

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PHISON ELECTRONICS CORPORATION,
Petitioner,

v.

VERVAIN, LLC,
Patent Owner.

PGR2025-00011
U.S. Patent No. 11,967,370

**PATENT OWNER'S PRELIMINARY RESPONSE ON THE MERITS AND
OTHER NON-DISCRETIONARY CONSIDERATIONS**

TABLE OF CONTENTS

	Page
EXHIBIT LIST	vii
I. INTRODUCTION	1
II. OVERVIEW OF THE CHALLENGED PATENT AND THE CHALLENGED CLAIMS	2
A. SLC and MLC Flash	3
B. Address Table.....	5
C. Data Integrity Tests	5
D. Independent Claims.....	10
III. PERSON OF ORDINARY SKILL IN THE ART	14
IV. CLAIM CONSTRUCTION	14
V. GROUND 1: 35 U.S.C. § 112(a).....	16
A. The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(a)	17
1. 1.A: “Memory Space”.....	17
2. 1.B: “Memory Element”	19
3. 1.C: “At Least One...Memory Element”.....	20
4. 1.D: “Transfer” Of “Stored Data” To SLC.....	22
5. 1.E: RRAM And MA-GRAM (Claims 6 And 10)	23
6. 1.F: “Temporary Transfer” From MLC To SLC (Claims 14 And 15)	24
VI. GROUND 2: 35 U.S.C. § 112(b)	25
A. The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(b)	25

1.	2.A: “Memory Space”	25
2.	2.B: “Memory Element”	28
3.	2.C: “Controller” Limited By Claimed Functions	29
4.	2.D: Claims Reciting Method Steps In The Context Of A System	34
5.	2.E: On What And When The Data Integrity Test Is To Be Performed	36
VII.	GROUND 3: 35 U.S.C. § 103	37
A.	Overview Of The Cited Prior Art.....	37
1.	Gavens (Ex. 1045).....	38
2.	Gorobets 912 (Ex. 1049).....	46
3.	Paley (Ex. 1051).....	48
4.	Gorobets PCT (Ex. 1058)	50
B.	The Petition Does Not Establish That Independent Claims 1 Or 18 Would Have Been Obvious Over Gavens.....	51
1.	[1b] and [18b]: “SLC non-volatile memory element”	51
2.	[1d] and [18d]: Address Map	56
3.	[1e] and [18e]: “data integrity test performed at the controller based upon received <i>data retained at the controller</i> ”	59
C.	The Petition Does Not Establish That Dependent Claims 2-17 Would Have Been Obvious Over Gavens.....	73
VIII.	CONCLUSION.....	73

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Acon Inc. v. Amo Dev., LLC</i> , IPR2021-00858, 2022 WL 17585777 (P.T.A.B. Dec. 12, 2022).....	24, 52
<i>Allergan et al. v. BTL Healthcare</i> , PGR2021-00020, Paper 16 (P.T.A.B. Jun. 16, 2021).....	22
<i>Amazon.com, Inc., et al. v. Avago Technologies General IP</i> <i>(Singapore) PTE. LTD.</i> , IPR2017-01112, Paper 11 (P.T.A.B. Aug. 23, 2017).....	58
<i>Application of Ratti</i> , 270 F.2d 810 (C.C.P.A. 1959)	56, 72
<i>Ariad Pharms., Inc. v. Eli Lilly & Co.</i> , 598 F.3d 1336 (Fed. Cir. 2010)	17, 20
<i>Atlas Powder Co. v. E.I. du Pont de Nemours & Co.</i> , 750 F.2d 1569 (Fed. Cir. 1984)	22
<i>Barkan Wireless IP Holdings, L.P. v. Samsung Elecs. Co., Ltd.</i> , 2019 WL 497902 (E.D. Tex. Feb. 7, 2019).....	32
<i>Callaway Golf Co. v. Acushnet Co.</i> , 576 F.3d 1331 (Fed. Cir. 2009)	50
<i>Capella Photonics, Inc. v. Fujitsu Network Commc'ns, Inc.</i> , 2021 WL 465430 (E.D. Tex. Feb. 9, 2021).....	31
<i>Captioncall, LLC v. Ultratec, Inc.</i> , IPR2020-01215, 2021 WL 278381 (P.T.A.B. Jan. 27, 2021)	57, 63
<i>In re De Seversky</i> , 474 F.2d 671 (C.C.P.A. 1973)	50
<i>Graham v. John Deere Co. of Kansas City</i> , 383 U.S. 1 (1966).....	69

<i>Intelligent Automation Design, LLC v. Zimmer Biomet CMF & Thoracic, LLC</i> , 799 F. App'x 847 (Fed. Cir. 2020)	33, 34
<i>IPXL Holdings, L.L.C. v. Amazon.com, Inc.</i> , 430 F.3d 1377 (Fed. Cir. 2005)	35, 36
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006)	53
<i>Martek Biosciences Corp. v. Nutrinova, Inc.</i> , 579 F.3d 1363 (Fed. Cir. 2009)	17
<i>Masimo Corp. v. Apple Inc.</i> , IPR2023-00745, Paper 44 (P.T.A.B. Oct. 15, 2024).....	58
<i>Mastermine Software v. Microsoft Corp.</i> , 874 F.3d 1307 (Fed. Cir. 2017)	35, 36
<i>Microbiotica, Limited v. Board of Regents, The University of Texas System</i> , PGR2023-00026, Paper 12 (P.T.A.B. Nov. 21, 2023)	17, 21, 23
<i>Nautilus, Inc. v. Biosig Instruments, Inc.</i> , 572 U.S. 898 (2014).....	37
<i>Nearmap US, Inc. v. Eagle View Technologies, Inc.</i> , IPR2024-00716, Paper 9 (P.T.A.B. Oct. 9, 2024).....	58
<i>Netflix, Inc. v. DivX, LLC</i> , 84 F.4th 1371 (Fed. Cir. 2023)	59
<i>Niazi Licensing Corp. v. St. Jude Med. S.C., Inc.</i> , 30 F.4th 1339 (Fed. Cir. 2022)	26, 29, 37
<i>Rain Computing, Inc. v. Samsung Elecs. Am., Inc.</i> , 989 F.3d 1002 (Fed. Cir. 2021)	32, 33
<i>Samsung Elecs. Co., Ltd. v. Mojo Mobility Inc.</i> , IPR2023-01099, 2024 WL 457731 (P.T.A.B. Feb. 6, 2024)	56
<i>Sound View Innovations, LLC v. Facebook, Inc.</i> , 2017 WL 2221177 (D. Del. May 19, 2017)	33

<i>In re Stepan Co.</i> , 868 F.3d 1342 (Fed. Cir. 2017)	52, 57, 70
<i>StrikeForce Techs. Inc. v. PhoneFactor Inc.</i> , 2015 WL 5708577 (D. Del. Sept. 29, 2015).....	34
<i>Sysmex Corp. v. Beckman Coulter, Inc.</i> , 2021 WL 1259710 (D. Del. Apr. 6, 2021)	30
<i>Telcordia Techs., Inc. v. Cisco Sys., Inc.</i> , 612 F.3d 1365 (Fed. Cir. 2010)	30
<i>Typhoon Touch Techs., Inc. v. Dell, Inc.</i> , 659 F.3d 1376 (Fed. Cir. 2011)	34
<i>Va. Innovation Scis., Inc. v. Amazon.com, Inc.</i> , 2019 WL 4259020 (E.D. Tex. Sept. 9, 2019).....	32, 33
<i>Vervain, LLC v. Phison Electronics Corp.</i> , 1:24-cv-00259 (W.D. Tex.)	15, 25
<i>Via Vadis, LLC v. Buffalo Americas, Inc.</i> , 2016 WL 5239626 (W.D. Tex. Sept. 20, 2016), <i>rev'd in part on</i> <i>other grounds Vadis, LLC v. Blizzard Ent., Inc.</i> , 815 F. App'x 539 (Fed. Cir. 2020).....	30, 32
<i>Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.</i> , 200 F.3d 795 (Fed. Cir. 1999)	15

Statutes

35 U.S.C. § 103	37
35 U.S.C. § 112	29, 30, 31, 33
35 U.S.C. § 112(a)	1, 16, 17, 21
35 U.S.C. § 112(b)	1, 25
35 U.S.C. § 112(f).....	29
35 U.S.C. § 312(a)(3).....	58
35 U.S.C. § 324(a)	1

Other Authorities

37 C.F.R. § 42.6(a)(3)	24, 52
37 C.F.R. § 42.22(a)(2)	53, 58

EXHIBIT LIST

Exhibit No.	Description
2001	Declaration of Dr. Sunil Khatri
2002	Chen et al., <i>Ultra MLC Technology Introduction</i> , Advantech Technical White Paper (Oct. 5, 2012) (“Chen”)
2003	Excerpts from Micheloni et al., <i>Inside NAND Flash Memories</i> (1 st ed. 2010) (“Micheloni”)
2004	[RESERVED]
2005	[RESERVED]
2006	<i>Vervain v. Western Digital Corp. et al.</i> , No. 6:21-cv-488-ADA, Dkt. 41 (W.D. Tex. Jan. 24, 2022) (Claim Construction Order)
2007	U.S. Patent App. No. 2007/0271609 assigned to Phison Electronics Corp.
2008	[RESERVED]
2009	[RESERVED]
2010	[RESERVED]
2011	[RESERVED]
2012	U.S. Patent No. 8,194,452 to G. R. Mohan Rao (incorporated by reference by the Challenged Patent)
2013	Oxford Dictionary of Computing (Sixth Edition, 2008) (page 312, defining “memory element”)
2014	U.S. Patent App. No. 2010/0186084 currently assigned to Kingston Digital, Inc.
2015	Microsoft Computer Dictionary (5th ed. 2002)
2016	[RESERVED]

Exhibit No.	Description
2017	[RESERVED]
2018	[RESERVED]
2019	[RESERVED]
2020	[RESERVED]
2021	[RESERVED]
2022	[RESERVED]
2023	[RESERVED]
2024	[RESERVED]
2025	[RESERVED]
2026	[RESERVED]
2027	[RESERVED]
2028	[RESERVED]
2029	<i>Phison Electronics Corp. v. Vervain, LLC</i> , PGR2024-00047, Paper 2 (P.T.A.B. Aug. 27, 2024) (petition for post grant review of U.S. Patent No. 11,830,546)
2030	[RESERVED]
2031	Preliminary Claim Constructions for <i>Vervain, LLC v. Phison Electronics Corporation</i> , No.1:24-cv-259-ADA and <i>Vervain, LLC v. Kingston Technology Company., et al.</i> , No. 1:24-cv-254-ADA – email from Mr. Brown received on February 5, 2025.
2032	Markman Hearing Transcript dated February 6, 2025, for <i>Vervain, LLC v. Phison Electronics Corporation</i> , No. 1:24-cv-259-ADA and <i>Vervain, LLC v. Kingston Technology Company., et al.</i> No. 1:24-cv-254-ADA.
2033	[RESERVED]

Exhibit No.	Description
2034	[RESERVED]
2035	[RESERVED]
2036	[RESERVED]
2037	[RESERVED]
2038	[RESERVED]
2039	<i>Phison Electronics Corp. v. Vervain, LLC</i> , PGR2024-00047, Paper 9 (P.T.A.B. Mar. 17, 2024) (Decision Denying Institution of Post-Grant Review of U.S. Patent No. 11,830,546)
2040	[RESERVED]
2041	<i>Phison Electronics Corp. v. Vervain, LLC</i> , PGR2024-00048, Paper 9 (P.T.A.B. April 28, 2025) (Decision Denying Institution of Post-Grant Review of U.S. Patent No. 11,854,612)

I. INTRODUCTION

Vervain, LLC (“PO” or “Vervain”) submits this Preliminary Response in response to the Petition for Post Grant Review (PGR) (Paper 2) of U.S. Patent No. 11,967,370 (“370 Patent” or “the Challenged Patent”) filed by Phison Electronics Corporation (“Phison” or “Petitioner”). The Petition includes four grounds challenging claims 1-18 (“Challenged Claims”). The first two grounds include cursory challenges under 35 U.S.C. §§ 112(a), and 112(b). The third ground relies on single-reference obviousness based on U.S. Patent No. 8,634,240 (“Gavens”) (Ex. 1045). These grounds fall far short of the mark, as explained below.

The Board should deny institution because Petitioner’s grounds are meritless. The asserted grounds are cursory and lack adequate explanation—some span only a few sentences—and the obviousness ground fails to reconcile key differences between the Challenged Claims and Gavens. Specifically, the Challenged Claims require a data integrity test performed at the controller *based upon received data retained at the controller*, a feature that Gavens neither discloses nor suggests.

Moreover, the Board already denied institution of petitions presenting similar grounds. Ex. 2039 (denying institution of PGR2024-00047); Ex. 2041 (denying institution of PGR2024-00048). As reflected in those decisions, Petitioner has not established it is more likely than not to prevail on any ground. 35 U.S.C. § 324(a).

Accordingly, PO requests that the Board deny institution of the Petition.

II. OVERVIEW OF THE CHALLENGED PATENT AND THE CHALLENGED CLAIMS

The Challenged Patent, entitled “Lifetime Mixed Level Non-Volatile Memory System” was filed on December 20, 2023, and has an effective filing date of July 19, 2011. Ex. 1001, Cover. Dr. G. R. Mohan Rao is the sole named inventor of the Challenged Patent.

At a high level, the Challenged Patent describes a reliable flash memory storage system combining both single-level cell (SLC) and multi-level cell (MLC) non-volatile memories.¹ Ex. 1001, Abstract; Ex. 2001, ¶37. Prior to the Challenged Patent, Dr. Rao recognized that “MLC NAND flash SSDs are slowly replacing and/or coexisting with SLC NAND flash in newer SSD systems” because “MLC flash memory is less expensive than SLC flash memory[] on a cost per bit basis.” Ex. 1001, 3:34-35, 5:47-50. However, while “MLC NAND flash enjoys greater density than SLC NAND flash” it comes “at the cost of a decrease in access speed and lifetime (endurance).” *Id.*, 3:50-53. As a result, various hybrid systems combining SLC and MLC (among others) are taught in the Challenged Patent, to

¹ Non-volatile memories can store information even after the system is powered off. Flash memory is a specific type of non-volatile memory, where data is stored in “blocks” of “pages.” Ex. 1001, 2:59-63; Ex. 2001, ¶27.

combine the benefits of both types of non-volatile flash storage at a low cost. *Id.*, 4:4-11.

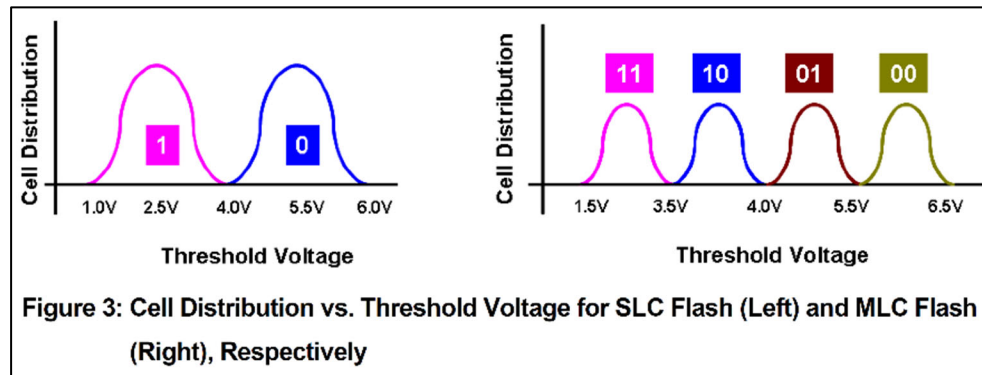
The Challenged Patent addresses improvements and solutions for managing the writing of data optimally for improved reliability and lifetime (endurance) of such hybrid memory systems. *Id.*, 4:4-11. Specifically, the Challenged Claims are directed to specific techniques for efficiently using SLC and MLC flash to improve the overall performance of the memory. *Id.*, claim 1. For example, if data stored in the MLC fails a data integrity test, then it is transferred to higher-performance SLC or less-used MLC. *Id.* By doing so, the number of errors is reduced, and overall endurance of the memory is increased. Ex. 1001, 4:9-11.

