

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS CO., LTD. and
SAMSUNG ELECTRONICS AMERICA, INC.,
Petitioner,

v.

RADIAN MEMORY SYSTEMS, LLC,
Patent Owner.

Case IPR2025-01289
Patent 11,681,614

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	PETITIONER’S ELLIS GROUNDS FAIL.	2
A.	Petitioner Does Not Show A Motivation To Combine With A Reasonable Expectation Of Success.	3
1.	Petitioner Modifies Ellis Without Explanation.....	4
2.	Petitioner Does Not Explain, Let Alone Motivate, How Son’s “SSD Module” Is Involved In The Combined System.....	8
a)	Petitioner Does Not Explain How Son’s SSD Module Is Involved In Any “management command from a host.”	9
b)	Petitioner Does Not Explain How Son’s SSD Module Is Involved In Any “query command from the host.”	16
B.	Petitioner Does Not Show That Ellis And Son Teach A Storage Device “operable to receive a management command from a host” (Ground 1, All Challenged Claims; Ground 2, All Challenged Claims Except 21-27 And 29; Ground 3, All Challenged Claims Except Claim 30).....	19
1.	Ellis Does Not Teach The Claimed Management Command “from a host.”	19
2.	Petitioner Does Not Show That Son Teaches The Claimed Management Command “from a host” In Its Combination. ..	22
C.	Petitioner Does Not Show That Ellis And Son Teach Logic Operable To Return “information about the zones in response to a query command from the host” (Grounds 1-3, All Challenged Claims).	24
III.	PETITIONER’S TAKEO GROUNDS FAIL.	30
A.	Petitioner Does Not Show A Motivation To Combine With A Reasonable Expectation Of Success.	31

1.	Petitioner’s Takeo Combinations Are Not Specified (Grounds 4-6, All Challenged Claims).....	31
a)	Petitioner Does Not Explain How Its Takeo Combinations Receive A “management command from a host.”	32
b)	Petitioner Does Not Explain Why A POSITA Would Modify Takeo To Return Its Alleged “information about [] zones” (Grounds 4-6, All Challenged Claims).....	38
(1)	Petitioner’s Ellis-Based Argument Is Unsupported.	38
(2)	Petitioner’s Sinclair-Based Argument Rests On A Faulty Analogy.	39
B.	Petitioner Does Not Show That Its Takeo Combinations Teach A “storage device . . . operable to receive a management command from a host” (Ground 4, All Challenged Claims; Ground 5, All Challenged Claims Except 21-27 and 29; Ground 6, All Challenged Claims Except Claim 30).....	48
C.	Petitioner Does Not Show That Its Takeo Combinations Teach Logic Operable To Write To Erase Units “using a common page address” (Grounds 5-6, All Challenged Claims).	53
1.	Sinclair Does Not Teach This Limitation In Petitioner’s Combinations.....	53
2.	Petitioner’s Analogy To Olbrich Does Not Close This Gap In Petitioner’s Analysis.	56
IV.	CONCLUSION	58

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

<i>ActiveVideo Networks, Inc. v. Verizon Commc'ns, Inc.</i> , 694 F.3d 1312 (Fed. Cir. 2012)	36
<i>Belden Inc. v. Berk-Tek LLC</i> , 805 F.3d 1064 (Fed. Cir. 2015)	7
<i>Harmonic Inc. v. Avid Tech., Inc.</i> , 815 F.3d 1356 (Fed. Cir. 2016)	3
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006)	39
<i>In re Kotzab</i> , 217 F.3d 1365 (Fed. Cir. 2000)	7, 16
<i>In re Kubin</i> , 561 F.3d 1351 (Fed. Cir. 2009)	7
<i>In re Magnum Oil Tools Int'l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016)	18, 52, 56
<i>KSR Int'l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007).....	39
<i>Metalcraft of Mayville, Inc. v. Toro Co.</i> , 848 F.3d 1358 (Fed. Cir. 2017)	36
<i>PersonalWeb Techs. LLC v. Apple, Inc.</i> , 848 F.3d 987 (Fed. Cir. 2017)	3
<i>TQ Delta, LLC v. Cisco Sys.</i> , 942 F.3d 1352 (Fed. Cir. 2019)	56

<i>Virtek Vision Int'l ULC v. Assembly Guidance Sys.,</i> 97 F.4th 882 (Fed. Cir. 2024)	31
--	----

AGENCY DECISIONS

<i>Apple, Inc. v. Voip-pal.com, Inc.,</i> IPR2016-01198, Paper 53 (Nov. 20, 2017)	23
<i>Intel Corp. v. VLSI Tech. LLC,</i> IPR2019-01196, Paper 11 (Jan. 7, 2020)	56
<i>Markforged Inc., v. Continuous Composites, Inc.,</i> IPR2022-01220, Paper 8 (Jan. 11, 2023)	37
<i>Matterport, Inc. v. Appliance Computing III, Inc.,</i> IPR2021-00939, Paper 15 (Nov. 23, 2021)	58
<i>Samsung Elecs. Co., Ltd. v. Dynamics Inc.,</i> IPR2020-00499, Paper 63 (Aug. 10, 2021)	47
<i>Samsung Elecs. Co., Ltd. v. Red Rock Analytics, LLC,</i> IPR2018-00556, Paper 18 (Aug. 20, 2018)	56
<i>UiPath, Inc. v. Rule 14 LLC,</i> IPR2024-01304, Paper 8 (Mar. 14, 2025)	33
<i>Valeo N. Am., Inc. v. Magna Elecs., Inc.,</i> IPR2014-01206, Paper 13 (Dec. 23, 2014)	57
<i>Vectair Sys., Inc. v. Fresh Prods., Inc.,</i> IPR2024-00824, Paper 9 (Nov. 12, 2024)	52

REGULATIONS

37 C.F.R. § 42.65(a)	56
----------------------------	----

EXHIBIT LIST	
2001	<i>Reserved</i>
2002	<i>Reserved</i>
2003	<i>Reserved</i>
2004	<i>Reserved</i>
2005	Declaration of Mike Jadon, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. May 6, 2025), Dkt. 43-13 [Jadon-Decl.]
2006	Laura Caulfield, <i>Project Denali to define flexible SSDs for cloud-scale applications</i> , MICROSOFT AZURE BLOG (Mar. 20, 2018), https://azure.microsoft.com/en-us/blog/project-denali-to-define-flexible-ssds-for-cloud-scale-applications/ [Project-Denali-Article].
2007	Defendants Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc.'s Amended Answer, Defenses and Counterclaims, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. Sept. 8, 2025), Dkt. 67 [Amended-Answer]
2008	Memorandum re: Interim Processes for PTAB Workload Management (Mar. 26, 2025) [3/26/25 Memo]
2009	Plaintiff's Complaint for Patent Infringement <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. Dec. 24, 2024) Dkt. 1 [Complaint]
2010	Order On Plaintiff's Motion to Withdraw Its Motion For Preliminary Injunction and Vacate The Hearing Set Before The Court, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. July 11, 2025), Dkt. 64 [Order]
2011	Plaintiff's Opposed Motion to Dismiss, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. July 17, 2025), Dkt. 69 [Motion-to-Dismiss]

2012	First Amended Docket Control Order, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. Apr. 14, 2025), Dkt. 32 [Docket-Control-Order]
2013	Samsung's P.R. 3-3 And 3-4 Invalidity Contentions, <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. June 18, 2025 [Invalidity-Contentions]
2014	Statement of Interest of the United States of America <i>Radian Memory Systems LLC v. Samsung Electronics Co., Ltd. et al.</i> , 2-24-cv-01073 (E.D. Tex. June 24, 2025) Dkt. 52 [US Statement of Interest]
2015	Press Release, TrendForce, NAND Flash Revenue Surged Over 20% in 2Q25, SK Group Market Share Jumped to 21%, Says TrendForce (Aug. 28, 2025), https://www.trendforce.com/presscenter/news/20250828-12688.html [TrendForce]
2016	[How much do you know about flash memory? - Part 4] The World's No.1 Flash Memory Company, Samsung Electronics, and Their Unending Journey, Samsung Tech Blog (June 29, 2020), https://semiconductor.samsung.com/news-events/tech-blog/all-there-is-to-know-about-flash-memory-part4-samsung-electronics-world-no1-in-flash-memory-still-learning-still-innovating/ [Samsung-Article]
2017	<i>Reserved</i>
2018	Median and Average Time To Trial for Judge Rodney Gilstrap, E.D. Tex., Docket Navigator [Gilstrap Time to Trial]
2019	<i>Reserved</i>
2020	Declaration of Dr. Mark Jones, Ph.D. [Jones-Decl.]
2021	U.S. Patent No. 8,949,689
2022	U.S. Patent No. 9,021,231

2023	U.S. Patent No. 9,098,399
2024	U.S. Patent No. 9,123,445
2025	U.S. Patent No. 9,170,941
2026	U.S. Patent No. 9,183,137
2027	U.S. Patent No. 9,244,519
2028	U.S. Patent No. 9,543,025
2029	U.S. Patent No. 9,671,962
2030	U.S. Patent No. 10,546,648

All emphases in quotations are added unless otherwise stated.

I. INTRODUCTION

The Petition should be denied because it fails to demonstrate a reasonable likelihood of prevailing as to even one challenged claim.

Petitioner's theories combine references in a piecemeal fashion and are driven by hindsight. Instead of articulating any coherent architecture, Petitioner shuffles technical responsibilities and identifications of prior art elements between the claimed "host" and the claimed "storage device" without attending to how its combined systems would work, or why a POSITA would have made Petitioner's choices (or rather, *some* specified set of choices compatible with what Petitioner argues about its combinations) with a reasonable expectation of success. Petitioner's arguments lean on boilerplate disclosures within multiple references and re-use rationales across different combinations—at some times treating the claims as a roadmap, and at other times simply ignoring *where* inputs relevant to components of the systems it is arguing about would be stored, *how* they would be accessed or updated, and *how* commands and data would flow between them.

Multiple claim requirements are not taught or rendered obvious by the cited art. In particular, Petitioner does not show a storage device "operable to receive a management command from a host," or show the claimed "logic to return information about the zones in response to a query command from the host." Petitioner's analogies about how these limitations might appear within their

combinations are untethered to how the cited references actually operate. Petitioner's hindsight-driven analysis falls well short of meeting its burden, and the Board should not institute trial.

II. PETITIONER'S ELLIS GROUNDS FAIL.

All of Petitioner's Ellis grounds rely upon Petitioner's combination of Ellis and Son¹ ("Ground 1") against independent claim 1. *See, e.g.*, Pet. 33-41 (Petitioner's Ground 2 referring to Petitioner's Ground 1 for all aspects of independent claims), 43-51 (Petitioner's Ground 3 addressing dependent claims only), 28-31 (Petitioner's Ground 1 relying upon its arguments as to claim 1 for all other claims challenged by this ground, which depend from claim 1).

