation 7[d] by patching together unrelated disclosures in Stroud, Cragon, and Saulsbury is pure hindsight.

3. Claims 9-10 are Patentable over Stroud, Cragon, and Saulsbury.

Because independent claim 7 is patentable over Stroud, Cragon, and Saulsbury the more specific dependent claims are also patentable for the same reasons. EX2021, ¶184.

D. Ground 4 Does Not Remedy the Defects in Grounds 1, 2, or 3.

Ground 4 simply combines Petitioners' Stroud and DaSilva arguments from ground 1 with their Stroud, Cragon, and Saulsbury arguments from ground 3 without any additional argument or evidence. Pet., 93-94. The flaws with those grounds discussed above thus apply in full force to ground 4. In particular, ground 4 does not render obvious claim 4 both because it does not render obvious the "module identifier" limitation of claim 1, *see* Section VII.A.1, and because it does not render obvious any of the memory arrangement limitations of claim 4, *see* Sections VII.B,

VII.C. Ground 4 does not render obvious claim 8 both because it does not render obvious any of the memory arrangement limitations of claim 7, *see* Sections VII.B, VII.C, and because it does not render obvious the “module identifier” of claim 8, *see* Section VII.A.1. Finally, ground 4 does not render obvious claim 15 both because it does not render obvious the “module identifier” limitation of claim 13, *see* Section VII.A.1, and because it does not render obvious any of the memory arrangement limitations of claim 15, *see* Sections VII.B, VII.C.

VIII. CONCLUSION

For the reasons stated above, the Board should find that Petitioners failed to carry their burden of proving unpatentability as to all grounds in the Petition.

Dated: December 19, 2025

By: /s/ Michael Matulewicz-Crowley
Michael Matulewicz-Crowley
PTO Reg. No. 72,220
Counsel for Patent Owner

CERTIFICATE OF COMPLIANCE

The undersigned hereby certifies that Patent Owners' Response (the "POR") consists of 13,288 words, excluding the table of contents, certificate of service, this certificate of compliance, or table of exhibits. The POR complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, counsel has relied on the word count of the word-processing system used to prepare this paper (Microsoft Word).

Dated: December 19, 2025

/s/ Michael Matulewicz-Crowley
Michael Matulewicz-Crowley
Counsel for Patent Owner

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6, the undersigned certifies that on December 19, 2025, a copy of the foregoing document was served by email upon the following:

Harper Batts (Reg. No. 56,160)
hbatts@sheppardmullin.com
legaltm-amd-concurrent@sheppardmullin.com
SHEPPARD MULLIN LLP
1540 El Camino Real, Suite 120
Menlo Park, CA 94025
T 650-815-2673 / F 650-815-4668

Christopher Ponder (Reg. No. 77,167)
cponder@sheppardmullin.com
legaltm-amd-concurrent@sheppardmullin.com
SHEPPARD MULLIN LLP
1540 El Camino Real, Suite 120
Menlo Park, CA 94025
T 650-815-2676 / F 650-815-4664

Jeffrey Liang (Reg. No. 69,043)
jliang@sheppardmullin.com
legaltm-amd-concurrent@sheppardmullin.com
SHEPPARD MULLIN LLP
1540 El Camino Real, Suite 120
Menlo Park, CA 94025
T 650-815-2685 / F 650-815-4666

/s/ Michael Matulewicz-Crowley
Michael Matulewicz-Crowley
Counsel for Patent Owner

CLAIM APPENDIX

The language of claims 1, 3, and 7 of the '753 patent is reproduced below. For convenience, Patent Owners have included the limitation labels used by Petitioners.

Claim 1:

[1pre] A platform for data flow processing, comprising:

[1a] one or more swappable pods or cards in one or more chassis,
coupled through a messaging interface network;

[1b] each of the one or more swappable pods or cards having one or more
hardware modules or one or more software modules;

[1c] one or more of the plurality of swappable pods or cards having a
portion for user-definable hardware;

[1d] the plurality of swappable pods or cards being user-configurable to
implement data flow processing architectures; and

[1e] a network coupled to the one or more swappable pods or cards and
supporting messaging-based communication using packets each having a header
with a chassis identifier, a board identifier, a module identifier, an instance
identifier, and a type identifier, so that each type of module, each instance of a
type of module, and each module on each board in each chassis can be addressed
through the header.

EX1001, cl. 1; *see also* Pet., 98 (claim appendix).

Claim 3:

[3] The platform of claim 1, wherein the one or more swappable pods or cards comprises at least one processor card having a at least one processor allocatable for offloading hardware tasks to software, orchestrating data flows, orchestrating configurations of hardware modules, and executing in cooperation with a sandbox of one of the plurality of swappable pods or cards.

EX1001, cl. 3; *see also* Pet., 99 (claim appendix).

Claim 7:

[7pre] A method for data flow processing in a platform, comprising:

[7a] configuring a plurality of swappable pods or cards in one or more chassis, coupled through a messaging interface network, to implement a data flow processing architecture;

[7b] configuring user-definable hardware modules or user-definable software modules in each of one or more of the plurality of swappable pods or cards;

[7c] communicating a data flow in accordance with the implemented data flow processing architecture among one or more hardware modules or one or more software modules in each of the plurality of swappable pods or cards, including the user-definable hardware modules or user-definable software modules; and

[7d] selecting a memory superposition in at least one of the plurality of swappable pods or cards having a plurality of types of memory with selectable arrangement of the plurality of types of memory as multilevel cache, interleaved memory, swapped memory over a specified address range, and memory allocated to specified address ranges according to parameterized task needs.

EX1001, cl. 7; *see also* Pet., 99–100 (claim appendix).