A. SLC and MLC Flash

SLC memory stores 1 bit per cell, and MLC memory stores more than 1 bit per cell. Ex. 1001, 2:21-27; Ex. 2001, ¶¶32-34. As noted above, there are advantages and disadvantages to SLC and MLC flash. In general, SLC is faster and less prone to errors, but requires more space and power to store a given amount of data. Ex. 1001, 1:64-2:2. The opposite is true of MLC. MLC flash is slower and more prone to errors, but stores data more densely with less power consumption. *Id.*, 3:50-53.

SLC and MLC flash memories both use the same type of transistor called a floating gate transistor. *Id.*, 3:59-62. They both store a charge in the floating gate

of each transistor (cell), which changes the threshold voltage of the transistor. Ex. 2001, ¶32. The memory uses the threshold voltage to determine what bit, or bits, were stored in the transistor. The MLC cell in the figure below illustrates threshold voltages for a 2-bit MLC cell.



Ex. 2002, 3.

The primary difference between SLC and MLC is what data each threshold voltage represents. Ex. 2001, ¶33. With SLC flash, the transistor stores only a 1 or 0, so a wide range of threshold voltages can be allotted to a single bit. Ex. 1001, 3:47-48. This allows for faster and more reliable memory access. On the other hand, MLC flash must be slowly and carefully programmed to a narrower, more precise range of threshold voltages, with each threshold voltage range representing a specific pair of bits (*see* figure above, which shows four pairs of bits—11, 10, 01, and 00—corresponding to smaller ranges of threshold voltages compared to the SLC). Ex. 1001, 3:48-50.

B. Address Table

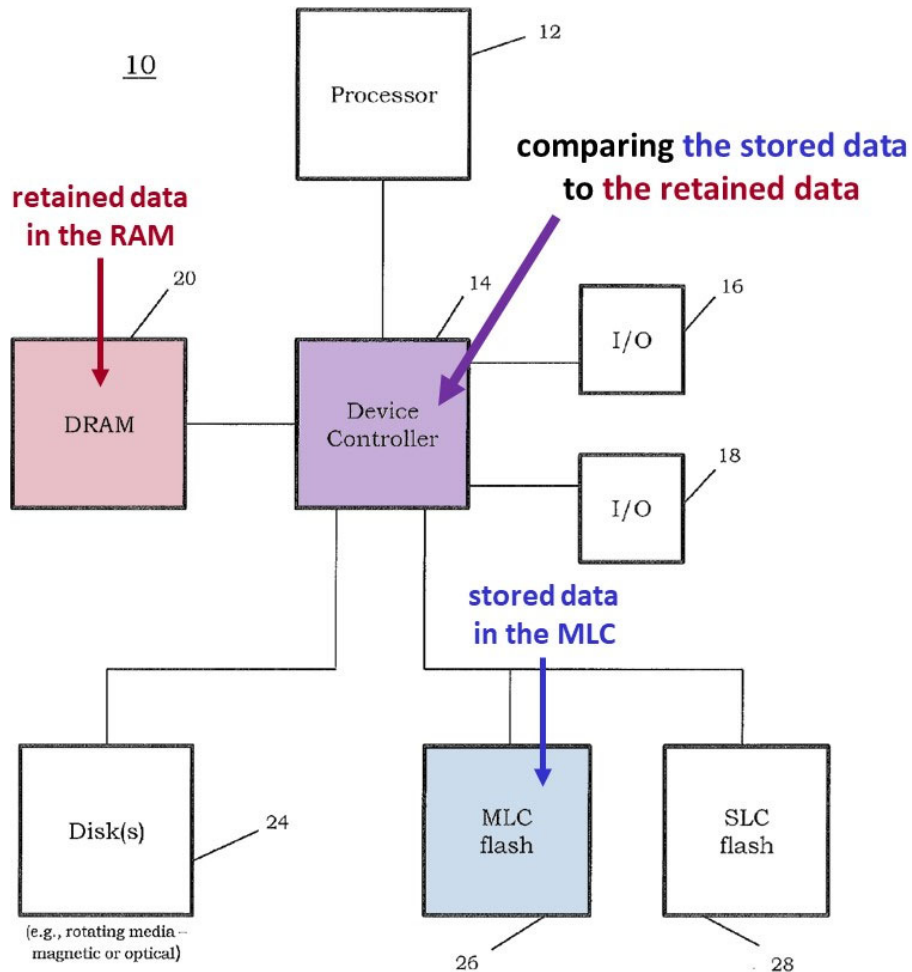
To provide wear leveling, garbage collection, and bad block management, a translation layer is used to map logical addresses to actual physical memory locations. Ex. 2003, 40-42; Ex. 1001, 3:10-44; *see also* Ex. 2001, ¶35. As part of this translation layer, “tables are widely used in order to map sectors and pages from logical to physical.” Ex. 2003, 40; *see also* Ex. 1001, 3:27-31. These tables map logical blocks to physical blocks. Ex. 2003, 40-42; Ex. 1001, 3:27-31. Using a “block” or similar granularity is important, since flash memory is arranged so that when erasing and rewriting data, a whole block is “erased together.” Ex. 2003, 22; *see also* Ex. 1001, 2:66-3:9. Dr. Rao explained that “[t]he address ranges within the translation table will assume some minimum quantum, such as, for example, one block....” Ex. 1001, 5:63-67. Dr. Rao further explained that memory is written and mapped on the granularity of a “quantum,” such as a block or page. *Id.*, 5:63-67; Figs. 3A-B.

C. Data Integrity Tests

As mentioned above, when data is stored in MLC memory, it is more prone to errors, and some data is more prone to errors than other data. Ex. 2001, ¶36. One reason for this is that the threshold voltage intervals for MLC memory are smaller than the intervals for SLC memory, and thus, errors can occur when writing or reading the data. *Id.*, ¶36. Errors can also be caused by the data stored in

neighboring cells. *Id.* A data integrity test checks the integrity of the data (*i.e.*, whether errors have occurred). This test can be run immediately after data is written, or at a later time. If the test reveals a problem such as corrupt data, the data can be remapped to SLC (which is less error-prone), and the address table is modified accordingly. Ex. 1001, 6:3-7:6. Alternately, MLC data can be remapped to other MLC blocks, and the address table is then modified accordingly. *Id.*, 3:21-44.

The Challenged Patent describes a particular way of writing the data to the MLC and performing the data integrity test. Dr. Rao recognized that there were advantages in using a dynamic random access memory (DRAM) to assist with both the write access operation and data integrity test. As seen in Figure 1 below, the controller (purple) controls a DRAM (red).



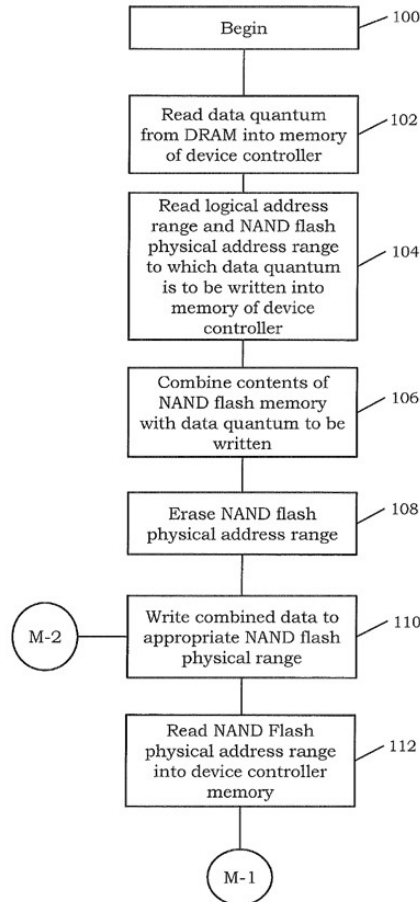
Ex. 1001, Fig. 1 (annotations added²), 5:43-44.

The DRAM is used during a write access operation to the MLC flash (highlighted blue). The data to be written to the MLC flash is maintained (or retained) in the DRAM. Ex. 1001, 6:1-6:3. If time permits, the controller performs a data integrity test on the data that was stored in the MLC. *Id.*, 6:3-6:6. If the stored

² Unless otherwise noted, PO added coloring to Figures.

data fails the data integrity test, the address range for the stored data is remapped to a new location (*e.g.*, the SLC flash). *Id.*, 6:6-6:9.

Figures 3A and 3B show in more detail how the DRAM is used during the write access operation and data integrity test.



Id., Fig. 3A.

The write access operation begins at step 100. *Id.*, 6:26-29. In step 102, the data to be written to the MLC is read from the DRAM into the controller memory. *Id.*, 6:29-31. In step 104, the controller obtains the logical and physical address ranges for the data to be written to the MLC. *Id.*, 6:31-34. In step 106, the controller

combines the data from the DRAM with data from the MLC. *Id.*, 6:34-36. In step 108, the controller erases the physical address range where the data is being written. *Id.*, 6:36-37. Finally, in step 110, the combined data is written to the MLC flash. *Id.*, Fig. 3A (step 110), 6:37-39.

The data integrity test begins at the bottom of Figure 3A and continues to Figure 3B (below). In step 112, the data that was stored in the MLC (“the stored data”) is read into the controller memory. *Id.*, 6:39-41. In step 114, the stored data is compared to the data that was retained in the DRAM (“the retained data”). *Id.*, 6:42-47.

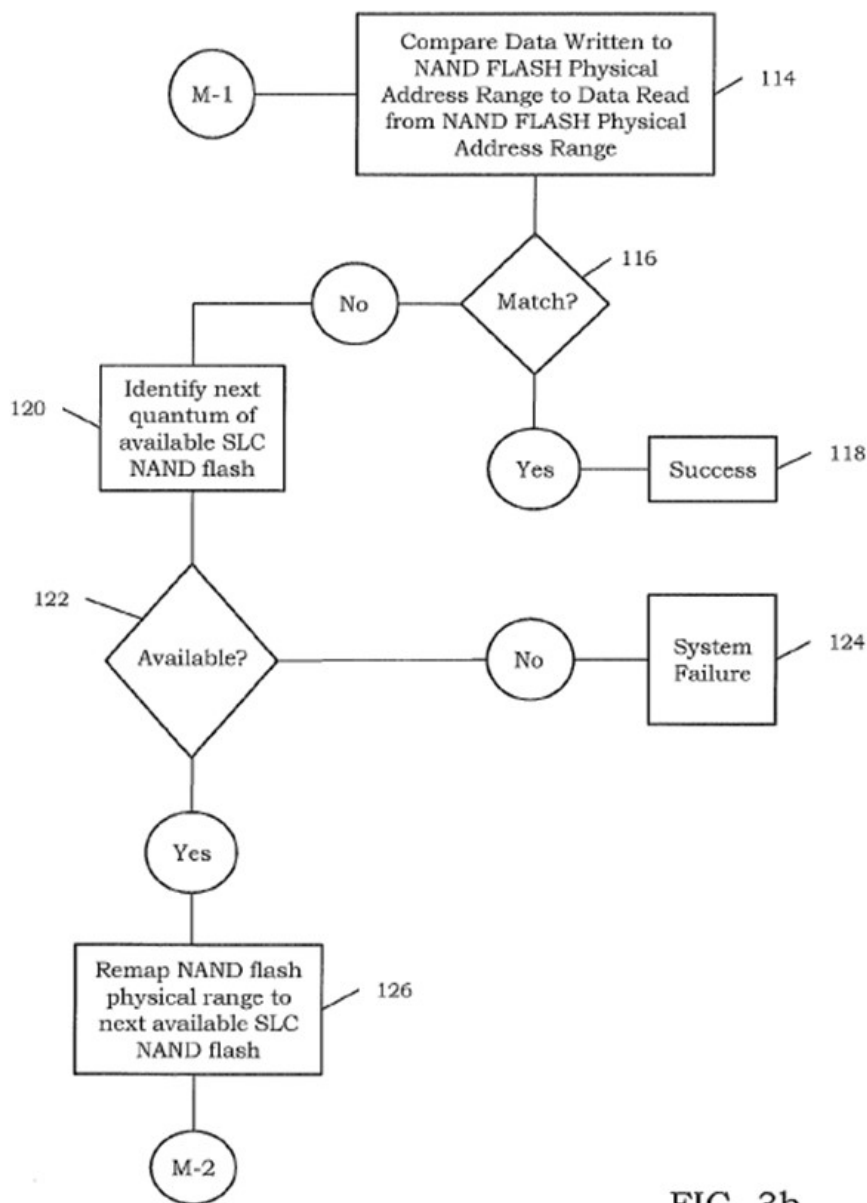


FIG. 3b

Id., Fig. 3B.

If there is a match, the data was successfully stored in the MLC. *Id.*, 6:47-49.

If not, the retained data is written to another location (*e.g.*, SLC). *Id.*, 6:49-53.

D. Independent Claims

The Challenged Patent includes two independent claims: Claims 1 and 18.

1. Claim 1

Claim 1 of the Challenged Patent requires a system for storing data comprising: (1) MLC non-volatile memory; (2) SLC non-volatile memory; and (3) a data integrity test performed at the controller based upon received data retained at the controller (highlighted red below). The data retained at the controller is a key feature of the invention. The data retained at the controller is particularly important to the “data integrity test” in [1e].

Claim 1	
[1Pre]	A system for storing data comprising:
[1a]	an MLC non-volatile memory space including at least one MLC non-volatile memory element;
[1b]	an SLC non-volatile memory module space including at least one SLC non-volatile memory element; and
[1c]	a controller coupled to the at least one MLC non-volatile memory element and the at least one SLC non-volatile memory element wherein the controller is adapted to:
[1d]	a) maintain an address map of the MLC and SLC non-volatile memory spaces, the address map comprising a list of logical address ranges accessible by a computer system, the list of logical address ranges having a minimum quanta of addresses, wherein each entry in the list of logical address ranges maps to a similar range of physical addresses within

	either the at least one SLC non-volatile memory element or within the at least one MLC non-volatile memory element;
[1e]	b) [1e1] determine if a range of addresses listed by an entry and mapped to a similar range of physical addresses within the at least one MLC non-volatile memory element, fails a data integrity test performed at the controller based upon received data retained at the controller and which received data is stored in the at least one MLC non-volatile memory element as stored data, and, in an event of such a failure, the controller [1e2] remaps the entry to a next available equivalent range of physical addresses within the at least one SLC non-volatile memory element and, [1e3] the stored data is transferred from the at least one MLC non-volatile memory element to the at least one SLC non-volatile memory element.

2. Claim 18

Claim 18 is a method claim that is similar Claim 1. The method described by Claim 18 refers to a method for storing data comprising: (1) MLC non-volatile memory; (2) SLC non-volatile memory; and (3) a data integrity test performed at the controller based upon received data retained at the controller.

Claim 18	
[18Pre]	A method for storing data comprising:
[18a]	providing an MLC non-volatile memory space including at least one MLC non-volatile memory element;

[18b]	providing an SLC non-volatile memory space including at least one SLC non-volatile memory element; and
[18c]	maintaining, using a controller coupled to the at least one MLC non-volatile memory element and the at least one SLC non-volatile memory element, ³
[18d]	an address map of the MLC and SLC non-volatile memory spaces, the address map comprising a list of logical address ranges accessible by a computer system, the list of logical address ranges having a minimum quanta of addresses, wherein each entry in the list of logical address ranges maps to a similar range of physical addresses within either the at least one SLC non-volatile memory element or within the at least one MLC non-volatile memory element;
[18e]	[18e1] determining, using the controller, if a range of addresses listed by an entry and mapped to a similar range of physical addresses within the at least one MLC non-volatile memory element, fails a data integrity test performed at the controller based upon received data retained at the controller and which received data is stored in the at least one MLC non-volatile memory element as stored data; in an event of such a failure, [18e2] remapping, using the controller, the entry to a next available equivalent range of physical addresses within the at least one

³ In its Petition, Petitioner incorrectly contends that limitation [18c] ends with a semicolon. Petition, 64. Limitation [18c] instead ends with a comma. *See* Ex. 1001, 9:1-3.