As the following sections show, Petitioner never supplies a coherent reason to modify Ellis in the way it proposes—and even if one takes Petitioner's own identification with claim terms with aspects of Ellis for granted, Petitioner never shows the claimed host-issued "management command" or logic to "return

¹ Although the Petition refers to Sinclair and Son as Exhibits "1005" and "1006," respectively, *see, e.g.*, Pet., vi, Petitioner marked and submitted these documents to the Board as Exhibits 1006 and 1005, respectively. Patent Owner will use the numbering of the exhibits as marked and submitted to the Board with additional parentheticals in citations to avoid confusion.

information about the zones in response to a query command from the host” to be present in the combined system.

A. Petitioner Does Not Show A Motivation To Combine With A Reasonable Expectation Of Success.

Petitioner’s Ellis grounds should be rejected because Petitioner does not adequately explain how its references would be combined. The Federal Circuit has made clear, an obviousness finding cannot stand without a “clear[] expla[nation]” of “*how* the combination of the two references was supposed to work.” *PersonalWeb Techs. LLC v. Apple, Inc.*, 848 F.3d 987, 994 (Fed. Cir. 2017) (emphasis in original). Indeed, “a clear, evidence-supported account of the contemplated workings of the combination *is a prerequisite* to adequately explaining and supporting a conclusion that a relevant skilled artisan would have been motivated to make the combination and reasonably expect success in doing so.” *Id.* Petitioner provides no such account here as to any of its grounds.

All of Petitioner’s Ellis grounds rely upon the combination of Ellis and Son. Yet Petitioner never actually specifies what the Ellis/Son combination is. Instead, it only offers element-by-element outcomes of supposed “modifications of Ellis,” without ever describing a coherent system. *See, e.g.*, Pet., 14, 16, 24. This is insufficient. *See, e.g., Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016) (“[I]t [is] [Petitioner’s] burden to explain to the Board how Haskell

combined with Rossmere rendered the challenged claims unpatentable.”).

The gaps in Petitioner’s combinations are all apparent in connection with claim 1, where Petitioner describes its combination only by implication and only in connection with selected limitations. But the flaw runs deeper than any one limitation: Petitioner’s piecemeal suggestions about its combinations are conclusory, at odds with Ellis’s and Son’s express disclosures, and never result in a clearly defined system—let alone one a skilled artisan would have been motivated to pursue with a reasonable expectation of success.

1. Petitioner Modifies Ellis Without Explanation.

Petitioner first begins arguing obviousness (rather than express disclosure in Ellis) in connection with the limitation of claim 1 that recites a storage device “operable to receive a management command *from a host*.” Pet. 11.² In connection

² Although not every claim recites a “management command from a host,” Petitioner incorporates its arguments about this limitation into an attempt to motivate its Ellis combinations when addressing a “query command from [a] host” limitation (Pet. 24) that *does* appear (in some form) in every challenged claim, Ex. 1001 [’614] 64:62, 65:60-61, 67:16-17. All of Petitioner’s grounds address this latter limitation solely by reference to Petitioner’s claim 1, Ground 1 argument, *see, e.g.*, Pet. 35, 39,

with this limitation, Petitioner identifies “wear leveling and garbage collection commands” within Ellis as alleged “management commands” in the sense of claim 1. Pet., 11. Petitioner then notes that in Ellis, a flash translation layer (“FTL”) within Ellis’s *memory controller* “performs maintenance operations on ‘the superblocks,’ such as ‘wear leveling.’” Pet., 11. This is a problem for Petitioner’s mapping onto claim 1, because claim 1 requires its “management command” to come “from a host,” and Ellis’s memory controller is not Ellis’s host. *See* Pet., 8, 11-12 (implicitly identifying Ellis’s host system—not Ellis’s controller—with the claimed “host”). *See also* Section II.B.1 (showing that this limitation is not taught in Petitioner’s Ellis combinations).

To be clear, Petitioner expressly recognizes that Ellis nowhere discloses wear leveling or garbage commands originating from anything that Petitioner identifies as a “host.” *See* Pet., 11 (arguing only that “the *memory controller* [in Ellis] may issue wear leveling and garbage collection commands”). Petitioner says, however, that “Ellis states that *either* ‘the memory controller *[or] the host system* ... can operate any of the modules and functions of the memory controller.” Pet., 11 (citing Ex. 1004 [Ellis] 4:58-65). Here, Petitioner proposes *modifying* Ellis, and does so

and Petitioner does not introduce (let alone motivate) any combination specific to claims that do *not* recite a “management command from a host.”

without explanation. The disclosure in Ellis that Petitioner is partially quoting from reads more fully as follows:

The functional units in the memory controller 104 can work individually and independently of the other functional units. For illustrative purposes, the memory controller 104 is described by operation of the memory controller 104 with the host system 108 and the memory array 106. It is understood that the memory controller 104, the host system 108, and the memory array 106 can operate any of the modules and functions of the memory controller 104.

Ex. 1004 [Ellis] 4:58-65. This statement has nothing to do with how wear leveling or garbage collection is performed in Ellis. Indeed, this generic statement is not even specific to Ellis: the same disclosure appears, verbatim or near verbatim, in at least ten unrelated patent applications filed by the initial assignee of Ellis, most of which list other sets of named inventors.³ This is boilerplate designed to ensure that the

³ See, e.g., Ex. 2021 [U.S. Patent No. 8,949,689] 4:32-39; Ex. 2022 [U.S. Patent No. 9,021,231] 4:60-67; Ex. 2023 [U.S. Patent No. 9,098,399] 4:66-5:6; Ex. 2024 [U.S. Patent No. 9,123,445] 4:43-49; Ex. 2025 [U.S. Patent No. 9,170,941] 9:29-37; Ex. 2026 [U.S. Patent No. 9,183,137] 4:58-65; Ex. 2027 [U.S. Patent No. 9,244,519] 6:41-49; Ex. 2028 [U.S. Patent No. 9,543,025] 6:3-10; Ex. 2029 [U.S. Patent No. 9,671,962] 4:58-65; Ex. 2030 [U.S. Patent No. 10,546,648] 4:49-56.

scope of a patent publication (like Ellis) is not generally limited to the specifics of disclosed embodiments. And this statement does not analyze any trade-offs, performance implications, or engineering considerations that might generally inform whether a host-based or controller-situated functionality is preferable in any system, let alone suggest to the POSITA any reason for changing what Petitioner elsewhere acknowledges: that in the system Ellis expressly discloses, it is “the ***memory controller*** [that] may issue wear leveling and garbage collection commands.” Pet., 11.

This paragraph is the only alleged support in Ellis that Petitioner identifies for modifying Ellis so that wear leveling commands might come “from a host.” See Pet., 11-12. But even in the context of a single reference, a motivation must be shown for combining features of different embodiments, or otherwise modifying the teachings of a reference. See, e.g., *In re Kotzab*, 217 F.3d 1365, 1370 (Fed. Cir. 2000). Petitioner fails to do that here. See, e.g., *In re Kubin*, 561 F.3d 1351, 1359 (Fed. Cir. 2009) (“[W]here a defendant merely throws metaphorical darts at a board filled with combinatorial prior art possibilities, courts should not succumb to hindsight claims of obviousness.”); *Belden Inc. v. Berk-Tek LLC*, 805 F.3d 1064, 1073 (Fed. Cir. 2015) (“[O]bviousness concerns whether a skilled artisan not only *could have made* but *would have been motivated to make* the combinations or modifications of prior art to arrive at the claimed invention.”) (emphasis in original).

2. Petitioner Does Not Explain, Let Alone Motivate, How Son's "SSD Module" Is Involved In The Combined System.

Because Petitioner effectively admits that the claimed “management command from a host” and “query command from [a] host” are not disclosed in Ellis, all of Petitioner’s Ellis grounds turn to Son—more specifically, disclosures concerning the operation of Son’s “SSD module”—for disclosures of how claimed “command[s]” might come “from a host” in a combination involving Ellis and Son. *See, e.g.*, Pet. 17–24 (Petitioner’s Ground 1 analysis of a “query command” limitation repeatedly discussing Son’s SSD module), 35 (incorporating this analysis into a challenge to independent claim 10), 39 (same, for independent claim 21). But Petitioner never identifies *how* aspects of Son’s SSD module are actually used to modify Ellis, let alone how the combined system issues commands “from a host.”

Instead, Petitioner pairs disclosures about Son’s SSD module with statements like “[a] POSITA would be motivated to combine Son’s teachings of a host query for returning information about superblocks [zones] with Ellis’s storage device,” or the statement that “Son discloses the benefits of returning the information about zones to the host so that the host may utilize the information to issue management commands.” *See, e.g.*, Pet., 24-25. These statements never identify how much (if any) of “Ellis’s storage device” remains in its combinations, or how much of Son’s SSD module is present in the “host” of its combinations, or what other modifications

are made to Ellis and Son so that “Son’s teachings” provide any claimed “command” in the combined system. As discussed in greater detail in connection with exemplary limitations of claim 1 below, such omissions are fatal to Petitioner’s attempt at showing a motivation to combine.

a) *Petitioner Does Not Explain How Son’s SSD Module Is Involved In Any “management command from a host.”*

Claim 1 recites a storage device “operable to receive a **management command from a host.**”⁴ Petitioner maps Ellis’s “memory system” (102 in Ellis’s Figure 1) to the “storage device,” Pet. 8, Ellis’s “host system” (108 in Ellis’s Figure 1) to the “host,” *see, e.g.*, Pet., 11-12, and wear leveling commands to the “management commands.” *See id.* But, as discussed previously, Ellis’s controller performs flash maintenance, not its host. *See* Pet., 11; Section II.A.1. To bridge that gap, Petitioner cites a generic sentence about modules being “operat[ed]” by either the controller or host and then relocates maintenance to the host—without articulating any implementation or rationale. *See id.*

⁴ Although not every claim recites a “management command from a host,” Petitioner incorporates its arguments about this limitation into its attempt to motivate its Ellis combinations as to a “query command from [a] host” limitation, which does appear in every challenged claim. *See supra* note 2.

Petitioner then turns to Son and points to Son's SSD module performing "maintenance operations including garbage collection and wear leveling" for flash memory present in Son's SSD array (145 in Son's Figure 1). *See* Pet., 12-13. Implicitly, when Son's SSD module initiates these operations, it does so by sending instructions to Son's controller, which directly accesses Son's SSD array. According to Petitioner, the controller's receipt of any such commands from Son's SSD module would teach the claimed "storage device operable to receive a management command from a host." *See id.* Petitioner further argues that Son teaches that "Ellis can be improved by configuring an 'SSD module' of a host to perform management operations," Pet., 14, and that Son allegedly motivates "modifying Ellis's device so that one or more management commands for the superblocks are issued by the host." Pet., 15.