	SLC non-volatile memory element; and [18e3] transferring the stored data from the at least one MLC non-volatile memory element to the at least one SLC non-volatile memory element using the controller.
--	--

III. PERSON OF ORDINARY SKILL IN THE ART

For purposes of this Preliminary Response only, PO adopts Petitioner’s definition of a person of ordinary skill in the art (POSA). Petition, 2; Ex. 2001, ¶¶22-26.⁴

IV. CLAIM CONSTRUCTION⁵

Petitioner proposed construction of eight terms: “MLC memory modules” (and purportedly underlying term, “MLC non-volatile memory”), “SLC memory module” (and purportedly underlying term, “SLC non-volatile memory”), “MLC memory space,” “SLC memory space,” “MLC memory element,” “SLC memory element,”⁶ “enhance a lifetime” (dependent claim 16), and “transfer.” Petition, 26-

⁴ PO submits the declaration of Dr. Sunil Khatri, an expert in the field of the Challenged Patent. Ex. 2001, ¶¶1-26.

⁵ PO addresses the indefiniteness grounds separately in Section VI below.

⁶ Petitioner proposes constructions for these terms (“MLC memory modules,” “SLC memory module,” “MLC memory space,” “SLC memory space,” “MLC memory

30. PO disagrees with Petitioner’s proposed constructions, but construction of these terms is not necessary to resolve the parties’ dispute regarding whether institution should be granted. *See Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999). Accordingly, PO does not contest Petitioner’s proposed constructions for the limited purpose of this Preliminary Response.

Notably, for “SLC memory module” and “MLC memory modules,” Petitioner proposes a *different* construction here than it proposed in its parallel petition for post grant review of U.S. Patent No. 11,830,546, of which the Challenged Patent is a continuation. *Compare* Petition, 27 *with* PGR2024-00047, Paper 2 (Petition) at 24 (submitted here as Ex. 2029). However, neither of Petitioner’s constructions makes sense.

Further, some of Petitioner’s current proposals mirror those already presented in the parallel district court litigation between Vervain and Phison (*Vervain, LLC v. Phison Electronics Corp.*, 1:24-cv-00259 (W.D. Tex.) (the “Parallel Litigation”)), where they were fully briefed and, during a *Markman* hearing on February 6, 2025, ***already rejected by the Court.*** *See* Exs. 2031-2032. The Court rejected adopted

element,” and “SLC memory element”) but none of those terms appear as presented by Petitioner in the claims. Petition, 27; *see, e.g.*, Ex. 1001, Claim 1 (including “MLC non-volatile memory space” instead of “MLC memory space”).

PO's proposed constructions for "MLC non-volatile memory," "SLC non-volatile memory," and "memory element," "memory space," "data integrity test," and "remapping...and transfer of data...to achieve enhanced endurance." *See* Ex. 2031, 3-5 (providing preliminary constructions for these terms prior to the *Markman* hearing); Ex. 2032 (hearing transcript from the *Markman* hearing, where the Court adopted all its preliminary constructions). For example, the Court construed "MLC non-volatile memory" and "SLC non-volatile memory" to refer to "nonvolatile memory that stores one bit of information per cell" and "nonvolatile memory that stores multiple bits of information per cell," respectively.

In addition, in an earlier litigation not involving Petitioner Phison, the district court adopted the same constructions for "SLC non-volatile memory" and "MLC non-volatile memory." *See* Ex. 2006, 2; *see also* Ex. 2031, 3-4; Ex. 2032. Thus, Petitioner's current proposals—which focus on the presence of certain circuitry rather than the number of bits stored per cell—are inconsistent with the Court's repeated prior constructions.

V. GROUND 1: 35 U.S.C. § 112(a)

Petitioner asserts that seven terms lack written description support and/or enablement. Petition, 30-35. These challenges fail.

Moreover, the Board has already considered—and rejected—four of Petitioner’s six § 112(a) arguments in denying institution of PGR2024-00047. Ex. 2039, 30-32, 33-34.

A. The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(a)

A term has sufficient written description support if “the disclosure of the application relied upon reasonably conveys to those skilled in the art that the inventor had possession of the claimed subject matter as of the filing date.” *Ariad Pharms., Inc. v. Eli Lilly & Co.*, 598 F.3d 1336, 1351 (Fed. Cir. 2010). “To meet the enablement requirement, the specification of a patent must teach those skilled in the art how to make and use the full scope of the claimed invention without undue experimentation.” *Martek Biosciences Corp. v. Nutrinova, Inc.*, 579 F.3d 1363, 1378 (Fed. Cir. 2009) (internal quotations and citation omitted).

1. 1.A: “Memory Space”

“Memory space” is clearly disclosed in the specification and is fully enabled. *See* Ex. 2001, ¶¶53-54. The Board has already rejected Petitioner’s argument that “memory space” lacks written description. *See* Ex. 2039, 30-31.

Before reaching the merits, however, this ground should be rejected outright as the Petition does not clearly identify or explain the basis for its challenge—indeed, Petitioner’s argument spans only three sentences. Petition, 31; *see Microbiotica, Limited v. Board of Regents, The University of Texas System*, PGR2023-00026,

Paper 12 at 24, 27 (P.T.A.B. Nov. 21, 2023) (denying institution of written description grounds when petitioner “d[id] not provide sufficient analysis of the full disclosure of the [challenged] patent”).

The Petition appears to argue that because the specification uses the term “memory space” to refer to a “storage capacity,” it is not proper for the claims to refer to “memory space” in any other sense. Petition, 31. But in fact, the specification’s use of the term “memory space” is entirely consistent with how the term is used in the claims to describe the space where data is stored. For example, the specification explains that the “total non-volatile *memory storage space*=MLC+SLC.” Ex. 1001, 5:31. This indicates that there are two spaces—MLC and SLC—where data is stored, and the total space is the combination of the two. Ex. 2001, ¶54. Figures 1, 2A-2B, and 4 of the Challenged Patent show examples of MLC and SLC memory space. For example, Figure 4 is shown below:

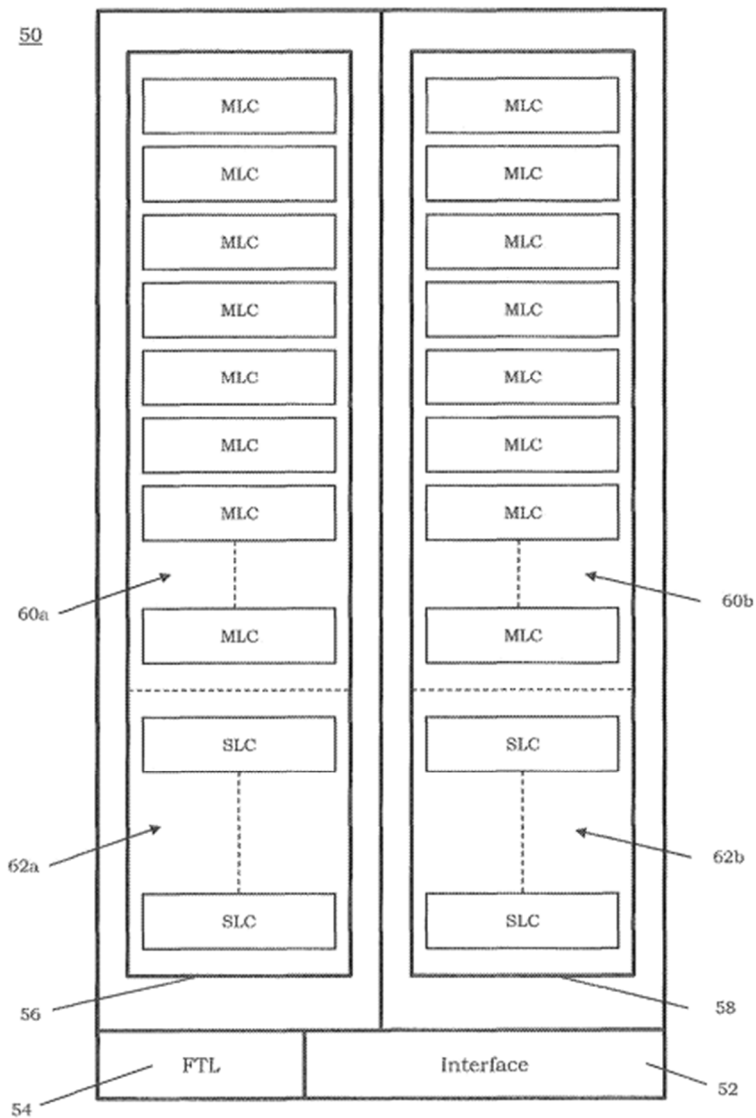


FIG. 4

Ex. 1001, Fig. 4.

2. 1.B: “Memory Element”

The Board has already determined that “memory element” has written description support. *See* Ex. 2039, 31-32.

“Memory elements” are clearly disclosed in the specification, including in a patent (Ex. 2012) which the Challenged Patent incorporates by reference. Ex. 2001,

¶55. In another threadbare argument, Petitioner suggests that “memory element” is not “mentioned...in the specification.” Petition, 31-32. But that is not the test for written description. The Federal Circuit has held that the specification does not need to use the exact language of the claims, noting that it is unnecessary to “recite the claimed invention *in haec verba*.” *Ariad Pharms., Inc. v. Eli Lilly & Co.*, 598 F.3d 1336, 1352 (Fed. Cir. 2010) (en banc).

Applying the correct test, a POSA would have understood that the Challenged Patent demonstrates possession of the claimed “memory elements.” See Ex. 2001, ¶55. The specification states that cells of SLC or MLC flash memory may store a single bit or multiple bits per cell, respectively. Ex. 1001, 2:21-27. The specification shows in great detail, albeit with different words, how MLC and SLC memory elements are used to store data. See, e.g., Ex. 1001, Figs. 1, 4, 5:52-7:18. “Memory element” was a readily understood term in the art, and simply refers to an element of memory. See Ex. 2001, ¶55.

3. 1.C: “At Least One...Memory Element”

The Board has already determined that “at least one memory element” has written description support and is fully enabled. See Ex. 2039, 31-32, 41-42.

A POSA would have found that the Challenged Patent’s specification clearly conveys possession of “at least one...memory element” and enables its full scope. Ex. 2001, ¶¶56-57. To start, Petitioner’s ground is again ambiguous, asserting a lack

of “support” (an apparent written description argument) but also suggesting that the system would not be “useful” (unclear, but potentially an enablement argument). Petition, 32-33. The lack of adequate explanation and the failure to present a clear §112(a) analysis is alone a basis to reject this ground. *See Microbiotica*, PGR2023-00026, Paper 12 at 24, 27.

In any event, the specification provides a clear, enabling description of the “*at least one MLC non-volatile* memory element” ([1a]) and “*at least one SLC non-volatile* memory element” ([1b]). For example, Figures 2A-2B (referencing MLC and SLC blocks) and Figure 4 (referencing MLC and SLC “flash memory modules”) disclose MLC and SLC memory elements. Considering this disclosure, a POSA would have understood how to make and use the full scope of the claimed invention without undue experimentation. Ex. 2001, ¶56.

As a practical matter, a POSA would understand that flash memory systems are large, typically containing megabytes, gigabytes, or even terabytes of storage, and comprising thousands of blocks. Ex. 2001, ¶57. But the Petition attempts to argue that a contrived, single page system would not be useful yet technically meet the claim language. Petition, 32-33 (arguing that the claimed system “is met with just one SLC memory element and one MLC memory element” and that “one page of physical memory satisfie[s] the literal requirements of the systems recited in the Challenged Claims”). That is not sufficient to prove unpatentability of a claim for

lack of enablement. Rather, Petitioner must show that the ““number of inoperative embodiments [is] *significant*”” and that a POSA would have needed to ““experiment unduly in order to practice the claimed invention.”” *See Allergan et al. v. BTL Healthcare*, PGR2021-00020, Paper 16 at 22-23 (P.T.A.B. Jun. 16, 2021) (*quoting Atlas Powder Co. v. E.I. du Pont de Nemours & Co.*, 750 F.2d 1569, 1576 (Fed. Cir. 1984)). There is nothing “significant” about a single page embodiment that would never be found in the real world.

4. 1.D: “Transfer” Of “Stored Data” To SLC

The Board already determined that this term is supported by the specification and fully enabled. Ex. 2039, 33-34, 40-41.

The specification clearly discloses and fully enables this term. Ex. 2001, ¶¶58-59. For example, Figures 2A-2B describe remapping a “physical address range” after a “failed data integrity test,” and Figures 3A-3B and the related disclosure describe transferring data to the remapped address. Ex. 1001, 6:10-61 (describing “repeat[ing]” the write process to the “next available” address after a failed data integrity test).

In another cursory challenge spanning only four sentences, Petitioner fails to demonstrate otherwise. Petition, 33-34. Petitioner incorrectly asserts that the claimed “transfer” of “data...found to have failed” is “nonsensical” and “inoperative.” Petition, 33-34. This is plainly incorrect—the claimed “transfer”

does not propagate “failed” data. As a POSA would understand, it is often possible to read degraded data correctly from flash memory after a failure, for example through the use of error correcting codes which are commonplace in the industry. *See* Ex. 2001, ¶¶58-59. As another example, the Challenged Patent’s specification explains that, when data is remapped after a failed data integrity test, “the write is repeated” from “memory within the device controller.” Ex. 1001, 6:24-61, 6:60-61 (returning to write “step 110” upon failure of a data integrity test); Ex. 2001, ¶68. The memory within the device controller would not contain the failed data written to the non-volatile memory. *Id.* Petitioner’s unsubstantiated challenges must be rejected. *See Microbiotica*, PGR2023-00026, Paper 12 at 24, 27.

5. 1.E: RRAM And MA-GRAM (Claims 6 And 10)

In an argument spanning two sentences, Petitioner appears to argue that the use of “resistive random-access memory (RRAM)” for MLC (Claim 6) and “magnetic random-access memory (MA-GRAM)” for SLC is unsupported because they are not “substitutable.” Petition, 34. To the extent Petitioner’s argument can be understood, Petitioner is wrong. A POSA would have understood that RRAM is operable as MLC, and MA-GRAM is operable as SLC. Ex. 2001, ¶60. This would have been clear from the Challenged Patent’s specification, which explains that RRAM, MA-GRAM, and Flash may “replace” each other, including for SLC and MLC. Ex. 1001, 2:17-36 (“RRAM, PCM, MAGRAM and others...may ultimately

replace flash memories...and later replace ‘SLC flash’ and ‘MLC flash.’”); *see also id.*, 5:52-54 (“The device controller 14 maintains a translation table/address map which may include address translations for all devices in the computer system.”).

Further, in its two-sentence Ground, Petitioner improperly cites paragraphs from its Expert Declaration instead of including adequate explanation in the Petition itself. Petition, 34. Petitioner cannot simply cite its expert declaration while omitting cogent arguments/analysis from the Petition itself. 37 C.F.R. § 42.6(a)(3); *Acon Inc. v. Amo Dev., LLC*, IPR2021-00858, 2022 WL 17585777, at *16 (P.T.A.B. Dec. 12, 2022) (“arguments may not be incorporated from an expert’s declaration wholesale into the [p]etition”).