To frame its proposed modification of Ellis in terms of where alleged "management commands" come from, Pet., 15—a formulation that plainly draws inspiration from claim language reciting "a management command from a host"—is to understate the technical implications of what Petitioner proposes to change in Ellis. Ex. 2020 [Jones-Decl.] ¶ 43. In Ellis, a memory controller performs both wear leveling and garbage collection without being commanded to do so by its host, and Ellis maintains all metadata needed to carry out these operations out within its controller—as Petitioner appears to recognize. *See, e.g.,* Pet., 11 (Petitioner

observing that “Ellis discloses that *the memory controller* 104 [of Ellis’s Figure 1] can include a flash translation layer (FTL),” and that “[t]he *FTL performs maintenance operations* . . . such as wear leveling”) (internal quotations omitted); *see also* Ex. 2020 [Jones-Decl.] ¶ 44.

Although Petitioner proposes “modifying Ellis’s device so that one or more management commands for the superblocks are issued by the host,” Pet., 15, Petitioner does not identify the *basis* on which Ellis’s modified host would issue such commands. In particular, Petitioner does not specify whether the host in its combination issues wear leveling commands using the same inputs that Ellis’s controller would use, or using the same inputs that Son’s SSD module would use, or some other possibility—and Petitioner does not identify where any relevant inputs are stored in its combination. As Dr. Jones explains:

When Petitioner speaks of “moving . . . management *operations* to the host,” Pet. 14 (emphasis added), Petitioner does not say how this “moving” would be reflected in changes to how Ellis or Son store, update, and access relevant *inputs* to those operations. This language does not, for example, identify what data structures are used for “management operations” in the combined system—whether it is Ellis’s controller-resident data structures (used by Ellis for what Petitioner identifies as “management operations”), the tables managed by Son’s host-side SSD module (used by Son for what Petitioner

identifies as “management operations”), or some unexplained hybrid of the two.

Ex. 2020 [Jones-Decl.] ¶ 46.

A POSITA would also understand that modifying where *commands* to perform wear leveling and garbage collection operations come from—such as moving the issuance of such commands from Ellis’s controller to Ellis’s host system, or any “host” in a combination involving Ellis—would likely also involve shifting at least some aspects of the underlying *tasks* of wear leveling and garbage collection to the host. Ex. 2020 [Jones-Decl.] ¶ 47. Such a shift would likely involve, at a minimum, modifying Ellis so that its host system has access to whatever flash maintenance information the system uses to decide when to initiate wear leveling and garbage collection. *Id.*

Petitioner does not identify how its combination would do this, although Petitioner’s individual references indicate the sorts of inputs that might be implicated. As one example, Son offers “whether one or more superblocks have a wear count above a predetermined threshold value” (Ex. 1005 [Son] [0079]) as a threshold for deciding whether to initiate wear leveling, and “whether the number of free superblocks in the SSD falls below a predetermined threshold” (Ex. 1005 [Son] [0064]) as a threshold for deciding whether to initiate garbage collection. Petitioner does not identify any disclosure in Ellis that suggests *Ellis’s* host system has access

to these inputs. As another example, Ellis discloses data such as “a number of times or cycles the memory devices ... associated with ... super blocks ... are programmed or erased” and “a valid-page count,” but expressly locates this information inside “metadata pages” managed by a “block control module” *within Ellis’s controller*, not Ellis’s host system. Ex. 1004 [Ellis] 4:66-5:3 (“The memory controller . . . can include . . . a block control module”), 11:58-67 (locating program/erase cycle counts and valid page counts in metadata pages “managed by the block control module”).

Despite these differences between Ellis and Son, Petitioner does not specify whether the “host” in its Ellis/Son combination would somehow *itself* maintain inputs relevant to wear leveling and garbage collection, or whether its “host” would obtain that information in some way from Ellis’s controller (or a modified version of Ellis’s controller). And as Dr. Jones explains, Ellis and Son do not point in the same direction on this issue.

In Ellis, inputs relevant to wear leveling and garbage collection are maintained and updated only at Ellis’s controller, while in Son, a host-side “SSD module” maintains and updates this information. Petitioner’s description of its combination is thus consistent with at least two architectures.

As one possibility, Ellis’s controller could continue to track and update metadata related to wear leveling and garbage collection, as a POSITA would understand that it does in the unmodified form of Ellis. But in this case, the basis on which the modified host would issue

commands for those operations is not clear. Petitioner does not, for example, identify how the host would be able to access the information it would use to issue commands relating to wear leveling or garbage collection.

Another possibility is that the memory controller in the combined system is effectively relieved of any responsibility for wear leveling and garbage collection in the combination. Son expressly envisions this possibility. *See, e.g.*, Ex. 1005 [Son], [0063] (“The SSD module performs various SSD maintenance operations including garbage collection and wear leveling, which can relieve the SSD controller and firmware of these responsibilities.”). In this type of combination of Ellis and Son, the “host” of the combination might more closely resemble Son’s “SSD module,” in that it would not only issue *commands* that initiate wear leveling and garbage collection, but also track and update the *inputs* used to decide when those operations take place.

Ex. 2020 [Jones-Decl.] ¶¶ 50-52. As Dr. Jones further explains, while the latter possibility is expressly contemplated by Son, this possibility would raise questions about how the controller in Petitioner’s combination differs from Ellis’s controller.

Son expressly envisions the use of its “SSD module” with memory controllers that are not tasked with wear leveling and garbage collection, by disclosing, for example, that “[i]n some embodiments, the controller and firmware are configured differently from many conventional systems in that *they do not perform onboard garbage collection or wear-leveling*, but rely upon the SSD module to manage

such operation,” Ex. 1005 [Son] [0017], and by containing a claim directed to a system where “the controller and firmware ***do not include garbage collection logic or wear leveling logic.***” *Id.*, cl. 4; *see also id.* [0063] (“The SSD module performs various SSD maintenance operations including garbage collection and wear leveling, which can relieve the SSD controller and firmware of these responsibilities.”) These disclosures about “some embodiments,” of course, do not rule out the use of Son’s SSD module with controllers (like the controller in Ellis) that ***do*** include garbage collection and wear leveling logic.

In the hypothetical case of a combination in which Son’s “SSD module” entirely takes over garbage collection and wear leveling, however, it is unclear exactly what aspects of Ellis’s unmodified host system and Ellis’s unmodified controller (if any) would remain in the combination, or why they would be retained in the combination.

Ex. 2020 [Jones-Dec1.] ¶¶ 52-53.

The Petition is expressly ambiguous on this point when it argues that “[m]oving ***some or all*** of the management operations to the host helps ‘relieve the SSD controller and firmware of these responsibilities.’” Pet. 14 (quoting Ex. 1005 [Son] [0063]) (emphasis added). Here, Petitioner does not specify ***which*** “management operations” are “mov[ed]” to the host of its combined system, or even choose between suggesting that only “some” commands are moved (without identifying which ones), and that “all” commands are moved (whatever “the management operations” are). Petitioner’s “some or all” also does not show why

relieving the controller of responsibility of any “management operations” would be desirable, let alone why a POSITA would be motivated to make such changes in the first place: with “some or all,” Petitioner effectively declares that these particulars are immaterial. This is not analysis, but an admission that the proposal lacks both a defined scope and a motivating rationale.

Petitioner thus fails to show a motivation to combine. *See, e.g., In re Kotzab*, 217 F.3d at 1371 (“a rejection cannot be predicated on the mere identification in [prior art] of individual components of claimed limitations. Rather, particular findings must be made as to the reason the skilled artisan, with no knowledge of the claimed invention, would have selected these components for combination in the manner claimed.”).

b) *Petitioner Does Not Explain How Son’s SSD Module Is Involved In Any “query command from the host.”*

Petitioner’s failure to motivate its Ellis combinations is also shown in connection with the claimed “logic to return information about the zones in response to a query command from the host”—a limitation that is present (in different wordings) in all challenged claims, *see, e.g.,* Ex. 1001 [’614] 64:61-62 (independent claim 1), 65:58-61 (independent claim 9), 67:15-18 (independent claim 21). Petitioner directly analyzes this limitation only in connection with claim 1, dealing with its appearance in other claims by reference only. *See, e.g.,* Pet., 35, 39.

Petitioner identifies the claimed “logic to return information about the zones” within Ellis’s “memory system 102,” Pet., 8 (identifying the “storage device” of claim 1 that comprises the claimed logic as Ellis’s memory system), and argues that interactions between Son’s “SSD module” with Son’s “management tables” would teach this limitation. *See generally* Pet., 17-21. (Although Petitioner initially purports to limit its arguments about Son “[t]o the extent Ellis does not expressly disclose returning the information recited in this limitation,” Pet., 17, Petitioner nowhere separately argues that Ellis alone discloses or teaches the claimed “query command from a host.” *See generally id.*)

Petitioner analogizes the claimed “query command from a host” to interactions between Son’s “SSD module” and data stored in various structures. Petitioner describes various instances of Son’s SSD module accessing the contents of “management tables” as depicted (for example) in Son’s Figures 3A-3D, *see, e.g.*, Pet., 17-18 (citing Son’s Figure 3C), 20, 21-22 (citing Son’s Figure 3C), 23 (citing Son’s Figures 3B, 3D) and “management data structures” as depicted (for example) in Son’s Figure 4. Pet. 18-19. Of course, none of these data structures in Son exist in *Ellis’s* memory system, and Petitioner never states that they would exist in Petitioner’s underspecified combination of Ellis and Son. Petitioner is arguing by pure analogy: because Son’s SSD module accesses these data structures in Son, a POSITA would have been motivated to task the host in Petitioner’s combination with

querying some component of Ellis’s “memory system” for similar information, such that this component would “return” information from these structures “in response to a query command from the host,” as the claim recites. *See, e.g.,* Pet., 24 (arguing that “A POSITA would be motivated to combine Son’s teachings of a host query for returning information about superblocks [zones] with Ellis’s storage device”).

Such an argument, which only barely falls short of simply asserting that a POSITA would have modified Ellis so that it satisfies the claim, does not identify what Petitioner’s combination is, let alone show any motivation to combine with a reasonable expectation of success. *See In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”).

B. Petitioner Does Not Show That Ellis And Son Teach A Storage Device “operable to receive a management command from a host” (Ground 1, All Challenged Claims; Ground 2, All Challenged Claims Except 21-27 And 29; Ground 3, All Challenged Claims Except Claim 30).

Claim 1, again,⁵ recites a storage device “operable to receive a management command from a host that results in the storage device performing an erasure of one or more erase units.” Petitioner identifies Ellis’s “memory system” (102 in Ellis’s Figure 1) with the claimed “storage device,” Pet. 8, Ellis’s “host system” (108 in Ellis’s Figure 1) with the claimed “host,” *see, e.g.*, Pet., 11-12, and wear leveling commands as the claimed “management commands.” *See id.*

1. Ellis Does Not Teach The Claimed Management Command “from a host.”

In arguing that it “would have been obvious to *modify* Ellis’s storage system” as to teach a storage device “operable to receive a management command from a host that results in the storage device performing an erasure of one or more erase

⁵ As noted above, all of Petitioner’s Ellis combinations rely on Petitioner’s Ground 1 argument involving claim 1. *See* Section II.B.2 *supra*. While independent claims 1 and 10 contain similarly worded versions of this limitation, independent claim 21 and its dependent claims (with the exception of claim 28) do not.

units,” Pet., 12, Petitioner effectively concedes that Ellis does not teach this limitation.

As previously discussed in connection with Petitioner’s failure to show a motivation to combine, *see* Section II.A.1 *supra*, Petitioner expressly acknowledges that in Ellis, a “flash translation layer (FTL)” within Ellis’s “**memory controller 104**”—that is, **not** Ellis’s “host system”—performs operations such as “wear leveling.” Pet., 11. And while Petitioner identifies these operations in Ellis with the claimed “management commands,” the only attempt Petitioner (briefly) makes to allege that these commands are received “**from a host**” is to cite a generic disclosure in Ellis. Pet. 11 (citing Ex. 1004 [Ellis] 4:58-65). To repeat the relevant language, Ellis states that

[t]he functional units in the memory controller 104 can work individually and independently of the other functional units. For illustrative purposes, the memory controller 104 is described by operation of the memory controller 104 with the host system 108 and the memory array 106. It is understood that the memory controller 104, the host system 108, and the memory array 106 can operate any of the modules and functions of the memory controller 104.

Ex. 1004 [Ellis] 4:58-65. Of course, a generic disclosure that any number of components can “operate any of the modules and functions” of Ellis’s controller is not a teaching of any particular division of tasks between Ellis’s host and Ellis’s

controller, let alone a motivation for Petitioner’s proposed modification. *See* Section II.A.1 *supra*. But in using the above paragraph to suggest that Ellis somehow discloses “management command[s] from a host,” *see* Pet., 11, Petitioner obscures how Ellis’s system actually functions—*i.e.*, where the things resembling what Petitioner would identify as “management commands” actually originate in an unmodified Ellis system, and how the metadata relevant to flash maintenance is stored and updated in Ellis’s system.

A POSITA would recognize that in Ellis, commands that initiate garbage collection and wear leveling originate in Ellis’s ***controller***, and do not come from Ellis’s “host system.” Ex. 2020 [Jones-Decl.] ¶ 66. As Dr. Jones explains:

A POSITA would certainly recognize that the flash memory system disclosed in Ellis performs garbage collection and wear leveling operations—as Petitioner’s expert recognizes. *See, e.g.*, Ex. 1003 [Litz-Decl.] ¶ 33 (describing garbage collection in section entitled “Technology Background and State of the Art” as a feature of managing solid-state drives); 35 (in the same section, describing wear leveling as “essential” in flash management)). Ellis expressly discloses, for example, that its controller keeps track of numbers of valid pages, program/erase cycles, and the like—inputs that a POSITA would understand to be important to garbage collection or wear leveling operations (again, as Petitioner’s expert would appear to generally acknowledge in the context of flash management; *see* Ex. 1003 [Litz-Decl.] ¶¶ 33-34 (describing the general role of “valid blocks” in

garbage collection), 35 (discussing P/E cycles in the context of wear leveling).

Although Ellis does not employ the term “garbage collection,” Ellis expressly describes its “memory access module” (which resides within Ellis’s controller, Ex. 1004 [Ellis] 4:67-5:3) as performing “recycle operations” for reasons “including freeing space and wear leveling,” Ex. 1004 [Ellis] 16:4:13), and a POSITA would recognize that garbage collection involves “freeing space.” Because Ellis never discloses any role for its host system in performing such operations (or in accessing or updating any of the relevant inputs), a POSITA would understand that flash garbage collection and wear leveling operations in Ellis are performed exclusively by Ellis’s controller, without involvement of Ellis’s host.

Ex. 2020 [Jones-Decl.] ¶¶ 66-67. Accordingly, and notwithstanding Ellis’s boilerplate disclosure, Ellis does not disclose or teach the claimed management command “from a host”—and the inputs that Ellis uses for what Petitioner identifies with the claimed “management” are stored at Ellis’s controller. This aspect of Ellis is particularly significant for purposes of understanding Petitioner’s proposed modification of Ellis by Son, which also fails to teach the limitation.

2. Petitioner Does Not Show That Son Teaches The Claimed Management Command “from a host” In Its Combination.

Separately from arguing that Ellis renders obvious the claimed management command “from a host,” Petitioner argues that Son teaches this limitation in

combination with Ellis. Petitioner cites various disclosures of Son’s “SSD module” initiating “maintenance operations including garbage collection and wear leveling” for flash memory present in Son’s SSD array (145 in Son’s Figure 1). *See* Pet., 12-13. Implicitly, when Son’s SSD module performs these operations in Son, it does so by sending instructions to the controller that manages Son’s SSD array—and according to Petitioner, the controller’s receipt of such commands from Son’s SSD module would teach the claimed “storage device operable to receive a management command from a host” in some modification of Ellis. *See id.*, 13; *see also id.*, 14 (arguing that Son teaches that “Ellis can be improved by configuring an ‘SDD module’ of a host to perform management operations”), 15 (arguing that Son allegedly motivates “modifying Ellis’s device so that one or more management commands for the superblocks are issued by the host”).

As already discussed in connection with Petitioner’s motivation to combine, however, this argument never identifies exactly how Ellis would be modified—or even indicate, for example, *which* inputs relevant to alleged “management operations” in the individual references would be present in the combined system, and *where* they would be stored in the combined system. *See* Section II.A.2.a) *supra*. Petitioner effectively only states a conclusion—that the combined system would satisfy the limitation. This is not enough. *Apple, Inc. v. Voip-pal.com, Inc.*, IPR2016-01198, Paper 53 (Final Written Decision) at 23 (Nov. 20, 2017)

(“Petitioner’s position ... is not grounded in what a skilled artisan would have gleaned from the teachings of the prior art, and is instead an impermissible exercise of hindsight with the claims ... serving as a guide”).

C. Petitioner Does Not Show That Ellis And Son Teach Logic Operable To Return “information about the zones in response to a query command from the host” (Grounds 1-3, All Challenged Claims).

Claim 1, again,⁶ recites a storage device with logic to “return information about the zones in response to a query command from the host.” Petitioner does not show that this limitation is present in its combinations involving Ellis and Son.

Petitioner begins by analogizing Ellis’s memory controller with the claimed “storage device,” but does not argue that Ellis alone teaches this limitation. *See, e.g.,* Pet., 17 (arguing only “[t]o the extent Ellis does not expressly disclose returning the information recited in this limitation”). For this, Petitioner turns to alleged teachings of Son, analogizing “information about ... superblocks” within Son’s “management tables,” to the claimed “information about the zones,” *see, e.g., id.*, and analogizing

⁶ As noted above, all of Petitioner’s Ellis combinations rely on Petitioner’s Ground 1 (Ellis/Son) argument involving claim 1, *see* Section II.B.2 *supra*, and all challenged claims contain a version of this limitation. Ex. 1001 [’614] 64:60-67 (claim 1), 65:60-66:5 (independent claim 10), 67:15-22 (independent claim 21).

Son's "memory controller" to the claimed logic operable to return that information. *See* Pet., 18-24. As has already been explained, this vague analogy does not adequately motivate Petitioner's combination. *See* Section II.A.2.b) *supra*. As explained further below, Petitioner's also significantly misreads Son's disclosures relating to its SSD module in proposing that Son teaches "return[ing] information . . . in response to a query command from the host" in its combination.

In an attempt to bolster an analogy between Son and the requirements of claim 1, Petitioner repeatedly speaks of Son's SSD module obtaining information from Son's "management tables" as a result of "querying" Son's memory controller, or as the result of Son's memory controller "returning" such information "in response to a query." *See, e.g.,* Pet., 18, 22. Petitioner's repeated use of the term "querying" and its variations derives from claim language, not disclosures in Son, and Petitioner's analogy depends on unstated assumptions that are contrary to express disclosures in Son.

Son does not show any "query command from a host" returning what Petitioner identifies with the claimed "information about the zones" in Son. Petitioner repeatedly relies upon unexplained citation to Son's paragraph 47 in attempting to characterize Son's SSD module as "querying" Son's memory controller for information. *See, e.g.,* Pet., 18 (arguing that Son's SSD module retrieves information from a data structure in step 410 of Son's Figure 4 "by

‘querying the controller’” in Son) (purportedly quoting Ex. 1005 [Son] [0047]), 22 (partially quoting “querying” language in Son’s paragraph 47 again), 22 (citing Son’s paragraph 47 as support for an argument that “Son discloses” its memory controller as “return[ing]” certain information “in response to a query from the SSD module in the host”), 24 (same). But Son’s paragraph 47 does not disclose that Son’s SSD module obtains the information that Petitioner is pointing to by “querying” its memory controller. Son’s paragraph 47, which discusses Son’s SSD module selecting a “superblock” during a write operation (at step 415 of Son’s Figure 4), reads in full:

At block 415, the SSD module selects a superblock and PLBA to store the source data in order to fulfill the requested operation. When selecting the superblock, the SSD module may evaluate any combination of several factors including, as non-exhaustive examples: the volume of source data to be written, the number of free, valid and/or invalid pages in each candidate superblock, the state of each candidate superblock, the data type of the source data, the data type of data already stored in each candidate superblock, the wear count and/or last modification time of each candidate superblock, the superblock associated with a previously-stored PLBA, the physical configuration or layout of the solid-state memory in the SSD. Other examples of factors include other recent, in-progress, or pending I/O requests received before, during, or after the current write request was received. In the case of a rewrite, the SSD module may also evaluate the

previously-stored PLBA. The SSD module may determine some or all of these various factors *by accessing and analyzing the various management data structures described previously* and/or *by querying the file system or the controller*.