6. 1.F: “Temporary Transfer” From MLC To SLC (Claims 14 And 15)

In another four sentence ground, Petitioner fails to demonstrate that the temporary transfer of data from MLC to SLC required by claims 14 and 15 is unsupported. Petition, 34-35. Petitioner admits that storing data temporarily is a known concept in flash, and never clearly argues that written description support is not had for these claims. Petition, 35 (admitting that “any data in a flash memory location is ‘temporary’ is a given for any updated data”). Petitioner never even cites the specification for its argument. But, contrary to Petitioner’s conclusory argument, a POSA would have found a “temporary” transfer fully supported. Ex. 2001, ¶¶61-62. For example, the Challenged Patent explains that data is “virtual,” and is not

written to the same physical location with every write. Ex. 1001, 3:15-20. Instead, data may move around with each write. Ex. 2001, ¶61.

Further, as shown in Figures 3A-3B, if data can be written to MLC successfully, it may be written there with priority. This is because the disclosed system aims to increase efficiency and decrease costs, which involves prioritizing MLC. Ex. 1001, 6:49-53; *see also id.*, 4:55-59, 5:60-64. Thus, for example, virtual data may be written to SLC only temporarily and could move back to MLC later. *Id.*; Ex. 2001, ¶62.

VI. GROUND 2: 35 U.S.C. § 112(b)

Petitioner asserts that five claim terms are indefinite. Petition, 35-42. These challenges fail. The Board has already rejected all of Petitioner’s indefiniteness challenges in denying institution of PGR2024-00047, which Petitioner filed against a parent of the 370 Patent. *See* Ex. 2039, 49-56. Further, in the Parallel Litigation, the district court already considered—and rejected—most of these arguments as well. *See* Exs. 2031, 2032; *see also* Ex. 2006 (another district court case involving parent patents of the 370 Patent, again rejecting additional indefiniteness challenge).

A. The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(b)

1. 2.A: “Memory Space”

Petitioner suggests in a cursory and ambiguous manner that “memory space” is indefinite because “there is no boundary for the abstract memory space[.]”

Petition, 36. It is unclear exactly what Petitioner is arguing but, notwithstanding, “breadth is not indefiniteness.” *Niazi Licensing Corp. v. St. Jude Med. S.C., Inc.*, 30 F.4th 1339, 1347 (Fed. Cir. 2022) (citation omitted). And the Board already determined that “memory space” is not indefinite in denying institution of related PGR2024-00047. *See* Ex. 2039, 56-57 (“we agree with Patent Owner that an ordinarily skilled artisan would understand ‘memory space’ as describing where data is stored”). Likewise, during a *Markman* hearing in the Parallel Litigation, the Court made this determination as well. *See* Ex. 2032, 31:14-36:25.

A POSA would not find any ambiguity about what this term refers to. Ex. 2001, ¶¶63-67. To the contrary, claim 1 of the Challenged Patent requires “an address map of the MLC and SLC non-volatile memory spaces.” (limitation [1d]). Particularly in light of this context, a POSA would find no ambiguity regarding what “memory space” refers to. Ex. 2001, ¶¶65-66.

“Memory space” simply refers to where data is stored. Ex. 2001, ¶64. This is a simple term that is readily understood by POSAs and lay persons alike.

A POSA reading the claim language, in light of the specification and prosecution history, would have found it abundantly clear from the face of the claim that the “memory space” is simply where data is stored. Ex. 2001, ¶65. Some of the data is stored in volatile memory space; some is stored in non-volatile memory space. *Id.* The data that is stored in non-volatile memory space may be stored in

MLC or SLC memory space. *Id.* To the extent there is any doubt, the surrounding claim language specifies that data may be stored in memory elements, and these memory elements may be mapped into the memory space. Ex. 1001, 7:46-63. Furthermore, the specification makes it clear that the “total non-volatile memory storage space=MLC+SLC.” Ex. 1001, 5:31. In other words, the “memory space” refers to where data is stored, and “memory storage space” refers to the sum of the memories that make up a given memory. Ex. 2001, ¶66.

Petitioner cannot demonstrate that this term is indefinite, when “memory space” is commonly used to refer to “where data are...stored.” Ex. 2003 (Micheloni textbook), 485; *see also id.*, 493. Even Petitioner’s own patents use “memory space” to refer to the “addressable locations” where data is stored. Ex. 2007 (U.S. Patent App. 2007/0271609), ¶0007. When a previous petitioner, Micron, filed an IPR against U.S. Patent No. 10,950,300 (“300 Patent”) (Ex. 1011), of which the Challenged Patent is a continuation, Micron’s expert opined that “memory space” means “a set of addresses” where data is stored. Ex. 2011, ¶63 (“Taken together, a set of addresses is known as an ‘address space’ or ‘memory space.’”). That confirms that a POSA would have fully understood what is meant by “memory space.” Ex. 2001, ¶67.

2. 2.B: “Memory Element”

The Board has already rejected Petitioner’s assertion that “memory element” is indefinite. Ex. 2039, 57-58 (“we agree with Patent Owner that the term “memory element” has a “broad but understood” meaning to an ordinarily skilled artisan”). The Court also determined that “memory element” is not indefinite during the *Markman* hearing. See Ex. 2032, 41:25-43:7.

Contrary to Petitioner’s conclusory argument—spanning three sentences—there is nothing confusing or indefinite about “memory element.” A POSA would have readily understood that a “memory element” is simply an element of memory. Ex. 2001, ¶¶68-73. “Memory element” is a well understood term in the art. For example, U.S. Patent 8,194,452, which the Challenged Patent incorporates by reference (*see* Ex. 1001, 1:38-46 (incorporating “U.S. patent application Ser. No. 12/915,177”)), refers to elements of memory. Ex. 2012, 2:52-54 (referring to “an exemplary NAND memory integrated circuit as one element of the...nonvolatile memory system”). Additionally, when a previous petitioner, Micron, filed an IPR against the 300 Patent, Micron’s expert explained that “flash memory elements” were “well known to any POSA” by 2010. Ex. 2011, ¶59. There is nothing ambiguous about this term, and it should be given the full scope of its plain meaning.

There are even dictionary definitions for this term. Ex. 2013 (Oxford Dictionary of Computing), 312 (defining “memory element” as a “device that stores

one item of information”). And one of Petitioner’s customers, Kingston, has patents that similarly uses “memory element” to broadly refer to an element where data is stored. Ex. 2014 (U.S. Patent App. 2010/0186084), ¶¶[0009], [0017]-[0020]. “These definitions [and references] confirm [this term] would have had [a] broad but understood meaning[] to a skilled artisan.” *Niazi*, 30 F.4th at 1350.

Petitioner argues that “memory element” is indefinite because it could refer to “blocks, pages, or bytes of data” or to “a cell.” Petition, 36. Here again, Petitioner confuses breadth with indefiniteness. *See Niazi*, 30 F.4th at 1347. A “memory element” is simply an element of memory, and the fact that it may be a broad term does not make it indefinite.

3. 2.C: “Controller” Limited By Claimed Functions

Petitioner goes for the “Hail Mary” argument that this term is not only governed by § 112, ¶6, but it is also indefinite for lack of disclosed structure. Petition, 36-38. But claim 1 never use the word “means,” and Petitioner cannot possibly overcome the presumption that § 112, ¶6 does not apply.⁷ Ex. 2001, ¶¶74-84.

⁷ Petitioner references 35 U.S.C. § 112(f) but this is not the applicable statute. *See* Petition, 37. The Challenged Patent is a continuation of an application filed before

In another jumbled argument, Petitioner contends that (1) the “‘Controller’ is...a nonce word” or, alternatively, (2) “recites a function but not sufficient structure for performing the function,” and, thus, § 112, ¶6 applies. Petition, 37. This argument has no merit. The Board and the Court have already rejected both arguments. *See* Ex. 2039, 59-61, 59 (“we agree with Patent Owner that the term ‘controller’...refers to a known class of structures”); Ex. 2032, 22:14-30:5.

The Federal Circuit and numerous other courts have consistently held that (1) “controller” is not a nonce term, and (2) it has a sufficiently definite structure. *See, e.g., Telcordia Techs., Inc. v. Cisco Sys., Inc.*, 612 F.3d 1365, 1377 (Fed. Cir. 2010) (“The record shows that an ordinary artisan would have recognized the controller as an electronic device with a known structure.”); *Via Vadis, LLC v. Buffalo Americas, Inc.*, 2016 WL 5239626, at *12 (W.D. Tex. Sept. 20, 2016), *rev’d in part on other grounds* *Vadis, LLC v. Blizzard Ent., Inc.*, 815 F. App’x 539 (Fed. Cir. 2020) (“Defendants are unable to overcome the presumption, arguing only that the word ‘controller’ is a nonce word, and that one of ordinary skill in the art would not have understood the term to imply or require any structure. The court disagrees.”); *Sysmex Corp. v. Beckman Coulter, Inc.*, 2021 WL 1259710, at *4 (D. Del. Apr. 6,

September 16, 2012, and, therefore, contrary to Petitioner’s argument, the applicable statute is pre-AIA 35 U.S.C. § 112, ¶6.

2021) (collecting four additional cases where courts found § 112, ¶ 6 does not apply to “controller”); *Capella Photonics, Inc. v. Fujitsu Network Commc’ns, Inc.*, 2021 WL 465430, at *23 (E.D. Tex. Feb. 9, 2021) (“[T]he Court understands that ‘control unit’ refers to a well-known class of structures also known as controllers.”).

The Challenged Patent’s specification confirms that the claimed “controller” is an electronic device with a known structure. The “Background” portion of the specification repeatedly refers to controllers as known devices that face specific challenges or perform specific tasks. Ex. 1001, 2:52-55 (“Sometimes, a plurality of files has to be combined and/or modified, which poses an enormous challenge for the memory *controller* device of a non-volatile memory system.”); 2:56-59 (“the vast majority of NAND flash in enterprise servers utilizes a SLC architecture, which further comprises a NAND flash *controller* and a flash translation layer (FTL)”); 3:21-27 (“wear leveling algorithms are used with the flash devices (as firmware commonly known as FTL or managed by a *controller*)”); 3:28-31 (“the *controller* maintains a lookup table”); 3:31-44 (“the *controller*’s wear-leveling algorithm determines which physical block to use each time data is programmed,... Depending on the wear-leveling method used, the *controller* typically either writes to....”).

A POSA would have understood the “controller” to be a component that uses firmware to “control” or “manage” the data flow to and from the memory. Ex. 2001,

¶79; Ex. 2003, 39 (Fig. 2.33, including a functional representation of a “Memory Controller”), 40-43. For example, a POSA would have known that firmware implemented by the controller may manage tasks such as wear leveling, garbage collection, and bad block management. Ex. 2003, 40; *see also* Ex. 1001, 3:21-27 (also referring to wear leveling performed by “firmware commonly known as FTL or managed by a controller”).

An overwhelming wealth of technical dictionaries and well-known textbooks confirms that a “controller” had a known structure. Ex. 2015 (Microsoft Computer Dictionary), 128 (defining “controller” as “a device that other devices rely on for access to a computer subsystem”); Ex. 2003, 39-40; *Va. Innovation Scis., Inc. v. Amazon.com, Inc.*, 2019 WL 4259020, at *13-15 (E.D. Tex. Sept. 9, 2019) (collecting “technical dictionary definitions...reinforcing that the term ‘controller’ refers to a known class of structures”); *Barkan Wireless IP Holdings, L.P. v. Samsung Elecs. Co., Ltd.*, 2019 WL 497902, at *21-23 (E.D. Tex. Feb. 7, 2019) (same); *see also Via Vadis*, 2016 WL 5239626, at *12 (“[D]ictionary definitions of ‘controller’ proffered by Plaintiffs show that one of ordinary skill in the art would understand this term to have an ordinary meaning.”).

In contrast, Petitioner provides no support for its argument that “controller” is a nonce term, or a term reciting a function but not sufficient structure for performing the function. Petition, 36-38. For example, Petitioner cites the *Rain Computing*

case, but this case involves an unrelated term and fails to counter the clear precedent establishing “controller” is not a nonce term. *See Rain Computing, Inc. v. Samsung Elecs. Am., Inc.*, 989 F.3d 1002, 1006 (Fed. Cir. 2021) (considering the term “user identification module,” where “module”—unlike “controller”—was a “well-known nonce word”). Further, Petitioner repeats the claim language but does not even attempt to adequately explain which claimed functionality a known controller could not have performed. *See Sound View Innovations, LLC v. Facebook, Inc.*, 2017 WL 2221177, at *11 (D. Del. May 19, 2017) (“Defendant also asserts that ‘an off-the-shelf server cannot perform the detailed operations in the claim without additional programming.’ I reject this argument. Defendant fails to cite a single source to support it.”).

Thus, § 112, ¶6 does not apply, and it is unnecessary to consider whether the specification provides sufficient structure. *See Virginia Innovation Scis., Inc. v. Amazon.com, Inc.*, 2019 WL 4259020, at *8 (E.D. Tex. Sept. 9, 2019) (explaining that “whether the patents disclose sufficient corresponding structure...is a distinct inquiry that arises **only if** 35 U.S.C. § 112, ¶ 6 is found to apply”).

Still, even if § 112, ¶6 applies, it is clear from the Challenged Patent’s specification that the “controller” provides a clear structure for performing the claimed functions. “[T]he specification need only disclose enough to permit one of ordinary skill in the art to perceive the bounds of the invention.” *Intelligent*

Automation Design, LLC v. Zimmer Biomet CMF & Thoracic, LLC, 799 F. App’x 847, 852 (Fed. Cir. 2020) (cleaned up). As Dr. Khatri explains, the “specification teaches a series of instructions for the computer to follow...that reasonably bounds how the claimed invention might [perform the functionality].” *Id.*; *see also* Ex. 2001, ¶¶83-84. A POSA would have understood that the specification discloses algorithms for the claimed functionality. Ex. 2001, ¶¶83-84; *see also Typhoon Touch Techs., Inc. v. Dell, Inc.*, 659 F.3d 1376, 1385 (Fed. Cir. 2011) (“[T]he patent need only disclose sufficient structure for a [POSA] to provide an operative software program for the specified function.”). For example, Figures 2A-2B describe the data integrity test, including mapping and remapping addresses, and Figures 3A-3B provide a “flow chart illustrating” the invention. Ex. 1001, Figs. 2A-2B, 3A-3B, 5:3-8; *see* Ex. 2001, ¶83; *see also StrikeForce Techs. Inc. v. PhoneFactor Inc.*, 2015 WL 5708577, at *1 (D. Del. Sept. 29, 2015) (“I agree...that the flow charts and/or specification prose disclose algorithms for the various means-plus-function terms.”).

4. 2.D: Claims Reciting Method Steps In The Context Of A System

Both the Board and District Court have already rejected Petitioner’s unclear assertion that the Challenged Claims are indefinite as reciting method steps in the context of a claim to a system. Ex. 2039, 61-62; Ex. 2032, 37:1-42:1. Those previous decisions should be maintained, because the Challenged Claims do not improperly mix apparatus elements and method steps, contrary to Petitioner’s

contention (Petition, 38-40). Rather, they explain how the controller is configured to operate in the claimed system. Ex. 2001, ¶¶85-87. This is exactly what the case law allows and is entirely consistent with *IPXL*. Here, the functional language in the claims describes how the controller is *configured* to operate. For example, claim 1 reads in part:

1. A system for storing data comprising:...

a controller coupled to the at least one MLC non-volatile memory element and the at least one SLC non-volatile memory element wherein the **controller is adapted to:...**

Ex. 1001, claim 1 (excerpts).

This claim simply describes the controller and how it is configured to operate within the system. In *IPXL*, the invalid claim recited a “*system of claim 2...wherein...the user uses the input means*” to change information. *IPXL Holdings, L.L.C. v. Amazon.com, Inc.*, 430 F.3d 1377, 1384 (Fed. Cir. 2005) (emphasis in original). ***Here, in contrast, there is nothing in any of the controller claims reciting a user action.*** A POSA would have understood that each verb simply describes how the *controller* is configured to operate, rather than referring to any user action. Ex. 2001, ¶86. Unlike *IPXL*, the Challenged Claims never refer to “the user.” “Though [the claim] includes active verbs—presents, receives, and generates—these verbs represent permissible functional language used to describe capabilities of the [claim elements].” *Mastermine Software v. Microsoft Corp.*, 874 F.3d 1307, 1315-16 (Fed.