Ex. 1005 [Son] [0047]. Here, the “various management data structures described previously” are the tables described in Son’s Figures 3A-3D, as Son describes (for example) in its paragraphs 37-43. And Petitioner identifies material within these “management data structures” as providing alleged examples of the claimed “information.” *See, e.g.,* Pet., 17 (citing Figures 3A-3D), 18 (reproducing Son’s Figure 3C), 18-19 (arguing that “retriev[ing] information from the management data structures” as in Son’s Figure 4 teaches this limitation). As Dr. Jones explains,

A POSITA would understand that when Son speaks of “accessing and analyzing the various management data structures described previously *and/or* by querying the file system or the controller,” it is describing *alternative* ways in which Son’s SSD module may obtain information. In particular, a POSITA would understand that when Son speaks of its SSD module accessing and analyzing the contents of tables in its Figures 3A-3D, this occurs *without* querying Son’s controller for that information.

Indeed, a POSITA would recognize that the ability of Son’s SSD module to access and analyze information without querying its controller is an essential feature of Son’s system. Son repeatedly says, for example, that one object of its design is to “mov[e] solid-state

management functions above the firmware level.” *See, e.g.*, Ex. 1005 [Son], Abstract, [0014]. Son also says, for example, that its system “may achieve various advantages over conventional systems” by allowing the use of “simplified SSD firmware.” *See id.* A POSITA would understand this reference to “simplified” firmware to include firmware that might support basic read and write functionality but not any specialized “querying” commands for flash status information.

Ex. 2020 [Jones-Decl.] ¶¶ 79-80.

Separately from its argument from Son’s paragraph 47, Petitioner makes an analogy between the garbage collection process described in Son’s Figure 6 and the claimed “query command from a host.” Pet., 19. Partially quoting from Son’s paragraph 65, Petitioner states that

Similarly, Figure 6 shows a garbage collection process that involves “identifying source superblocks and target superblocks” at step 603. [Ex. 1005] [Son], Fig. 6. Son discloses that in the identify step the SSD module evaluates “information obtained from the controller,” such as “information shown in the superblock table, other management data structures, or information obtained from the controller.” [Ex. 1005] [Son], [0065].

Pet., 19. As Dr. Jones explains:

I do not agree with Petitioner’s implied argument—that the “information shown in the superblock table” or the “other management data structures” referenced in Son’s paragraph 65 are examples of what

Son refers to when it refers to “information obtained from the controller.” A POSITA would instead understand looking to “information shown in the superblock table” and “other management data structures” the garbage collection process of Figure 6 to be *alternatives* to what Son refers to as consulting “information obtained from the controller.”

Ex. 2020 [Jones-Decl.] ¶ 82. Son’s paragraph 65 states, in relevant part:

At block 603, the SSD module identifies a source superblock and target superblocks. The SSD module may consider various combinations of factors to select the source and target superblocks, including the following such factors, as nonexhaustive examples: free page count, valid page count, invalid page count, last modification time, wear count, the data type of data stored in a superblock (e.g., metadata versus user versus raw), or any other information about superblocks (or their constituent pages, including page status or DIF header information), such as the other information shown in the superblock table, other management data structures, *or* information obtained from the controller or file system.

Ex. 1005 [Son] [0065]. As Dr. Jones explains:

A POSITA would recognize this disclosure for what it is: generic, boilerplate language signaling that nothing in Son’s design strictly limits its SSD module to considering data from only one source when deciding what superblocks are to be the source or target of garbage collection. Paragraph 65 does not identify any of the information that it enumerates (or any examples that Petitioner

identifies in connection with this limitation) as being obtained by querying Son’s controller. Indeed, here Son suggests that the SSD module’s use of the “superblock table” or “other management data structures” are *alternatives* to using “information obtained from the controller or file system.”

Elsewhere, Son makes clear that the SSD module itself maintains its “management data structures” (e.g., its page status table, mapping table, superblock table, page detail array), and a POSITA would understand that Son’s SSD module may access the information that is kept in these tables without “querying” Son’s memory controller for that purpose. *See, e.g.*, Ex. 1005 [Son], Figs. 3A-3D; [0033]-[0043].

Ex. 2020 [Jones-Decl.] ¶ 84-85.

By reading Son *as if* it discloses what Petitioner identifies as the claimed “information” being returned “in response to a query command” from what Petitioner identifies as a “host,” Petitioner is thus using claim language to guide a modification of Son. This is an impermissible use of hindsight. Accordingly, Petitioner does not show how its combination teaches a storage device “return[ing] information about the zones in response to a query command from the host.”

III. PETITIONER’S TAKEO GROUNDS FAIL.

All of Petitioner’s Takeo grounds rely upon Petitioner’s combination of Takeo, Son, and Sinclair (“Ground 4”) against independent claim 1. *See, e.g.*, 72-81 (Petitioner’s Ground 5 referring to Petitioner’s Ground 4 analysis of claim 1 for all

aspects of independent claims), Pet. 82-85 (Petitioner’s Ground 6 discussing dependent claims only), 68-72 (all of Petitioner’s Ground 4 arguments incorporating Petitioner’s Ground 4 arguments for claim 1). As explained below, Petitioner does not motivate its proposed modification of Takeo, and fails to show that the claimed “management command from a host” and writes “using a common page address” are present in its combinations.

A. Petitioner Does Not Show A Motivation To Combine With A Reasonable Expectation Of Success.

As already mentioned in the context of Petitioner’s Ellis grounds, a failure to adequately specify a combination is a failure to show a motivation to combine with a reasonable expectation of success. *See* Section II.A *supra*. Similarly, merely pointing to disparate disclosures in the prior art does not establish a motivation to combine. *Virtek Vision Int’l ULC v. Assembly Guidance Sys.*, 97 F.4th 882, 886 (Fed. Cir. 2024) (reversing Board finding of obviousness when prior art disclosure of “alternative arrangements” known in the art did “not provide any reason why a skilled artisan would use” one and not another).

1. Petitioner’s Takeo Combinations Are Not Specified (Grounds 4-6, All Challenged Claims).

Like Petitioner’s combinations involving Ellis, Petitioner’s combinations involving Takeo are described only indirectly—in terms how they allegedly satisfy individual claim limitations. *See, e.g.*, Pet., 58, 60-61, 64. Petitioner never

articulates a coherent description of what the combined system actually is or how it would function.

- a) *Petitioner Does Not Explain How Its Takeo Combinations Receive A “management command from a host.”*⁷

Petitioner reveals aspects of Petitioner’s Takeo combinations when discussing a limitation of claim 1 that recites a storage device “operable to receive a management command from a host that results in the storage device performing an erasure of one or more erase units of the set of erase units.” Ex. 1001 [’614] 64:52-55 (claim 1). All of Petitioner’s obviousness arguments involving Takeo incorporate its Ground 4 argument as to claim 1,⁸ *see, e.g.*, Pet., 73-75, 78-80, where Petitioner admits that “Takeo does not expressly discuss management commands.” Pet., 58.

⁷ This argument addresses: Ground 4, all challenged claims; Ground 5, all challenged claims except claims 21-27 and 29; and, Ground 6, all challenged claims except claim 30.

⁸ While independent claims 1 and 10 contain similarly worded versions of this limitation, independent claim 21 and its dependent claims (with the exception of claim 28) do not. Petitioner does not, however, separately articulate any combinations as to the claims that do not contain this limitation.

Petitioner first attempts to overcome Takeo’s failure to disclose any “management commands” by merely stating that “Son discloses the claimed management command”—an assertion that Petitioner supports only with reference to its earlier arguments involving a combination of *Ellis* and Son. *See, e.g., Pet.*, 58. As already shown, however, Petitioner does not explain how its combination of *Ellis* and Son actually issues the claimed “management commands.” *See* Sections II.A.2, II.B.2 *supra*.

Even assuming *arguendo* that Petitioner had shown how the claimed “management commands” are taught in a combination of *Ellis* and Son, this would not identify (let alone motivate) how any alleged “management commands” are implemented in a combination involving *Takeo* and Son. Offering identical rationales to combine disparate references is evidence of hindsight, not a showing of a motivation to combine. *See, e.g., UiPath, Inc. v. Rule 14 LLC*, IPR2024-01304, Paper 8, 33 (Mar. 14, 2025) (denying institution) (“Petitioner’s reasons to combine Liang and Lucovsky or Mellin are identical to its previously stated reasons to combine Kirsch and Lucovsky or Mellin. While not necessarily improper, repeating the identical argument emphasizes Petitioner’s conclusory reasoning that one skilled in the art would simply combine the references for the purposes of adding the feature, entirely independent of the primary references’ teachings.”).

Petitioner alternatively argues that “[a] POSITA would understand that a flash memory system should perform management operations such as wear leveling and garbage collection” and that “[b]ecause Takeo is silent regarding these management operations, a POSITA would look to references like Son that describe these operations.” Pet., 58. This argument fails for several reasons.

First, as explained below in connection with Petitioner’s failure to show that this limitation is satisfied, the fact that Takeo does not use the express terms “wear leveling” or “garbage collection” does not mean that Takeo is agnostic as to which component of its system performs these operations: that component is Takeo’s memory controller—not Takeo’s host. *See* Section III.B *infra*. Accordingly, to “look to references like Son that describe these operations” as Petitioner proposes, Pet., 58, is not to look to a reference for details of how these operations might be implemented in an embodiment of Takeo, but to ***change*** how Takeo allocates the responsibility for flash management. In failing to acknowledge this, Petitioner does not explain why a POSITA would be motivated to modify Takeo’s controller to expect to receive wear leveling or garbage collection instructions “from a host.”

Second, in arguing that a POSITA “would look to references like Son that describe these operations,” Pet., 58, Petitioner ignores the fact that many references “describe the[] operations” of wear leveling and garbage collection in ways that do not locate this functionality at a host, let alone Son’s host. As one example,

Petitioner’s Ellis grounds cite a combination of Ellis and Son for alleged teachings of this limitation, and Ellis locates this functionality at the controller. *See, e.g.*, Pet., 11 (arguing that Ellis discloses a “memory controller” that “may issue wear leveling and garbage collection commands”). *See, e.g.*, Section II.A.1 *supra*. As another example, Petitioner’s Takeo combinations all involve Sinclair in combination with Son and Takeo, and Sinclair not only teaches that it is common to locate similar functionality in a controller, *see, e.g.*, Ex. 1006 [Sinclair] [0083] (referring to “[d]ata consolidation and garbage collection” as “operations [that] are normally scheduled by the memory system controller to take place in the background as much as possible,” and noting that “the need to perform these operations can cause the controller to have to give the host a busy status signal until such an operation is completed”), but employs a controller that may “automatically” perform similar operations. *See, e.g., id.* [00155] (addressing a situation where “the memory controller manages the physical memory in way that automatically consolidates data from partially written blocks in order to reclaim new erased block capacity”). Ex. 2020 [Jones-Decl.] ¶ 95.

So even if Takeo’s design were assumed *arguendo* to be entirely agnostic on where garbage collection and wear leveling might be managed in its system, Petitioner does not explain why this silence would motivate a POSITA to look to a reference like Son—and not, for example, to a reference like Ellis, or Sinclair, or any

other reference that “describe[s] these operations.” Treating the alleged “silen[ce]” of one reference as “motivation” to look in a particular place for a particular kind of modification betrays the role of hindsight in Petitioner’s analysis. *See Metalcraft of Mayville, Inc. v. Toro Co.*, 848 F.3d 1358, 1367 (Fed. Cir. 2017) (“[W]e cannot allow hindsight bias to be the thread that stitches together prior art patches into something that is the claimed invention.”). Of course, the mere silence of one reference on a topic that is mentioned in another reference is not itself a motivation to combine. *See, e.g., ActiveVideo Networks, Inc. v. Verizon Commc ’ns, Inc.*, 694 F.3d 1312, 1327 (Fed. Cir. 2012) (testimony that “modular” nature of components rendered obvious “how to combine any of a number of references to achieve the claimed invention” was “not sufficient” and “fraught with hindsight bias”).

Third and finally, having argued that Takeo’s alleged silence points to Son for details of host-initiated wear leveling and garbage collection, Petitioner does not complete its argument by making any connection between what Son allegedly teaches and its proposed modification of Takeo. The following is the entirety of what Petitioner has to say about this issue:

And as discussed above (see Ground 1, Limitation 1[a.ii]), the prior art includes express teachings, suggestions and motivations for combining Son’s disclosure of host-issued management operations to Takeo’s storage device, including that it would allow the system to

incorporate device conditions in its management logic, improve execution speed, and avoid the controller becoming overloaded.

Pet., 59. Here Petitioner’s “Ground 1” is ***not*** a Takeo ground, but one of Petitioner’s Ellis grounds, and it does not address “combining Son’s disclosure of host-issued management operations to Takeo’s storage device.” And this vague narrative about the “teachings, suggestions, and motivations for combining Son’s disclosure ... to Takeo’s storage device” does not identify what the “host” in the Takeo/Son combination would be—except to imply that it would satisfy a claim limitation that Petitioner admits Takeo does not disclose.

This vague statement is insufficient to specify a combination, let alone to provide a motivation to combine. *See, e.g. Markforged Inc., v. Continuous Composites, Inc.*, IPR2022-01220, Paper 8, 36 (Jan. 11, 2023) (denying institution) (“Petitioner appears to be arguing that a POSITA could have combined unspecified teachings of Lipsker in unspecified ways to improve Wohrl’s method.... Accordingly, the Petition lacks an adequate explanation as to why a skilled artisan reading Wohrl would have looked specifically to Lipsker for its particular teachings, and how a skilled artisan would have combined the teachings of Wohrl and Lipsker in the manner claimed.”).

b) *Petitioner Does Not Explain Why A POSITA Would Modify Takeo To Return Its Alleged “information about [] zones” (Grounds 4-6, All Challenged Claims).*

Every challenged claim recites logic operable to return “information about [] zones in response to a query command from [a] host,” where the claimed “information” comprises, among other things, a “zone size,” and “logical addresses associated with respective zones” (or a “logical address associated with a zone of the zones”). *See, e.g.*, Ex. 1001 [’614] 64:60-67 (claim 1), 65:59–66:5 (independent claim 10), 67:15-22 (independent claim 21). All of Petitioner’s arguments regarding these limitations and Takeo refer back to the arguments made in Petitioner’s Ground 4 against this limitation as it appears in independent claim 1. *See, e.g.*, Pet., 74, 79 (incorporating Petitioner’s arguments for “limitation 1[b.i]”).

Although Petitioner identifies what it regards as a “zone size” and “logical addresses associated with respective zones” in Takeo, Petitioner admits that Takeo does not disclose returning a “zone size” or “logical addresses associated with respective zones” in response to a query command from a host. *See, e.g.*, Pet., 65-66. Petitioner offers two arguments that the POSITA would be motivated to modify Takeo as to satisfy this limitation. As explained below, these arguments do not even specify Petitioner’s proposed modifications, let alone motivate them.

(1) *Petitioner’s Ellis-Based Argument Is Unsupported.*

First, Petitioner argues that “Son renders obvious returning the claimed

information about the zones in response to a query command from the host for the reasons discussed in connection with Ground 1, Limitation 1[b.i].” Pet., 62. Here Petitioner’s “Ground 1” refers to Petitioner’s combination of *Ellis* and Son—not to any combination involving Takeo. As Petitioner’s Takeo grounds expressly and exclusively rely upon Takeo alone for some limitations of claim 1 (*see, e.g.*, Pet., 55-57 (Petitioner’s limitations [1pre], [1a], [1a.i]), 62 (limitations [1a.iv] and [1b]), 68 (limitations [1b.iii] and [1c])), and expressly rely upon Takeo and Sinclair for another limitation (*see, e.g.*, Pet. 66-67 (limitation [1b.ii])), it is not clear how Petitioner proposes to transfer its arguments concerning *Ellis* and Son (some of which include argument specific to *Ellis*’s “storage device,” Pet., 17, 24-25) to a combination of Takeo, Son, and Sinclair, such that the limitation is shown. Simply pointing to various disclosures in various references is not enough to identify a combination, let alone provide a motivation to combine. *See, e.g., KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (an assertion of obviousness “cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness”) (quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)).

(2) *Petitioner’s Sinclair-Based Argument Rests On A Faulty Analogy.*

Petitioner’s second argument ignores Son (and any modifications made in

view of Son) entirely, and focuses on pairing various disclosures in Takeo with disclosures in Sinclair. *See generally* Pet., 62-66. Petitioner contends that “Sinclair teaches that if the host ‘know[s] the physical characteristics of the memory with which it is operating’ ... it can avoid having files scattered across metablocks.” Pet., 66. But Sinclair’s “metablocks” are not Takeo’s “zones,” so the problem of having “files scattered across metablocks” does not exist in Takeo’s system. And Petitioner’s rationale for combining the systems rests upon an analogy between Sinclair’s metablocks and Takeo’s zones that does not make technological sense.

In relevant part, claim 1 recites “[a] storage device comprising . . . logic operable to[] return information about the zones in response to a query command from the host, the information about the zones comprising . . . a zone state, . . . [and] logical addresses associated with respective zones” Ex. 1001 [’614] 64:46, 64:60-65. Petitioner identifies Takeo’s flash memory system (labeled 1 in Takeo’s Figure 1) as the claimed “storage device,” Pet. 55, Takeo’s host system (labeled 4 in Takeo’s Figure 1) as the claimed “host,” Pet., 55-56, and Takeo’s physical zones with the claimed “zones,” Pet., 57 (citing Ex. 1012 [Takeo] ¶ 41). Petitioner states that the number of bytes in a block of one of Takeo’s physical zones determines the claimed “zone size.” Pet., 58.

With these identifications, Petitioner admits that Takeo does not disclose returning the claimed “zone size” or “logical addresses associated with respective

zones” in response to a query from what Petitioner identifies as Takeo’s “host.” *See* Pet., 65. Petitioner argues instead that Sinclair teaches this limitation and does so by analogy.

Petitioner states that Sinclair’s memory system exposes “the ‘number of pages per metablock’ and the ‘size of a physical page,’” as well as “the lowest logical address that is mapped to a first page of a physical metablock,” to Sinclair’s host. *See* Pet., 65-66. Implicitly analogizing Sinclair’s “metablocks” to Takeo’s “zones,” Petitioner argues that “[a] POSITA would have been motivated to modify Takeo to return this information as part of Takeo’s vendor internal information acquisition command,” Pet., 66, purportedly because:

Sinclair discloses that when writing data files they may become “scattered among a larger number of metablocks when file boundaries are not known.” Ex. 1006 [Sinclair], ¶ 86. Sinclair teaches that if the host “know[s] the physical characteristics of the memory with which it is operating” (e.g., the zone size and logical addressing information discussed above) it can avoid having files scattered across metablocks.

Pet., 66.

But a POSITA would not analogize Sinclair’s metablocks to Takeo’s “zones” in this way, let alone as inspiration for Petitioner’s proposed modification of Takeo. As Dr. Jones explains:

Sinclair’s “metablocks” are not the same as Takeo’s “zones.” Both Sinclair’s “metablocks” and Takeo’s “zones” are groups of individually eraseable units of flash memory (Sinclair calls its units “blocks,” *see, e.g.*, Ex. 1006 [Sinclair], [0062], while Takeo calls these units “erasure units,” “block units,” or “blocks,” *see, e.g.*, Ex. 1012 [Takeo] ¶ 3). But these groups are managed differently in the two systems.

Sinclair discloses that the units of flash within its “metablocks” are written and erased together. Ex. 1006 [Sinclair] [0062]; *see also id.* (noting that while a “block” represents “the smallest number of memory cells that are physically erasable together[,] [f]or increased parallelism . . . the blocks are operated in larger metablock units”). Petitioner’s reading of Sinclair expressly relies on aspect of Sinclair’s “metablocks.” Pet., 53 (stating that “‘metablocks’ . . . comprise a group of blocks across multiple planes that may be operated together (e.g., *written and erased together*)”).

A POSITA would understand that the units of flash that make up Takeo’s “zones,” by contrast, are not written and erased together; in particular, the fundamental unit of erasing flash memory in Takeo is a block. In the main example discussed in Takeo, there are hundreds of blocks in each of Takeo’s “zones.”

Ex. 2020 [Jones-Decl.] ¶ 109-111.

Confirming Dr. Jones’s opinion, Takeo nowhere discusses, for example, any necessity or advantage of writing to all blocks within a Takeo “zone” at once, or the any necessity or advantage of erasing all blocks within a Takeo “zone” at once, or

otherwise refer to writing to or erasing flash memory in Takeo as an operation that only occurs at the level of Takeo “zones.” By contrast, Takeo expressly discloses storing information relating to block-level write operations *within individual blocks*, on the basis of *individual blocks*, clearly indicating to the POSITA that Takeo’s write operations are not limited to writing entire Takeo “zones” at a time. Ex. 1012 [Takeo] ¶ 43 (“logical address information corresponding to the data that has been written will be written to the redundant region of *the physical block where the data was written*”); Ex. 2020 [Jones-Decl.] ¶ 112.

As Dr. Jones further explains:

A simple but plain indication to the POSITA that Takeo’s “zones” are not written to or erased as groups is the fact that Takeo’s central example is a system with exactly eight “zones.” *See, e.g.*, Ex. 1012 [Takeo], Figure 4 (listing counts for “PZN #0” through “PZN #7”). A POSITA would not understand *any* system with only eight collectively erasable divisions of flash memory as a practical system, let alone a system appropriate for illustrating aspects of a technical disclosure like Takeo’s patent publication.