Cir. 2017) (collecting cases); *see also Mastermine Software, Inc. v. Microsoft Corp.*, 874 F.3d 1307, 1316 (Fed. Cir. 2017) (explaining that it is permissible to “claim the system’s capability to receive and respond to user selection”); Ex. 2039, 62 (the Board found that, “[i]n contrast to claim 25 in *IPXL*, the ’546 patent’s claims[—where the ’546 patent is a parent of the Challenged Patent here—]do not require any user action”).

5. 2.E: On What And When The Data Integrity Test Is To Be Performed

There is nothing ambiguous about this term, and a POSA would have readily understood its scope. Ex. 2001, ¶88. The Board already considered—and rejected—Petitioner’s argument in denying institution of PGR2024-00047. Ex. 2039, 62; *see* Petition, 40-42.

Claim 1 makes clear that the “data integrity test” is “performed at the controller based upon received data retained at the controller.” (Limitation [1e]). And the specification provides further insight “on what and when” a “data integrity test” may be performed. Petition, 40. For example, the data integrity test may be performed “after every write.” Ex. 1001, 5:23-24 (“The controller conducts a data integrity test after every write.”); *see also id.*, 4:38-40. Alternately, the data integrity test may be performed only when “time permits.” *Id.*, 6:3-6 (“After each write to an address within a particular address range, the device controller 14 will—as time permits—perform a read on the address range to ensure the integrity of the written

data.”). Thus, a POSA would have readily understood that data integrity tests performed after every Write access operation, or only after some Write access operations, to be within the scope of claim 1. Ex. 2001, ¶88.

Petitioner suggests that claim 1 does not provide sufficient guidance regarding “on what and when” the data integrity test must be performed and, accordingly, is somehow indefinite. Petition, 40-42. Petitioner suggests that the claim must explicitly state whether the data integrity test is performed after “every write” or “if time permits.” *Id.*, 41 (emphasis omitted). But Petitioner cites no cases to support this contrived theory. Instead, again, Petitioner confuses breadth with indefiniteness. *See Niazi*, 30 F.4th at 1347. The claims need only inform the scope of the invention with “reasonable certainty,” which claim 1 does here. *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014); *see also* Ex. 2039, 62 (“As Patent Owner contends, Petitioner’s arguments concerning uncertainty confuse ‘breadth with indefiniteness.’”).

VII. GROUND 3: 35 U.S.C. § 103

A. Overview Of The Cited Prior Art

Petitioner asserts that the claims are obvious over Gavens (U.S. Patent No. 8,634,240) (Ex. 1045) in view of the knowledge of a POSA. Petition, 42-67.

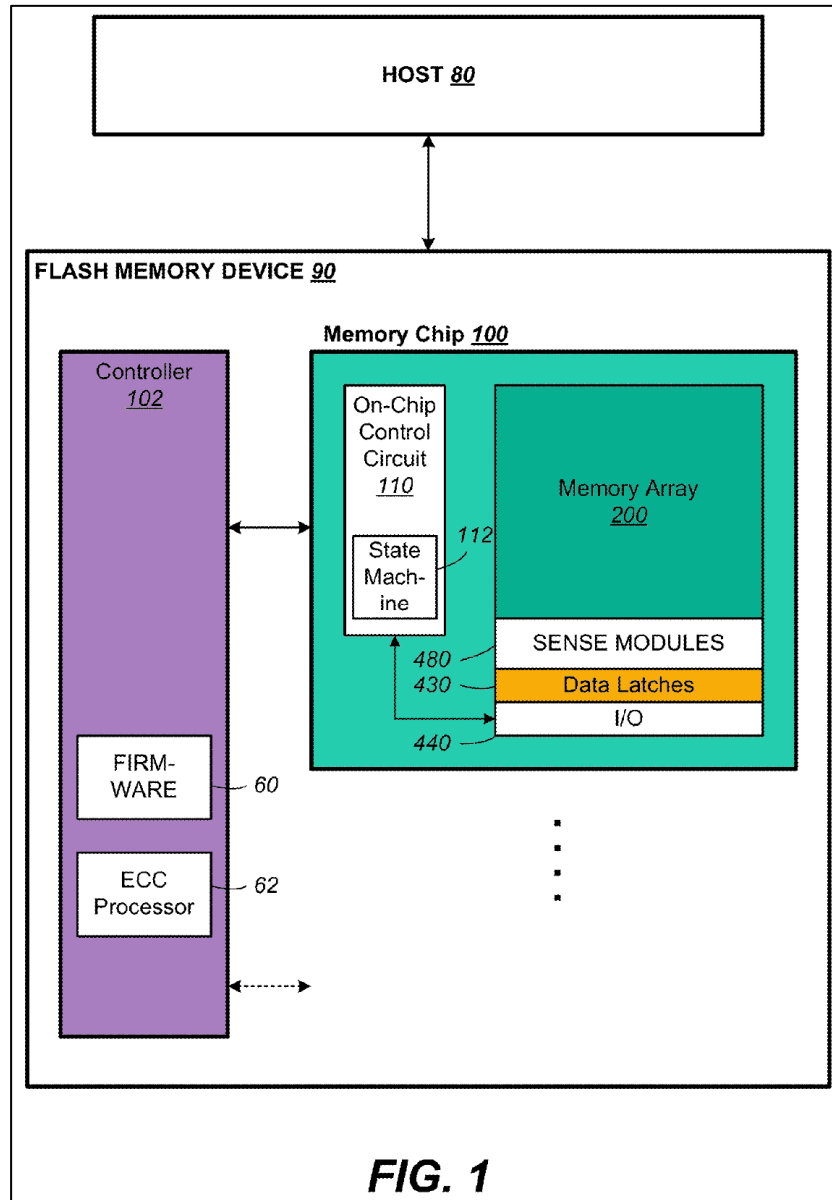
In its single-reference § 103 ground, Petitioner also cites a range of tertiary references that it contends relate to the claimed random access volatile memory.

Petitioner does not explicitly combine these references with Gavens, but instead cites them in connection with its conclusory expert testimony regarding modifications a POSA allegedly would have made. These tertiary references include documents that Gavens purportedly incorporates by reference, such as U.S. Patent Appl. Pub. No. 2011/0153912 (“Gorobets 912”) (Ex. 1049), U.S. Patent Appl. Pub. No. 2010/0172180 (“Paley”) (Ex. 1051), and PCT Pub. No. WO 03/027828 (“Gorobets PCT”) (Ex. 1058).

1. Gavens (Ex. 1045)

Gavens discusses a problem that occurs with flash memory (a type of non-volatile memory)—as it ages, its error rate increases. Ex. 1045, 3:56-4:5; Ex. 2001, ¶¶91-106. To ensure data integrity in such situations, the memory uses a resource-intensive error correction code (ECC) to correct errors. Ex. 1045, 3:56-4:5. Gavens explains that this poses a problem because, “for most of the life time of the device [(before the device ages and the error rate increases)], the ECC is only marginally utilized, resulting in its large over-heads being wasted[.]” *Id.*, 3:67-4:2.

To perform ECC, Gavens describes a flash memory device including **controller 102** (highlighted purple) and **memory chip 100** (highlighted teal).



Ex. 1045, Fig. 1 (annotated), 8:13-39; Ex. 2001, ¶92. The memory chip includes an *on-chip* control circuit 110 with state machine 112. The memory chip also includes input/output (I/O) circuits, **data latches 430**, sense modules, and a non-volatile flash **memory array 200**. *Id.*, 4:13-16, 8:13-39. In contrast to the Challenged Patent, there is no mention of RAM in the memory chip 100, controller chip 102, or anywhere else in the flash memory device 90. In fact, the only time that Gavens

mentions RAM is in the “Background of the Invention,” when Gavens is *distinguishing* its purported invention from RAM. *Id.*, 1:26-29 (“Unlike RAM...that is also solid-state memory, flash memory is non-volatile”).

Before writing a page of data to the memory array, “ECC is computed for the data page” by the ECC Processor 62. *Id.*, 12:51-63. The page of write-data then consists of two portions: the “user data” and the computed ECC. *Id.*

When a page is read from the memory array, it is “shifted out” (or “toggled”) to **controller 102**. There, first an error detection code (EDC) “indicates the existence of any errors in the read data page[,]” and if there are errors, “the data page’s existing ECC is compared to a second version of the ECC computed on the read data [(the “user data”)].” *Id.*, 12:64-13:8.⁸

Gavens teaches that using ECC in this way—where the ECC must be programmed to correct the “worst-case” number of bits—is burdensome in terms of processing time as well as chip area on the controller:

Using ECC to correct a worst-case number of error bits will consume a great amount of processing time. The more bits it has to correct, the

⁸ This EDC/ECC relies entirely on data that is read from the non-volatile memory array. Thus, unlike the claimed invention of the Challenged Patent, this EDC/ECC does not involve a comparison of data read from the non-volatile memory with data *retained* in volatile memory (such as RAM). Ex. 2001, ¶94 n.1.

more computational time is required. The memory performance will be degraded. Additional dedicated hardware may be implemented to perform the ECC in a reasonable amount of time. *Such dedicated hardware can take up a considerable amount of space on the controller ASIC chip.* Moreover, for most of the life time of the device, *the ECC is only marginally utilized*, resulting in its large over-heads being wasted and realizing no real benefits.

Ex. 1045, 3:48-4:2 (emphasis added); *see also id.*, 5:15-30, 13:9-16.

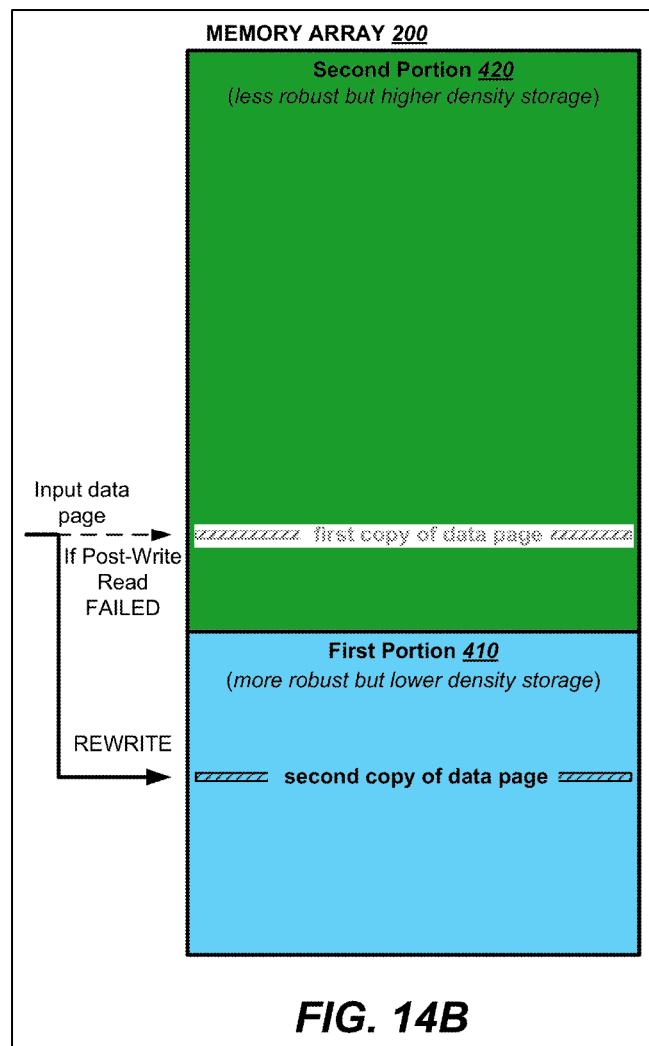
To address this problem, Gavens discloses the embodiments of Figures 14A-14B and 16A-16C, which describe the **memory array 200** of Figure 1 in greater detail. *See id.*, 8:13-15 (“Fig. 1 illustrates...a memory device in which the features of the present invention are embodied.”).

These embodiments rely on a memory with a first SLC portion (which is less error prone) and a second MLC portion (which is more error prone). Gavens explains that optimizing where data is stored (in the SLC or MLC portion) based on the expected number of error bits “allows a smaller and more efficient...ECC...to be designed for correcting a smaller number of error bits, thereby improving the performance and reducing the cost of the memory.” *Id.*, 4:28-31.

a. First Embodiment (Figs. 14A-14B)

Gavens describes a first embodiment with respect to Figures 14A-14B. *See* 1045, 17:24-26 (referring to “the embodiment described in Fig. 14A and Fig. 14B”). For this embodiment, as shown in Figure 14B below, Gavens describes a memory

array that relies on a “**lower density**” first portion 410 (*i.e.*, SLC) and a “**higher density**” second portion 420 (*i.e.*, MLC). *Id.*, 4:13-19, 16:14-28; Ex. 2001, ¶¶98-100. The MLC portion is more error prone than the SLC portion but can be used “for efficient storage.” Ex. 1045, 4:16-19, 16:41-45. Gavens explains that data can first be written to the MLC portion, then “read back in a post-write read operation to check for excessive error bits.” *Id.*, 4:19-21; *see also id.*, Fig. 14A, 16:41-56.



Id., Fig. 14B (annotated); Ex. 2001, ¶98.

Gavens explains that the post-write error detection process can check for errors in one of two ways: “either by comparison with the original copy which may be cached or by checking the EDC portion of the ECC.” Ex. 1045, 16:46-49. While Gavens at 16:46-49 does not specify how the “original copy” is cached, Gavens later explains that “[i]n the preferred embodiment” (where the antecedent basis of “the” refers to the immediately preceding “preferred embodiment” of Figures 14A-14B) the “data is cached in [a] first section of the first portion” of non-volatile memory (*i.e.*, a section of the non-volatile SLC memory). *Id.*, 18:1-8; *see also id.*, 16:14-16. Gavens does not describe caching the “original copy” in volatile memory.

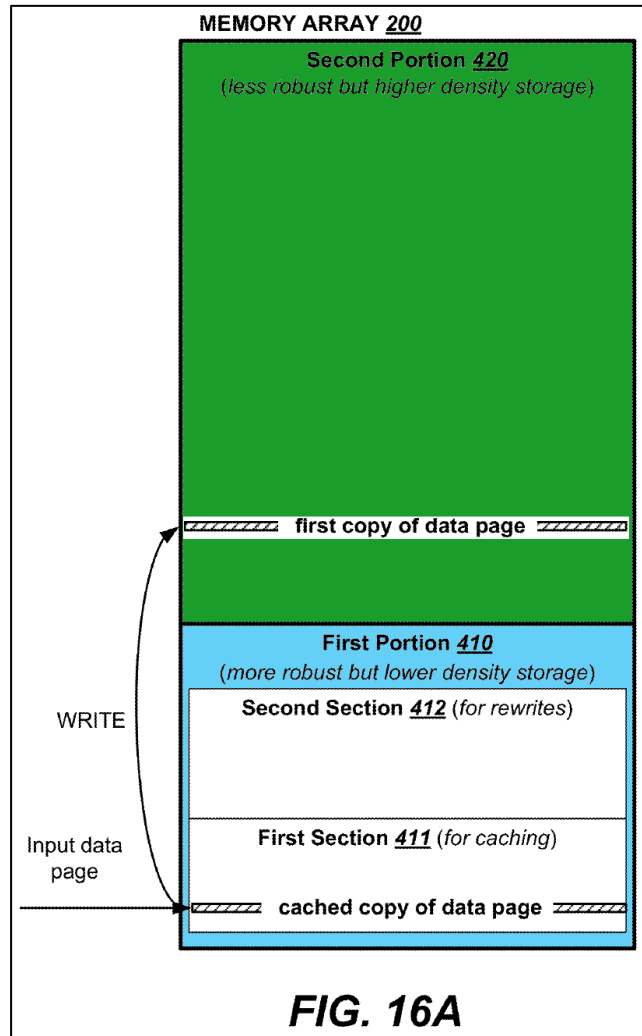
Once the error rate is checked, the device determines whether to keep the data in MLC or instead move it to SLC. If the stored data has an acceptable number of error bits, then the stored copy remains in the MLC and becomes the “valid” copy. *Id.*, 16:41-56. “If the error bits exceeded a predetermined amount, the data is” instead stored in “the less error-prone” SLC portion. *Id.*, 4:21-22; *see also id.*, Fig., 14B, 17:7-20, 18:1-8. To rewrite the data, Gavens teaches that the memory may rely on the “cached” copy of the original data or “correcting the errors bits [from the copy stored in the MLC] with the ECC.” *Id.*, 17:12-14; *see also id.*, 18:1-8.

b. Second Embodiment (Figs. 16A-16C)

Figures 16A-16C depict a second embodiment of Gavens. *See* Ex. 1045, 7:26-28 (referring to “the embodiment described in Fig. 16A to Fig. 16C”); Ex. 2001,

¶¶101-106. For this embodiment, Gavens teaches splitting the first portion (*i.e.*, the SLC portion) into a first section “for caching” and a second section “for rewrit[ing]” data. *Id.*, Fig. 16A, 18:9-23. Gavens emphasizes that this approach limits reliance on the controller 102. *See* Ex. 1045, Fig. 1, 3:60-4:2, 18:36-39.