Managing flash on the basis of only eight units that are written and erased together is not a plausible operating model: every time such a device needed free space, it would be forced to evacuate whatever valid data still resides in a targeted zone and then erase 12.5% of the entire medium—just to continue an ordinary write that might involve far less data than 12.5% of the entire medium. The practical effect

would be severe and immediate: latency would spike during each reclamation cycle, and such a device would frequently end up moving and rewriting far more data than the host actually needed to write. The cumulative effect of this method of management would accelerate wear on the flash, and any practical system would not be designed so that small host writes do not trigger enormous volumes of internal writes. Of course, Takeo is not designed that way, because Takeo’s “zones” are not written and erased together, like Sinclair’s “metablocks.” Takeo’s “zones” are more roughly analogous to higher-level administrative partitions—functionally analogous to separate volumes or drives—than a fundamental unit of writing to or reclaiming flash memory.

Ex. 2020 [Jones-Decl.] ¶¶ 113-114.

Because Takeo’s “zones” are different from Sinclair’s “metablocks,” the analogy that Petitioner uses in an attempt to motivate sending what Petitioner identifies as Takeo’s “zone size” and “logical addresses associated with respective zones” to Takeo’s host breaks down. As Dr. Jones further explains:

The fact that the blocks within Sinclair’s “metablocks” are all written and erased together underlies what Petitioner offers as an alleged motivation for modifying Takeo—Sinclair’s concern for “scattering” a file across multiple metablocks, which in Sinclair is addressed by host-visible metablock size information. *See* Pet. 66 (suggesting that a host with a “query command from the host” that

returns a “zone size” can “avoid having files scattered across metablocks”).

In more detail, if a single file is spread across many of Sinclair’s metablocks, it may occur that these metablocks each hold a small amount of valid data (in the form of their respective portions of that single file) and a lot of reclaimable space. Compacting this space at the metablock level (as may happen during garbage collection) requires copying the valid portions out of each metablock and erasing each entire metablock—work that scales with the number of affected metablocks. *See, e.g.*, Ex. 1006 [Sinclair], [0081]-[0082]. Again, this is because Sinclair teaches that the units of flash within a “metablock” are erased together. As described by Petitioner, Sinclair’s system does not regain storage capacity by erasing smaller groups of individually erasable units of flash memory (*e.g.*, individual blocks within Sinclair’s “metablocks”).

Ex. 2020 [Jones-Decl.] ¶¶ 116-117.

Thus, when Sinclair describes its host having access to what Petitioner analogizes to the claimed “zone size,” so that Sinclair’s host can attempt to place files across smaller numbers of Sinclair’s “metablocks,” Sinclair is attempting to mitigate what would otherwise be the increased costs of performing data compaction and garbage collection operations on the basis of its metablocks. *See, e.g.*, Ex. 1006 [Sinclair], [0006], [0067], [0078]-[0080]; Ex. 2020 [Jones-Decl.] ¶ 118. As Dr. Jones explains, however, this problem does not exist in Takeo:

A POSITA would understand that compaction and garbage collection in Takeo do not implicate writing to or erasing entire Takeo “zones” at once. A POSITA would instead understand that when Takeo’s memory controller performs data compaction and garbage collection, it can write to and erase individual blocks within Takeo “zones.” What Petitioner identifies as a “zone size” in Takeo is thus not relevant to flash maintenance in Takeo in way that Sinclair’s metablock size is relevant to flash maintenance in Sinclair.

The quantitative difference between a “metablock” in Sinclair and a Takeo “zone” is well illustrated by an example in Takeo in which there are 1024 physical blocks (used for storing 1000 logical blocks)) in every “zone.” *See, e.g.*, Ex. 1012 [Takeo] ¶ 42. For purposes of comparison with Petitioner’s analogy to Sinclair, Sinclair’s Figure 4 offers an example in which “four blocks ... are shown to form one metablock” in Sinclair. Ex. 1006 [Sinclair] [0062]; Pet. 53 (citing this example). If one were to use Petitioner’s analogy to “metablocks” to re-“zone” Takeo’s system of 8000 logical blocks to a system whose Takeo “zones” have only four blocks each (like Sinclair’s exemplary “metablocks”), the resulting system would employ thousands of “zones,” not eight of them. Even then, considerations applicable to “metablocks” in Sinclair would not directly apply to Takeo “zones,” for example, because in this hypothetical embodiment of Takeo system, the four blocks within a Takeo “zones” in this hypothetical would not necessarily be written and erased as a group, as “metablocks” are in Sinclair.

Ex. 2020 [Jones-Decl.] ¶¶ 119-120.

As Dr. Jones further explains, Petitioner’s analogy between “zones” and “metablocks” also selectively reads Sinclair, to the potential exclusion of other conceivable analogies with Sinclair that would not support its claim mapping:

Other features of Sinclair also undercut the analogy that Petitioner draws between “zones” and “metablocks” in Sinclair.

As one example, Sinclair expressly contemplates systems that make use of metablocks of variable size—that is, metablocks that can have different numbers of blocks to match different kinds of workloads. *See* Ex. 1006 [Sinclair] ¶ 84. A modification of Takeo consistent with this aspect of Sinclair would not necessarily adopt the single, fixed “metablock size” that Petitioner relies on in supposedly motivating Takeo’s host returning a “zone size.”

As another example, Sinclair discloses an alternative interface that “eliminate[s] use of the logical address space,” i.e., a file-object interface that does not traffic in host-visible logical block addresses at all. *Id.* ¶ 87. A modification of Takeo aligned with this aspect of Sinclair might not involve returning what “logical addresses associated with respective zones” at all.

Ex. 2020 [Jones-Decl.] ¶¶ 121-123.

Where references contain distinct and conflicting teachings about the same things, a Petitioner must do more than simply choose among them to provide a motivation to combine. *See, e.g., Samsung Elecs. Co., Ltd. v. Dynamics Inc.*, IPR2020-00499, Paper 63, 32-34 (Aug. 10, 2021) (Final Written Decision)

(“Petitioner proffers arguments that a person of ordinary skill in the art could have combined the teachings of [the two references], but fails to explain sufficiently why they would have chosen the specific features relied upon by Petitioner to read on the claim.”). Accordingly, Petitioner has failed to motivate its combinations of Takeo, Son, and Sinclair.

B. Petitioner Does Not Show That Its Takeo Combinations Teach A “storage device . . . operable to receive a management command from a host” (Ground 4, All Challenged Claims; Ground 5, All Challenged Claims Except 21-27 and 29; Ground 6, All Challenged Claims Except Claim 30).

Claim 1⁹ recites a storage device . . . operable to receive a management command from a host that results in the storage device performing an erasure of one or more erase units of the set of erase units.” This limitation is not shown in Petitioner’s Takeo combinations.

Petitioner expressly admits that “Takeo does not expressly discuss management commands” and relies on Son for this limitation. Pet., 58. According

⁹ As noted above, all of Petitioner’s Takeo combinations rely on Petitioner’s Ground 4 Takeo/Son/Sinclair argument involving claim 1. *See* Section III *supra*. While independent claims 1 and 10 contain similarly worded versions of this limitation, independent claim 21 and its dependent claims (with the exception of claim 28) do not.

to Petitioner, upon “[a]pplying Son’s teaching to Takeo, the flash memory system of Takeo would receive a management command from the host system that results in the erasure of erase units in a zone.” Pet., 58. (Petitioner does not rely on Sinclair at all for this limitation. *See id.*, 58-59.)

Although Petitioner does not expressly identify any specific example of a “management command” in its combination of Takeo and Son and Sinclair, Petitioner’s statements that “[a] POSITA would understand that a flash memory system should perform management operations such as wear leveling and garbage collection,” and that “Son . . . describe[s] these operations,” Pet., 58, are implicitly an argument that the “management command” in Petitioner’s proposed combination would relate to wear leveling or garbage collection in a modified version of Takeo.

As noted above in connection with Petitioner’s alleged motivation to combine, however, Petitioner fails to fully articulate what its combination of Takeo, Son, and Sinclair even is, let alone how wear leveling and garbage collection is performed in the combination. *See* Section III.A.1.a) *supra*. Petitioner therefore fails to show that this limitation is taught in its Takeo grounds for at least this reason.

Petitioner also argues that “a POSITA would look to references like Son that describe” wear leveling and garbage collection because “Takeo is silent regarding these management operations.” Pet., 58. This statement ignores the fact that a POSITA would understand that wear leveling and garbage collection are performed

by the controller (not the host) in Takeo's system. As Dr. Jones explains:

Although Takeo does not use the terms “garbage collection” or “wear leveling,” a POSITA would readily understand that Takeo's system necessarily performs garbage collection and wear leveling. As of Takeo's 2007 filing date, these background maintenance operations were well-recognized as essential to the operation of NAND flash memory, which has limited program/erase cycles and cannot be overwritten in place—as Petitioner elsewhere appears to recognize when it says that “A POSITA would understand that a flash memory system should perform management operations such as wear leveling and garbage collection.” Pet., 58.

A POSITA would also understand that in Takeo's system, it is Takeo's controller that is tasked with wear leveling and garbage collection. Takeo's controller is designed to interoperate with hosts that may not even be aware that they are using flash storage rather than magnetic storage. *See, e.g.*, Ex. 1012 [Takeo] ¶ 4 (noting that “memory systems using NAND flash memory are increasingly used in applications that replace the conventional magnetic disk devices” and “[f]or this reason, the interface of memory systems using NAND flash memory typically uses AT Attachment (ATA), which is used in magnetic disk devices”), Abstract (describing Takeo as providing a way “[t]o efficiently transfer internal information within a flash memory system that is connected to an ATA interface to the host system side”), [0005] (describing how host access works “even in the case of a flash memory system in which a peripheral device uses flash memory as a storage medium”), [0007] (describing the host's use of an “ATA

interface”). A POSITA would understand that such a controller could not rely on the host to direct any flash-specific housekeeping such as garbage collection and wear leveling, and that such a controller would not rely on host input for that purpose.

Ex. 2020 [Jones-Decl.] ¶¶ 129-130. As Dr. Jones further explains, another indication that garbage collection and wear leveling are performed by Takeo’s controller is the absence of any disclosure in Takeo of its host sending anything other than standard input/output commands (that is, commands to read and write data), and “vendor-specific” commands that are exclusively described in terms of reporting status information back to Takeo’s host:

Takeo does not disclose its host sending any commands to its controller to initiate garbage collection or wear leveling. Takeo presents its controller as receiving commands from its host via an interface that includes support for host reads, host writes, and a vendor-specific query path—the “internal-information acquisition command”—which Takeo uses for the host to request, and the controller to report, state information. *See generally* Ex. 1012 [Takeo], Fig. 1 (depicting an ATA host/device architecture), Fig. 4 (depicting per-zone state information as can be reported back to Takeo’s host), Figs. 5-6 (flowcharts for computing/buffering and returning information via the ATA data path).