As shown in Figure 16A below, Gavens explains that the lower density First Portion (*i.e.*, SLC) includes two sections—a “first section” 411 for caching write data, and a “second section” 412 for rewriting data from the higher density Second Portion (*i.e.*, MLC). *Id.*, 18:9-23.



Id., Fig. 16A (annotated); Ex. 2001, ¶103.

Incoming data is written both to the MLC and as a “cached copy” to the first section of the SLC. Ex. 1045, 18:21-29. Detecting the number of error bits is then “accomplished by comparison [of the data written to MLC] with the cached copy.” *Id.*, 18:40-44.

Gavens specifies that this comparison may be performed at the **data latches 430** (highlighted orange in Figure 1 above) on the **flash memory chip 100**. *Id.*,

16:46-49, 18:30-39. Gavens strongly recommends using **data latches 430** because it eliminates the need to “toggle” the data out to the **controller chip 102**, as would be required if using ECC Processor 62 for EDC-based error detection. *Id.*, 18:36-39; *see also id.*, 3:60-4:2. Gavens explains that “much time can be saved” by eliminating the need to toggle data to the controller chip, thus increasing the performance of the device. *Id.*

If there is an acceptable number of error bits, the MLC portion becomes the “valid” copy. *Id.*, 18:45-52, Fig. 16B. If there is an unacceptable number of error bits, the “cached copy” in the “first section” of the SLC is rewritten to the “second section” of the SLC. *Id.*, 18:53-67, Fig. 16C.

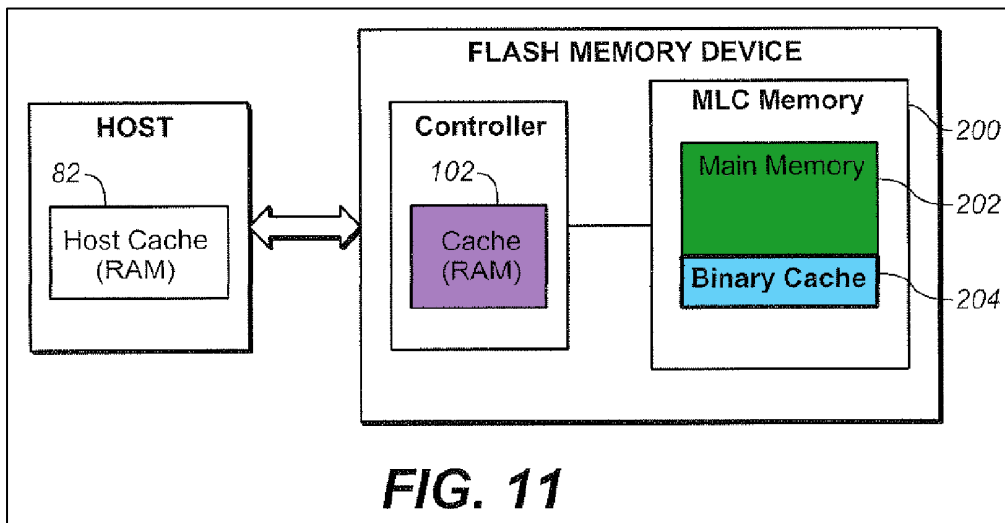
By caching the data in the non-volatile memory array and using the latches to compare the data, Gavens lowers overhead and reduces reliance on the controller chip. *Id.*, 18:40-67.

2. Gorobets 912 (Ex. 1049)

Like Gavens, Gorobets 912 includes a non-volatile memory with SLC and MLC portions. Gorobets 912 describes using the non-volatile SLC as a cache to aggregate data before it is written to MLC. *See* Ex. 2001, ¶¶107-109.

As shown in Figure 11 below, Gorobets 912 describes a flash memory device with an **SLC portion** and an **MLC portion**. Ex. 1049, ¶¶104, 109. The **SLC**

portion acts as “a cache for data to be written to the [MLC] memory.” *Id.*; see also *id.*, ¶108 (explaining the utility of the non-volatile SLC cache).



Ex. 1049, Fig. 11 (annotated); Ex. 2001, ¶108.

The flash memory device also relies on a controller, including a generic volatile **RAM-based cache**. Gorobets 912 explains that this **cache** is “high-speed storage for *temporarily* storing data being passed between a high-speed and slower-speed component of the system.” Ex. 1049, ¶109 (emphasis added); see also *id.*, ¶130.

Gorobets 912 does not describe a volatile cache that stores data after it is written to the non-volatile memory—data is not *retained* in volatile memory. See *id.*, ¶¶24, 109, 130, 144.

3. Paley (Ex. 1051)

Paley is closely related to Gorobets 912⁹ and, as shown in Figure 15 below, also describes a non-volatile memory with **SLC** and **MLC** portions, where the **SLC portion** is used as a cache to aggregate data before it is written to **MLC**. Ex. 1049, ¶192 (“A second portion forms a cache for data to be written to the main memory.”), ¶193 (“individual cells in the cache portion are each configured to store one bit of

⁹ Paley and Gorobets 912 share multiple named inventors. *Compare* Ex. 1051, Cover (listing inventors, including “Sergey...Gorobets”) *with* Ex. 1049, Cover (listing inventors, including “Sergey...Gorobets”). Additionally, much of these references’ disclosure is substantially the same. *Compare* Ex. 1051, Figure 15 *with* Ex. 1049, Figure 11 (other than differences in numbering, these figures are the same).

data while the cells in the main memory portion each stores more than one bit of data”); *see* Ex. 2001, ¶¶107-109.

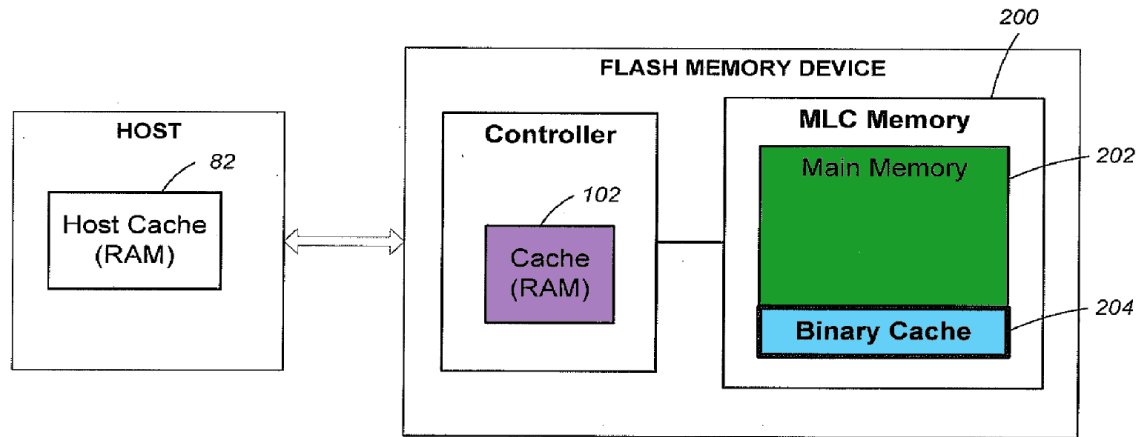


FIG. 15

Ex. 1051, Fig. 15 (annotated); *see also* Ex.1049, Fig. 11 (Gorobets 912’s Figure 11 is substantially the same as Paley’s Figure 15); Ex. 2001, ¶108.

Paley further describes where the controller includes a volatile **RAM-based cache**. Paley—just like Gorobets 912—explains that this **cache** is “high-speed storage for *temporarily* storing data being passed between a high-speed and slower-speed component of the system.” Ex. 1051, ¶197 (emphasis added); Ex. 1049, ¶109 (same). Thus, both Gorobets 912 and Paley describe a memory system including a **RAM-based cache** that only *temporarily* stores data being passed between components.

4. Gorobets PCT (Ex. 1058)

Petitioner also alleges that Gorobets 912 and Paley incorporate by reference Gorobets PCT. Petition, 49-50. This is not true. Gorobets 912 and Paley merely state that “WO 03/027828 [(Gorobets PCT)] disclose[s] a memory system...” Ex. 1049, ¶19; Ex. 1051, ¶23 (same). These references do not state, with the required specificity, that Gorobets PCT is incorporated by reference—“To incorporate matter by reference, a host document must contain language ‘clearly identifying the subject matter which is incorporated and where it is to be found’; a ‘mere *reference* to another application, or patent, or publication is not an *incorporation* of anything therein....” *Callaway Golf Co. v. Acushnet Co.*, 576 F.3d 1331, 1346 (Fed. Cir. 2009) (quoting *In re De Seversky*, 474 F.2d 671, 674 (CCPA 1973)) (emphasis in original).

Notwithstanding, Gorobets PCT—just like Gorobets 912 and Paley—describes a volatile RAM-based buffer used to *temporarily* store data. Ex. 1058, 11:10-13 (describing “SRAM block 30 [that] stores all *temporary* information within the controller” (emphasis added)). Data “being transferred” from system components to a flash memory “is *buffered* in the SRAM 30.” *Id.*, 11:21-23 (emphasis added); *see also* Ex. 2001, ¶113.

B. The Petition Does Not Establish That Independent Claims 1 Or 18 Would Have Been Obvious Over Gavens

As discussed below, the Petition’s analysis is fatally flawed regarding multiple aspects of Claims 1 and 18 of the Challenged Patent, and those flaws propagate to each challenged dependent claim. Each of the below flaws warrants denial of institution.

1. [1b] and [18b]: “SLC non-volatile memory element”

Limitation [1b] requires, in part, “at least one SLC non-volatile memory element.” Petitioner’s obviousness analysis is deficient and fails to show why the modification to Gavens proposed by Petitioner would have been obvious to a POSA. *See* Petition, 46-47; Ex. 2001, ¶¶115-123.

For this limitation, Petitioner applies its own construction for “SLC non-volatile memory”: “non-volatile memory cells arranged with circuitry *incapable* of storing multiple logical pages in a single physical page of cells.” Petition, 27 (emphasis added). Applying this construction, Petitioner suggests that limitation [1b] requires a “*native* SLC flash chip .” Petition, 46 (“Gavens does not disclose the use of a ‘native’ SLC flash memory module or chip...because the Gavens memory module or chip 100 is capable, with its circuitry, of programming more than one logical page into a physical page[.]”). Petitioner contends that Gavens’s use of “pseudo-SLC” flash—that is, flash that is capable of operating in an MLC mode *or* SLC mode—is insufficient under its construction. *Id.* While PO disagrees with this

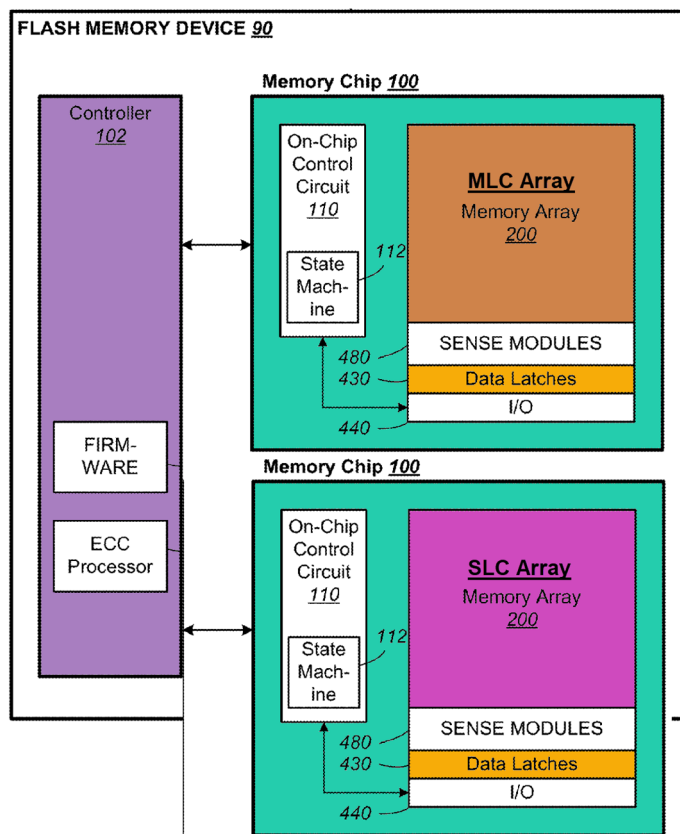
construction, even if it is applied—where Petitioner presents its ground under its own construction for “SLC non-volatile memory”—Petitioner’s ground fails.

First, Petitioner fails to adequately explain why it would have been obvious to substitute one of Gavens’s SLC memory *portions* (operating in pseudo-SLC) with a “‘native’ SLC flash chip.” Petition, 46-47. Instead, Petitioner offers only a conclusory assertion, citing paragraph 102 of its expert report. *See* Petition, 31 (asserting, without explanation, that “[a]s explained in [Sechen ¶ 102], it would be obvious to a POSITA to substitute a ‘real’ SLC flash memory module for the Gavens first portion, with a similar, if not wider margin of error than binary mode operation of MLC (see [Sechen ¶ 49]) and achieve similar, if not better results with the same NAND flash controller...with great expectation of success.” (emphasis added)). But “arguments may not be incorporated from an expert’s declaration wholesale into the Petition,” and Petitioner provides no rationale for its obviousness theory in its ground for limitation [1b] itself. *Acon Inc. v. Amo Dev., LLC*, IPR2021-00858, 2022 WL 17585777, at *16 (P.T.A.B. Dec. 12, 2022); *see also* 37 C.F.R. § 42.6(a)(3).

For example, Petitioner fails to explain why a POSA would have been motivated to make the suggested modification. Simply alleging that this “substitut[ion]” would yield “similar” results does not answer why a POSA would have been motivated to pursue it. Petition, 47; *see In re Stepan Co.*, 868 F.3d 1342, 1346 n.1 (Fed. Cir. 2017) (“there must be a motivation to make the combination”).

“[O]bviousness grounds 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.” *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006); *see also* 37 C.F.R. § 42.22(a)(2) (requiring an “explanation of the significance of the evidence”). Thus, Petitioner fails to demonstrate why it would have been obvious to modify Gavens to use a “native SLC flash chip.”

Second, contrary to Petitioner’s conclusory assertion, and as illustrated in the annotated version of Gavens’s Figure 1 below, a POSA would not have been motivated to substitute one of Gavens’s SLC portions—which, along with the MLC portion, make up the memory array of one of Gavens’s memory chips—with an entirely separate “**native SLC flash chip**.” *See* Ex. 2001, ¶119; *see also* Section VII.A.1 (describing Gavens, including its “memory chip” comprising the “memory array” of Figure 1, where each memory array includes a “first portion” (SLC) and “second portion” (MLC) as shown in Figures 14 and 16). As shown in the modified version of Gavens’s Figure 1 below—altered to include the separate “SLC Array”—Petitioner’s proposed modification would result in **one memory chip consisting entirely of MLC**, and **another memory chip consisting entirely of SLC**.



Ex. 1045, Figure 1 (altered to include a second memory chip comprising the “SLC array”).

Gavens, however, explains that each of its memory chips are already capable of operating with one portion as MLC, and another portion as SLC (albeit, not “native” SLC). *See* Ex. 1045, Abstract, 4:13-31, 6:17-25, 15:62-16:12, 16:17-25, Fig. 14, Fig. 16. That is the essence of Gavens’s invention—*in a single memory chip*, taking advantage of the efficiencies of MLC storage, by relying on a “less error-prone” SLC *portion* when an MLC portion has too many errors. *See id.*, 4:13-31, 4:18-22 (“Data is written to the second portion for efficient storage. Afterwards, the data is readback in a post-write read operation to check for excessive error bits.

If the error bits exceeded a predetermined amount, the data is rewritten or kept at the less error-prone first portion.”), 15:66-16:13, 16:41-45.

Additionally, Gavens explains that the SLC portion is also used to cache data, as described in its first and second embodiments (*see supra* Section VII.A.1), to avoid needing to “toggle[]” data out to **controller 102** when checking data written to the MLC portion for errors. *See* Ex. 1045, 18:21-39, 18:36-39. This is because “**data latches**” within the memory chip (which includes *both the MLC and SLC portions*) enable data comparison without relying on external controller 102. *See id.*, 18:30-39, Fig. 1 (showing “Data Latches” 430 as part of the Memory Chip, separate from the Controller 102). Gavens explains: “By not making the comparison at the controller, the data does not have to be toggled out to the controller, [and] much time can be saved. Fig. 1 shows the data latches 430, *which is on-chip*, for the data comparison to take place.” Ex. 1045, 18:36-39 (emphasis added).

Replacing the SLC portion of the *same* memory chip with a *separate* SLC chip connected to the controller 102 would negate this benefit, as data cached in a **separate SLC chip** would need to be toggled out to the **controller** for comparison with data stored in the **MLC chip**. Ex. 2001, ¶122. Therefore, Gavens teaches away from Petitioner’s proposed modification, and a POSA would not have been motivated to substitute an SLC portion of a memory chip with a separate “‘native’ SLC flash chip,” as this modification would have undermined Gavens’s intended

advantage—efficiently utilizing higher-density MLC memory. *See* Ex. 2001, ¶122. Petitioner’s proposed modification “would require a substantial reconstruction and redesign of the elements shown in [Gavens] as well as a change in the basic principles under which [Gavens] was designed to operate” and cannot be correct. *Samsung Elecs. Co., Ltd. v. Mojo Mobility Inc.*, IPR2023-01099, 2024 WL 457731, at *10 (P.T.A.B. Feb. 6, 2024) (denying institution) (quoting *Application of Ratti*, 270 F.2d 810, 813 (C.C.P.A. 1959)).

Furthermore, Limitations [18b] and [1b] are similar, and Petitioner relies on the same ground for both limitations. *See* Petition, 64. Accordingly, the Petition’s analysis of limitation [18b] fails for the same reasons as those given above for limitation [1b].

2. [1d] and [18d]: Address Map

For limitation [1d], Petitioner fails to show how Gavens—alone or in combination with the numerous cited tertiary references—discloses or suggests what is claimed. Petition, 48-51. In its ground, Petitioner ambiguously references teachings from: Gavens; allegedly incorporated references such as Gorobets 912, Paley, and Gorobets PCT¹⁰; and the unrelated reference Goodson (Ex. 1070).

¹⁰ As explained above in Section VII.A.4, Petitioner is incorrect in its assertion that Gorobets 912 and Paley incorporate by reference Gorobets PCT.

Petition, 52-54. Petitioner, however, does not clearly map these references' cited teachings to limitation [1d], including, for example, to indicate *which* disclosure it is relying on for each of the claimed features of the “address map.” *See* Petition, 52-54. Further, Petitioner also does not explain how the distinct disclosure from the tertiary references would/could have been combined with the previously cited disclosure of Gavens. Petitioner cannot pick and choose between distinct elements from different disclosures. *See Captioncall, LLC v. Ultratec, Inc.*, No. IPR2020-01215, 2021 WL 278381, at *7 (P.T.A.B. Jan. 27, 2021) (denying institution when petitioner relied on “distinct” embodiments from a reference and another reference that was incorporated by reference, explaining that “[a]nticipation requires all elements arranged as in the claim without the need to pick and choose elements from different disclosures”).

Finally, while Petitioner cites multiple distinct disclosures in its ground, the Petition does not state that it is relying on an obviousness analysis—Petitioner relies only on what “Gavens...discloses” and what “Gavens and its incorporated references disclose.” Petition, 51; *see also In re Stepan Co.*, 868 F.3d 1342, 1346 n.1 (Fed. Cir. 2017) (“Whether a rejection is based on combining disclosures from multiple references, combining multiple embodiments from a single reference, or selecting from large lists of elements in a single reference, there must be a motivation to make the combination and a reasonable expectation that such a combination

would be successful, otherwise a skilled artisan would not arrive at the claimed combination.”).

Thus, Petitioner has not demonstrated with particularity how limitation [1d] maps to disclosure in the prior art, as required by 35 U.S.C. § 312(a)(3). The Petition lacks an “explanation of the significance of the evidence.” 37 C.F.R. § 42.22(a)(2). Petitioner’s scattershot approach—citing disparate teachings from tertiary references without clearly identifying how they are relying on them—must be rejected. *See Masimo Corp. v. Apple Inc.*, IPR2023-00745, Paper 44 (final written decision), 36 (P.T.A.B. Oct. 15, 2024) (determining that the challenged claim is not unpatentable, explaining: “In essence, Petitioner invites the Board to piece together a single theory that meets all claim elements from these disparate theories that each require distinct structures....”) (citing *Amazon.com, Inc., et al. v. Avago Technologies General IP (Singapore) PTE. LTD.*, IPR2017-01112, Paper 11, at 13 (P.T.A.B. Aug. 23, 2017) (“Petitioner is inviting us to assume the role of archeologist of the record, which invitation we decline.”)); *Nearmap US, Inc. v. Eagle View Technologies, Inc.*, IPR2024-00716, Paper 9, 18 (P.T.A.B. Oct. 9, 2024) (denying institution of *inter partes* review, explaining that “[t]he question before us is not whether we could come up with a way to understand how the asserted prior art could render the challenge claims unpatentable, but rather did Petitioner provide the arguments and evidence to show that the challenged claims are unpatentable.”);

Netflix, Inc. v. DivX, LLC, 84 F.4th 1371, 1379 (Fed. Cir. 2023) (“While the Board should review a petition holistically, it is not obligated to cobble together assertions from different sections of a petition or citations of various exhibits in order to infer every possible permutation of a petitioner’s argument.” (cleaned up)).

Furthermore, limitations [18d] and [1d] are similar, and Petitioner relies on the same ground for both limitations. *See* Petition, 65. Accordingly, the Petition’s analysis of limitation [18d] fails for the same reasons as those given above for limitation [1d].

3. [1e] and [18e]: “data integrity test performed at the controller based upon received *data retained at the controller*”

For limitation [1e], Petitioner fails to demonstrate that Gavens discloses or suggests retaining data “at the controller” for a data integrity test. Petition, 52-54.

Here, Petitioner’s ground is again conclusory and inadequately explained. Notwithstanding, the Petition asserts that Gavens’s disclosure of a “‘post write read’...check for errors ‘by comparison with the original copy *which may be cached*’” (Ex. 1045, 16:46-49) would have been understood to refer to a RAM-based cache based on the teachings of Gorobets 912 and Paley. Petition, 52-53 (“The ‘original data’ temporarily stored (‘cached’ in RAM, *understood to be* RAVM)...” (emphasis added)). Petitioner then appears to suggest that this RAM-based cache corresponds to a “controller memory” for retaining data, as required by limitation

[1h] (“data retained *at the controller*”). Petition, 53 (“It would be obvious that the ‘stored data’ to be compared would be *written to the controller memory* (where the *Gavens/Gorobets cache* has space for at least two write units, nominally, ‘pages’) for comparison using logic on the controller chip...” (emphasis added)). While not clearly explained in the Petition itself, this may be because Gorobets 912 and Paley disclose RAM-based caches located in the controller. *See supra* Sections VII.A.2-VII.A.3 (summarizing relevant disclosure of Gorobets 912 and Paley); *see also* Ex. 1049, Fig. 11 (including RAM 102 in the controller); Ex. 1051, Fig. 15 (same).

A POSA would not have shared Petitioner’s understanding, as Gavens’s “cache[]” that retains the “original copy” (Ex. 1045, 16:46-49) refers to a *non-volatile cache* which is not in the controller, not a RAM-based cache in the controller like Gorobets 912 or Paley. And Petitioner fails to show that a POSA would have interpreted Gavens differently based on those allegedly incorporated references. *See* Ex. 2001, ¶¶124-141.

First, Petitioner cannot show that Gavens itself discloses a RAM-based cache in the controller which retains data for the post-write error check. *See* Ex. 1045, 16:46-49 (describing a post-write error detection process that can check for errors “by comparison with the original copy which may be cached”). Gavens instead explains that the “original copy” is cached in SLC non-volatile memory. *See supra* Section VII.A.1; *see also* Ex. 2001, ¶127.

As a preliminary matter, Gavens does not disclose or suggest that its system includes *any* RAM. *Contra* Petition, 52 (“In the Gavens NAND flash technology corpus, ‘using RAM in a write cache operating with flash memory ***has been disclosed***’...” (emphasis added)). Gavens explicitly distinguishes its teachings from systems that use RAM, explaining how “nonvolatile storage” is “***unlike*** RAM.” Ex. 1045, 1:26-29 (“Unlike RAM (random access memory) that is also solid-state memory, flash memory is non-volatile, and retaining its stored data even after power is turned off.”). This is consistent with Gavens’s description of its system in Figure 1, as shown below, which includes only the ***non-volatile*** “Memory array 200.” *See* Ex. 1045, 1:17-20 (“This application relates to the operation of re-programmable ***non-volatile memory systems*** such as semiconductor flash memory, and, more specifically, to handling and efficient managing of errors in memory operation.” (emphasis added)), 4:13-16 (“According to a general aspect of the invention,...a ***flash*** memory having an array of memory cells is configured with a first portion and a second portion.” (emphasis added)).

Therefore, “Gavens’s invention is directed specifically to non-volatile memory” and “a [POSA] would [not] have found Gavens to teach or suggest [a memory system] that includes RAM.” Ex. 2041, 42 (decision denying institution of PGR2024-00048); *see also* Ex. 2039, 76 (decision denying institution of PGR2024-00047, where the Board determined that “Gavens does not disclose ‘random access

volatile memory’ according to claim 1, e.g. in a ‘system’ with, among other things, a ‘controller’ and a ‘bank of nonvolatile memory’’”).

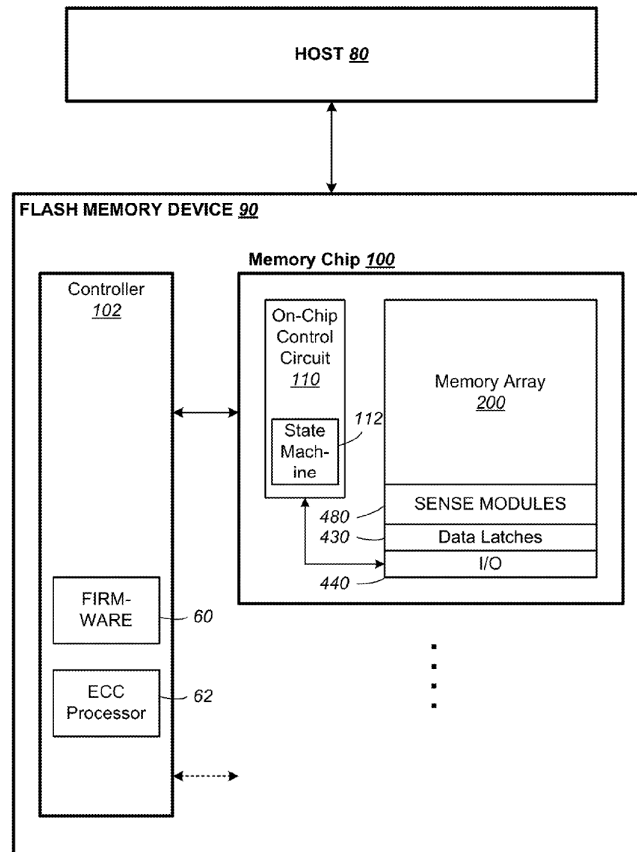


FIG. 1

Ex. 1045, Fig. 1.

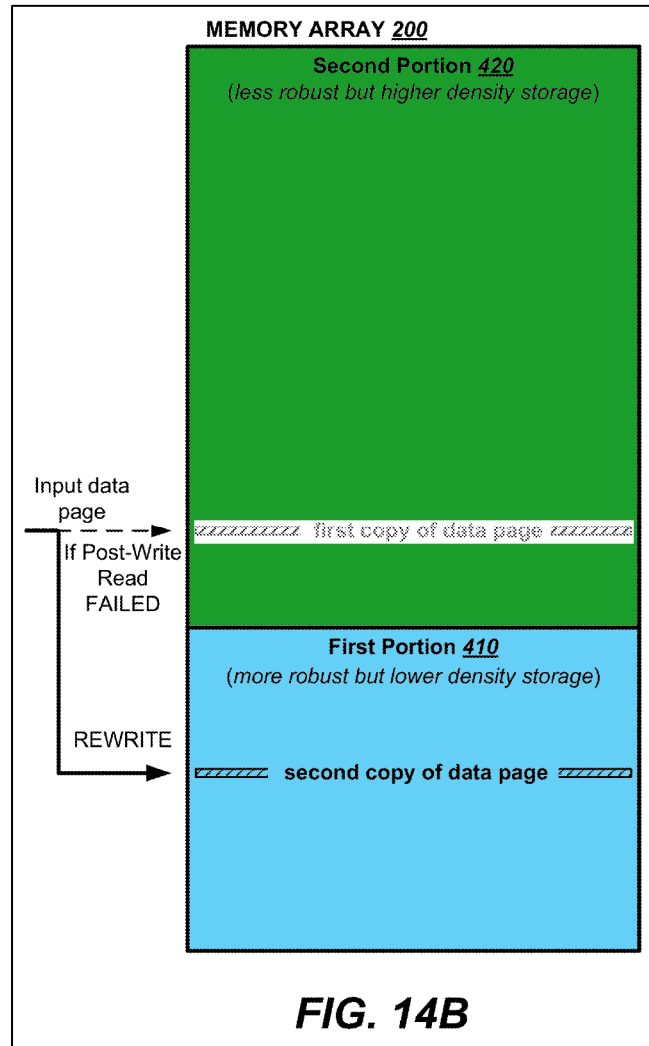
Potentially recognizing that Gavens itself does not disclose RAM, Petitioner ambiguously cites Gorobets 912 and Paley. *See* Petition, 52 (“In the Gavens NAND flash technology corpus, ‘using RAM in a write cache operating with flash memory has been disclosed’ (e.g., Gorobets ‘912 [Ex. 1049] ¶ [0024], Paley ‘180 [Ex. 1051] ¶ [0027]).”). Petitioner ignores, however, that the cited

embodiments from these references—each allegedly disclosing a RAM-based cache—are *distinct from the embodiments of Gavens*. Ex. 2001, ¶129. For one, as explained above, Gavens never discloses the use of random access volatile memory. And Petitioner provides no explanation how the distinct disclosure of the allegedly incorporated references (Gorobets 912 and Paley) applies to or can be combined with the previously cited disclosure of Gavens. Petitioner cannot pick and choose between distinct elements from different disclosures. *See Captioncall, LLC v. Ultratec, Inc.*, No. IPR2020-01215, 2021 WL 278381, at *7 (P.T.A.B. Jan. 27, 2021) (denying institution when petitioner relied on “distinct” embodiments from a reference and another reference that was incorporated by reference, explaining that “[a]nticipation requires all elements arranged as in the claim without the need to pick and choose elements from different disclosures”).¹¹

¹¹ Petitioner made a similar error in PGR2024-00048, which challenged U.S. Patent No. 11,854,612—a parent of the 370 Patent challenged in this Petition. There, Petitioner cited Gorobets 912’s teaching of RAM for a limitation related to RAM, and the Board found that, “As Patent owner correctly notes, ‘Petitioner...does not state whether Gorobets’s ‘RAM’ is being used to modify Gavens’s system, and why a POSA would have made any modification.’” Ex. 2041, 43.