Ex. 2020 [Jones-Decl.] ¶ 132.

Accordingly, when Petitioner analogizes Takeo to a reference that “fail[s] to

disclose the details” and Son to a secondary reference that “discloses the details,” *cf.* Pet., 59, and when Petitioner characterizes its combination as “combining Son’s disclosure of ***host-issued management operations*** to Takeo’s storage device,” Pet., 59, Petitioner is silently modifying where Takeo’s system performs wear leveling and garbage collection. (By not even acknowledging that this proposal ***does*** modify the operation of Takeo’s controller, of course, Petitioner also fails to explain how the limitation is present in the combined system. *See* Section III.A.1.a) *supra*.

Furthermore (and as already noted in connection with the motivation to combine, *see* Section III.A.1.a) *supra*), even if Takeo were assumed to be agnostic about where its garbage collection and wear leveling operations are managed, and even if a POSITA would be motivated to look to Son for details of implementing these operations in a system like Takeo, Petitioner never actually explains how Son in combination with Takeo teaches the limitation. Pet., 58-60. Petitioner simply refers back to its arguments involving Ellis and Son and implies that the limitation would be satisfied. *Id.* This is not enough. *In re Magnum Oil Tools*, 829 F.3d at 1380 (petitioner must “articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness”); *see also, e.g., Vectair Sys., Inc. v. Fresh Prods., Inc.*, IPR2024-00824, Paper 9, 19 (Nov. 12, 2024) (denying institution when challenge included “attorney argument with no cited evidentiary support,” and “Petitioner’s primary argument regarding key elements of claim 1 lack[ed] adequate

evidentiary support”).

C. Petitioner Does Not Show That Its Takeo Combinations Teach Logic Operable To Write To Erase Units “using a common page address” (Grounds 5-6, All Challenged Claims).

Dependent claim 3 and independent claims 10 and 21 (from which all other claims challenged in Grounds 5-6 depend) each recite logic to write “two or more erase units of the set of erase units of the specified zone using a common page address” (claim 3), or to write a “set of erase units of a specified zone using a common page address” (claim 10), or to write a “set of erase units of the zone of the zones using a common page address” (claim 21). *See, e.g.*, Ex. 1001 [’614] 65:17-20, 66:6-7, 67:17-19. Petitioner addresses all limitations identically, with reference to its Ground 5 argument for dependent claim 3. *See, e.g.*, Pet., 72, 75,¹⁰ 80. Petitioner does not argue that this limitation is disclosed or taught by Takeo. *See id.*

1. Sinclair Does Not Teach This Limitation In Petitioner’s Combinations.

In arguing that its combinations teach the claimed write “using a common page address,” Petitioner first argues as follows:

¹⁰ This reference (“See Ground 4, Claim 3”) contains a typographical error.

As Ground 4 has no such challenge, Petitioner likely means to refer to Ground 5, Claim 3.

Sinclair renders obvious the logic is operable to write the data to the two or more erase units of the set of erase units of the specified zone. See Ground 4, Limitation 1[b.ii].

Pet. 72. (Petitioner’s reference to “the logic” refers back to Takeo’s memory controller, which Petitioner identified in claim 1 as the “logic” in claim 1 that claim 3’s “logic ... operable to write” refers back to. See Pet., 62.)

Following Petitioner’s citation “Ground 4, Limitation 1[b.ii]” to its source, one finds the following:

Sinclair discloses that erase blocks that span multiple planes may be grouped together into a metablock [zone] and that data may be written to the set of erase blocks [sets of erase units] of a specified metablock [specified zone] using a multi-plane write command such that the data is written concurrently to two or more erase blocks of the set of erase blocks. See Ground 1, Limitation 1[b.ii].

Pet., 66-67. Following that citation to its source, one finds mostly arguments about Ellis—and Petitioner’s Ground 1 is an Ellis ground, not a Takeo ground. Pet., 25-26. But one also finds the following:

To the extent Ellis does not explicitly disclose the use of a multi-plane command, it was at least obvious in view of the general knowledge of a POSITA. EX1003, ¶109. Multi-plane commands were well-known in the art. EX1014, §1.3 (“SSD manufacturers have provided advanced commands, such as... multi-plane...”), §2.2 (“Multi-plane command activates multiple read, program or erase

operations in all planes of the same die”), Fig. 4. The prior art includes express teachings, suggestions and motivations *to utilize a multi-plane command with Ellis’s storage device* by disclosing that the commands permit “parallel” operations and that “parallelism has been regarded as a key to achieving the peak performance.” EX1004, 5:32-52; EX1010, 2:41-43; EX1014, §1.4.

Pet., 26.

Petitioner cites this material for the proposition that “*Sinclair discloses* that erase blocks that span multiple planes may be grouped together into a metablock [zone] and that data may be written to the set of erase blocks [sets of erase units] of a specified metablock [specified zone] using a multi-plane write command such that the data is written concurrently to two or more erase blocks of the set of erase blocks,” Pet., 66-67, but Petitioner never shows how Sinclair discloses this. *But see* Pet. 67 (referring to this as “Sinclair’s teaching” in further arguing that a POSITA “would have been motivated to combine *Sinclair’s teaching* with Takeo”).

To the extent that Petitioner’s chain of citations can be regarded as arguing that “the general knowledge of a POSITA” (Pet. 26) somehow teaches modifying Takeo in combination with Son, Sinclair, and Olbrich as to satisfy the claim limitation, this argument is vaguely supported, inspecific to Petitioner’s grounds, conclusory, and entitled to no weight. After following that chain of citations, after all, the argument reduces to a string citation to a single reference offered in support

of a statement that “[m]ulti-plane commands were well-known in the art,” Pet., 26 (citing Ex. 1014), and a statement from Ellis that “parallelism has been regarded as a key to achieving the peak performance.” This does not show how the limitation is present in the cited art. *See, e.g.*, 37 C.F.R. § 42.65(a); *see also In re Magnum Oil Tools*, 829 F.3d at 1380; *see also TQ Delta, LLC v. Cisco Sys.*, 942 F.3d 1352, 1360 (Fed. Cir. 2019) (generic testimony to the effect that a POSITA would want to “build something better” “fail[ed] 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”); *Samsung Elecs. Co., Ltd. v. Red Rock Analytics, LLC*, IPR2018-00556, Paper 18, 21 (Aug. 20, 2018) (denying institution) (finding testimony that a proposed modification would provide “increased utility” to be “akin to stating in a conclusory fashion that the combination ‘would have been obvious’”); *Intel Corp. v. VLSI Tech. LLC*, IPR2019-01196, Paper 11, 11 (Jan. 7, 2020) (denying institution) (reference not shown to teach limitation when “Petitioner ... lists a string of citations to [a reference] but does not explain how these disclosures teach” the limitation).

2. Petitioner’s Analogy To Olbrich Does Not Close This Gap In Petitioner’s Analysis.

As an alternative to arguing that Sinclair provides the “common page address” aspect of claim 3 in its combinations involving Takeo, Son, and Sinclair, Petitioner argues that this limitation is taught by Olbrich. Pet., 72-73. Petitioner argues:

As discussed above, Olbrich discloses a multi-plane write command that uses a common page address to concurrently write to pages of a superblock that are on different planes. *See* Ground 2, Claim 3. Olbrich’s superblocks and super pages are similar to Sinclair’s metablocks and metapages.

Pet., 73. But Petitioner does not elaborate upon what it means by “Olbrich’s superblocks and super pages are similar to Sinclair’s metablocks and metapages,” here, or elsewhere. *See, e.g.*, Pet., 32-33 (describing Olbrich in isolation, without comparing it to Sinclair). Claim 3 also depends upon claim 1, and Petitioner’s argument for claim 3 incorporates and depends upon its argument claim 1. *See* Pet., 72. In analyzing claim 1, Petitioner identifies the claimed “zones” (which include the “specified zone” of claim 3’s recitation of writing data to “erase units ... of the specified zone using a common page address”) with “zones” in Takeo, not “superblocks” in Sinclair. *See* Pet., 56 (identifying the claimed “zones” with Takeo’s “zones”), 62 (same), 64 (identifying the claimed “specified zone” with a “zone” in Takeo). Petitioner’s unexplained and unsupported analogy therefore fails to show how Olbrich teaches the limitation in the Ground 4 combination of Takeo, Son, Sinclair, and Olbrich. *See, e.g., Valeo N. Am., Inc. v. Magna Elecs., Inc.*, IPR2014-01206, Paper 13, 15 (Dec. 23, 2014) (denying institution) (“Petitioner’s obviousness analysis is incomplete in that it ... does not explain persuasively the relevance of the evidence, and presents conclusory assertions regarding how the

references are combined.”); *Matterport, Inc. v. Appliance Computing III, Inc.*, IPR2021-00939, Paper 15, 16-17 (Nov. 23, 2021) (denying institution) (“Petitioner does not identify differences between [cited art] and element 1.2 or how Petitioner proposes to combine or modify the asserted prior art to meet element 1.2.”).

IV. CONCLUSION

For each of the foregoing reasons, the Board should not institute trial.

Respectfully submitted,

Dated: October 20, 2025

By: /s/ Kenneth Weatherwax

Kenneth Weatherwax (Reg. No. 54,528)

Nathan Lowenstein (*pro hac vice*)

Dennis Courtney (*pro hac vice*)

Colette Woo (*pro hac vice*)

LOWENSTEIN & WEATHERWAX LLP

Xu Zhou (*pro hac vice*)

CALDWELL CASSADY CURRY PC

Attorneys for Patent Owner

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Preliminary Response (the “POPR”) consists of 13227 words, excluding table of contents, table of authorities, certificate of service, this certificate, or table of exhibits. The POPR 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 the paper (Microsoft Word).

Respectfully submitted,

LOWENSTEIN & WEATHERWAX LLP

Dated: October 20, 2025

By: /s/ Dennis Courtney

Dennis Courtney
Attorney for Patent Owner
(310) 307-4500

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the following documents were served by electronic service, by agreement between the parties, on the date below:

PATENT OWNER'S PRELIMINARY RESPONSE

EXHIBITS 2020-2030

The names and addresses of the parties being served are as follows:

James M. Glass

John T. McKee

Nima Hefazi

Gregory M. Miraglia

jimglass@quinnemanuel.com

johnmckee@quinnemanuel.com

nimahefazi@quinnemanuel.com

gregmiraglia@quinnemanuel.com

Respectfully submitted,

Dated: October 20, 2025

By: /s/ Michael Cosley

Michael Cosley