Accordingly, Gavens does not disclose a system including *any* RAM, and Petitioner's citations to Gorobets 912 and Paley are inadequate.

Furthermore, Petitioner's apparent understanding of Gavens's "cache" is incorrect. In its ground, Petitioner cites teachings from the "first embodiment" of Gavens, including Figure 14; an overview of this embodiment is provided in Section VII.A.1.a above. *See* Petition, 52 (citing Gavens's Figures 14 and 15, and its disclosure at 16:13-17:42). With Figure 14B, as illustrated below, Gavens describes a memory including an **SLC portion** which is less error prone, and **MLC portion** which is more error prone but may be used "for [more] efficient storage." Ex. 1045, 4:16-19, 16:41-45; *see also id.*, 4:13-19, 16:14-28. Gavens explains that data written to the MLC portion can be "read back in a post-write read operation to check for excessive error bits." *Id.*, 4:19-21. Gavens at 16:46-49, as cited by Petitioner, refers to this post-write error check of data written to the MLC portion.



Ex. 1045, Fig. 14B (annotated).

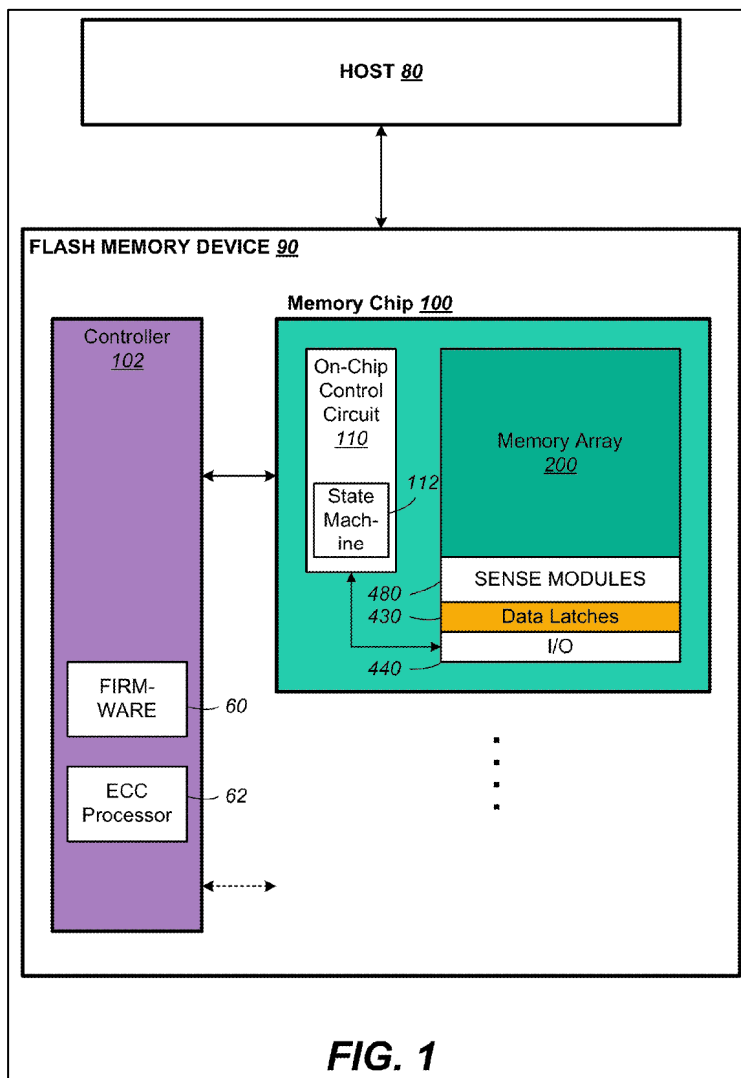
Gavens specifies that the post-write error check may be performed “by [a] comparison with the original copy which may be cached[.]” Ex. 1045, 16:46-49. At that passage, Gavens does not specify how the “original copy” is cached, but Gavens later explains that “[i]n the preferred embodiment” (where the antecedent basis of “the” refers to the immediately preceding “preferred embodiment” of Figures 14A-14B) the “data is cached in [a] first section of the first portion” of non-volatile

memory (*i.e.*, a section of the non-volatile SLC memory). *Id.*, 18:1-8 (referring to “the preferred embodiment”); *see also id.*, 16:14-16 (providing the antecedent basis of “a preferred embodiment”). This non-volatile memory from Gavens (including the MLC and SLC portions) is separate from Gavens’ controller. *See* Ex. 1045, Figure 1 (including Memory Array 200 separate from Controller 102). Thus, Gavens describes caching the “original copy” in non-volatile memory—not in RAM in the controller, as Petitioner appears to contend. Ex. 2001, ¶131.

Additionally, Gavens explains with the embodiment of Figure 16 why caching the original data in non-volatile memory improves the system. Like in the embodiment of Figure 14 discussed above, in the embodiment of Figure 16, Gavens further describes caching data in non-volatile memory (“first section 411”) where it is retained for use in error detection. *See* Ex. 1045, 18:21-44; *see also supra* Section VII.A.1.b (describing Gavens’s second embodiment of Figure 16). Gavens’s non-volatile cache retains data even after the data is written to a “high[er] density second portion” of the non-volatile memory. *See* Ex. 1045, 18:24-29. The retained data in the non-volatile cache can then be compared to the data written to the “higher density second memory” to check if the data was written correctly. *See id.*, 18:30-44; Ex. 2001, ¶132.

Gavens explains that data used for error detection should be cached in the non-volatile memory (*i.e.*, the “First section 411” of Figure 16) to avoid needing to toggle

data to the controller for error detection. Ex. 1045, 18:36-39 (“By not making the comparison at the controller, the data does not have to be toggled out to the controller, much time can be saved.”). Specifically, as illustrated in Gavens’s Figure 1 below, when data is cached in the SLC portion of the **non-volatile memory array 200** at the time it is written to the MLC portion of that same **array**, the **on-chip data latches 430** (part of **memory chip 100**) can be used for the post-write error check—thus avoiding the need to toggle the written data out to the **controller 102**. Gavens explains that caching data in the non-volatile memory improves performance by *reducing reliance on the controller chip*. See *id.*, 18:36-67; Ex. 2001, ¶133.



Ex. 1045, Fig. 1 (annotated).

Thus, a POSA would not have understood Gavens to disclose or suggest caching data in RAM for the “comparison with the original copy” cited by Petitioner. Gavens instead teaches that, for the cited “comparison,” data is cached in the non-volatile SLC memory. Further, Gavens *teaches away* from caching data *in the controller*—for example, in a “controller memory” as Petitioner contends. *See* Ex.

1045, 18:36-39 (“By not making the comparison at the controller, the data does not have to be toggled out to the controller, much time can be saved.”); Ex. 2001, ¶134.

Second, Petitioner fails to provide adequate obviousness analysis to explain, for example, why Gavens would have been modified or re-interpreted based on the teachings of Gorobets 912 and Paley to rely on a RAM-based cache in the controller for the disclosed data comparison. Petitioner suggests how Gavens should be “understood” but provides insufficient analysis to support its conclusions. Petition, 52-53 (asserting that Gavens’s “original copy” would be “***understood to be*** [cached in] RAVM” (emphasis added)), 53 (“***It would be obvious*** that the ‘stored data’ to be compared would be written to the controller memory (where the Gavens/Gorobets cache has space for at least two write units, nominally, ‘pages’) for comparison using logic on the controller chip....” (emphasis added)).

For example, Petitioner does not explain what aspect of the claim limitation is missing in Gavens and, correspondingly, does not explain what from Gavens must be modified to arrive at the claim language. And Petitioner also does not explain whether any of the allegedly incorporated references’ RAM is being used to modify Gavens’s system, and why/how a POSA would have been motivated to make any modification. Petitioner fails to perform a proper obviousness analysis, which requires, *inter alia*, ascertaining the “differences between the prior art and the claims at issue.” *Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17-18 (1966); *see*

also In re Stepan Co., 868 F.3d at 1346 n.1. The Petition’s obviousness analysis is insufficient and must be rejected.

Furthermore, contrary to Petitioner’s conclusory assertions, a POSA would have understood that Gavens’s non-volatile cache should not be replaced with a RAM-based cache in the controller, as disclosed by Gorobets 912 and Paley. Ex. 2001, ¶137. Specifically, as explained above, the non-volatile SLC cache of Gavens cited by Petitioner is used to **retain data** so it can be used in the subsequent post-write error-check. *See, e.g.*, Ex. 1045, 16:46-49. In contrast, as was also explained above, the RAM-based caches of Gorobets 912 and Paley cited by Petitioner do not retain data, and instead only “temporarily” store when it is being transferred to non-volatile memory. *See* Ex. 1049, ¶¶109, 144; Ex. 1051, ¶197; *see also supra* Sections VII.A.2-VII.A.3.

Thus, unlike the non-volatile cache of Gavens, Gorobets 912’s and Paley’s RAM-based based caches **do not retain data** and, instead, act only as write buffers. *See, e.g.*, Ex. 1049, ¶144 (explaining that the RAM-based cache is of “relatively limited capacity,” so as soon as data is “transferred out of [the] RAM” to the non-volatile memory, the RAM “open[s] up” the space). A POSA would not have been motivated to replace Gavens’s non-volatile cache with the RAM-based caches taught by Gorobets 912 or Paley, as these caches perform entirely different functions—**retaining** data (as in Gavens, and as required by the claims) versus **temporarily**

storing data (as in Gorobets 912 and Paley). *See* Ex. 2001, ¶138; *see also* Ex. 2039, 78-79 (in its decision denying institution of PGR2024-00047, the Board determined that Petitioner’s cited portions of Gorobets 912 “say ‘nothing about retaining data’ in RAM ‘for later use in a data integrity test performed by the controller’” and that “Paley’s disclosure...does not cure the deficiencies in Gavens and Gorobets [912]”).

Moreover, Gorobets 912 and Paley disclose RAM-based caches located in the controller. *See* Ex. 1049, ¶109 (“Typically high-speed volatile RAM are employed as cache as in a host cache 82 and/or in controller cache 102 of the memory controller”); Ex. 1051, ¶197 (same); *see also* Ex. 1049, ¶130. For example, Figure 11 of Gorobets 912, and the substantively identical Figure 15 of Paley, show “Cache (RAM) 102” in the memory controller. In contrast, Gavens teaches *against* using the controller for its error detection process. As described above, Gavens explains that data used for error detection should be cached in the non-volatile memory *to avoid having to toggle data to the controller*. Ex. 1045, 18:36-39 (“By not making the comparison at the controller, the data does not have to be toggled out to the controller, much time can be saved.”); *see also supra* Section VII.A.1. Gavens explains that caching data in the non-volatile memory increases performance by reducing reliance on the controller chip. *See id.*, 18:36-67. Relying on a cache in the controller (as may be required if the RAM-based cache of Gorobets 912 or Paley was used, and as required by Petitioner’s ground) rather than a cache in the non-

volatile memory would contradict the purpose of Gavens's invention. Ex. 2001, ¶139.

Petitioner provides no rationale to overcome this clear teaching away. Thus, replacing Gavens's non-volatile cache with a RAM-based cache taught by Gorobets 912 or Paley "would require a substantial reconstruction and redesign of the elements shown in [Gavens] as well as a change in the basic principles under which [Gavens] was designed to operate" and cannot be correct. *Application of Ratti*, 270 F.2d at 813; *see also* Ex. 2039, 71, 76 (decision denying institution of PGR2024-00047, finding that "Petitioner does not explain how the teachings of Gorobets [912] and Paley 'combine with the cited teachings of Gavens' to yield" a claim limitation requiring "**comparing** the stored data in the controller memory in the given one of the MLC nonvolatile memory elements to the **retained data** that was associated with the stored data in the random access volatile memory" (emphasis added)).

Thus, Petitioner fails to show that Gavens discloses or makes obvious the features of limitation [1e].

Furthermore, Limitations [18e] and [1e] are similar, and Petitioner relies on the same ground for both limitations. *See* Petition, 66-67. Accordingly, the Petition's analysis of limitation [18e] fails for the same reasons as those given above for limitation [1e].

**C. The Petition Does Not Establish That Dependent Claims 2-17
Would Have Been Obvious Over Gavens**

Petitioner's challenges to dependent claims 2-17 also lack merit, at least because these claims depend from claim 1, for which Petitioner's ground is deficient for the reasons stated above.

VIII. CONCLUSION

For the foregoing reasons, Petitioner has not established that it is more likely than not that it will prevail on any ground with respect to even one of the Challenged Claims. Accordingly, the Petition should be denied.

Date: May 14, 2025

Respectfully submitted,

/Alan Whitehurst/

Alan Whitehurst

Reg. No. 43,263

awhitehurst@mckoolsmith.com

Arvind Jairam

Reg. No. 62,759

ajairam@mckoolsmith.com

Christopher McNett

Reg. No. 64,489

cmcnett@mckoolsmith.com

Christian Dorman

Reg. No. 79,118

cdorman@mckoolsmith.com

McKool Smith, P.C.

1717 K St. NW, Suite 1000

Washington, DC 20006

Telephone: (202) 370-8300

Facsimile: (202) 370-8344

Counsel for Patent Owner

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. §42.24(d), Patent Owner hereby certifies that the word count for the foregoing **PATENT OWNER'S PRELIMINARY RESPONSE ON THE MERITS AND OTHER NON-DISCRETIONARY CONSIDERATIONS** is 13,906 words, which is within the limit set in 37 C.F.R. §42.24(b)(1). This word count does not include the table of contents, table of authorities, exhibit list, certificate of word count, and certificate of service. This word count was prepared using Microsoft Word.

Date: May 14, 2025

By: /Alan Whitehurst/
Alan Whitehurst, Reg. No. 43,263

CERTIFICATE OF SERVICE

I hereby certify, in accordance with 37 C.F.R. § 42.205, that the foregoing
**PATENT OWNER'S PRELIMINARY RESPONSE ON THE MERITS AND
OTHER NON-DISCRETIONARY CONSIDERATIONS** was served
via electronic mail on May 14, 2025 on the following counsel of record for

Petitioner:

Hsuanyeh Chang, PhD
hsuanyeh@hsuanyeh.com
Stephen Y. Chow
Stephen.Y.Chow@hsuanyeh.com
Douglas E. Chin
Doug.Chin@hsuanyeh.com
Peter Yi
Peter.Yi@hsuanyeh.com
HSUANYEH LAW GROUP PC
11 Beacon Street, Suite 900
Boston MA 02108

By: /Alan Whitehurst/

Dated: May 14, 2025

Alan Whitehurst, Reg. No. 43,263