

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PHISON ELECTRONICS CORPORATION,
Petitioner,

v.

VERVAIN, LLC,
Patent Owner.

PGR2025-00010
U.S. Patent No. 11,967,369

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

TABLE OF CONTENTS

	Page
EXHIBIT LIST	viii
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. Claim 1	11
III. PERSON OF ORDINARY SKILL IN THE ART	14
IV. CLAIM CONSTRUCTION	14
V. GROUND 1: 35 U.S.C. § 101	16
A. The Petition Does Not Establish Patent Ineligibility Under § 101	16
1. Petitioner is Wrong About <i>Alice</i> Step One	17
2. Petitioner is Wrong About <i>Alice</i> Step Two	19
VI. GROUND 2: 35 U.S.C. § 112(a).....	20
A. The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(a)	20
1. 2.A: “Memory Space”	21
2. 2.B: “Memory Element”	23
3. 2.C: “Mapping...As Necessitated By The System To Maximize Lifetime”	24

4.	2.D: “RAVM, Controller Memory And Their Respective Relationships To Stored And Retained Data”	26
5.	2.E: “At Least One Memory Element”	29
VII.	GROUND 3: 35 U.S.C. § 112(b)	31
A.	The Petition Does Not Establish That Any Term Is Unpatentable Under 35 U.S.C. § 112(b)	31
1.	3.A: “Memory Space”	31
2.	3.B: “Memory Element”	34
3.	3.C: “Mapping...As Necessitated By The System To Maximize Lifetime”	35
4.	3.D: “Controller” Limited By Claimed Functions.....	37
5.	3.E: Claims Reciting Method Steps In The Context Of A System.....	43
6.	3.F: Whether And When The Claimed Comparison Test Is Required	45
VIII.	GROUND 4: 35 U.S.C. § 103	46
A.	Overview Of The Cited Prior Art.....	46
1.	Gavens (Ex. 1045).....	47
2.	Gorobets 912 (Ex. 1049).....	55
3.	Paley (Ex. 1051).....	57
4.	Harari (Ex. 1057)	59
5.	Gorobets PCT (Ex. 1058)	60
6.	Moshayedi (Ex. 1043).....	60
B.	The Petition Does Not Establish That Independent Claim 1 Would Have Been Obvious Over Gavens.....	62

1.	[1c]: “at least one <i>random access volatile physical memory element</i> that exists in the logical volatile memory space”	62
2.	[1d]: “at least one controller to operate physical memory elements and associated memory space for associated Write access operations to the physical memory elements and Read access operations from the physical memory elements, <i>the controller having a controller memory associated therewith</i> for storing received data therein”	68
3.	[1f]: “the controller...[1f3] operable to store the received data in the given one of the physical MLC nonvolatile memory elements as stored data and <u><i>retain such received data in the random access volatile physical memory element</i></u> as retained data associated with stored data”	71
4.	[1g]: “the <u><i>controller</i></u> performing a data integrity test...and [1g3] <i>comparing the stored data in the controller memory</i> in the given one of the physical MLC nonvolatile memory elements <i>to the retained data</i> that was associated with the stored data <i>in the random access volatile physical memory element</i> by the controller during the Write access operation”	81
C.	The Petition Does Not Establish That Dependent Claims 2-10 Would Have Been Obvious Over Gavens And/Or Obvious Over Gavens In View Of Colligan.....	84
IX.	CONCLUSION.....	84

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Alice Corp. Pty. v. CLS Bank Int’l</i> , 573 U.S. 208 (2014).....	16, 17, 19
<i>Allergan et al. v. BTL Healthcare</i> , PGR2021-00020, Paper 16 (P.T.A.B. Jun. 16, 2021).....	30
<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).....	67
<i>Ancora Techs., Inc. v. HTC Am., Inc.</i> , 908 F.3d 1343 (Fed. Cir. 2018)	18
<i>Application of Ratti</i> , 270 F.2d 810 (C.C.P.A. 1959)	83
<i>Ariad Pharms., Inc. v. Eli Lilly & Co.</i> , 598 F.3d 1336 (Fed. Cir. 2010)	21, 24
<i>Atlas Powder Co. v. E.I. du Pont de Nemours & Co.</i> , 750 F.2d 1569 (Fed. Cir. 1984)	30
<i>Barkan Wireless IP Holdings, L.P. v. Samsung Elecs. Co., Ltd.</i> , 2019 WL 497902 (E.D. Tex. Feb. 7, 2019).....	40
<i>Becton, Dickinson & Co. v. Tyco Healthcare Grp., LP</i> , 616 F.3d 1249 (Fed. Cir. 2010)	70
<i>Callaway Golf Co. v. Acushnet Co.</i> , 576 F.3d 1331 (Fed. Cir. 2009)	59, 60
<i>Capella Photonics, Inc. v. Fujitsu Network Commc’ns, Inc.</i> , 2021 WL 465430 (E.D. Tex. Feb. 9, 2021).....	39
<i>Captioncall, LLC v. Ultratec, Inc.</i> , No. IPR2020-01215, 2021 WL 278381 (P.T.A.B. Jan. 27, 2021)	65, 71, 78
<i>In re De Seversky</i> , 474 F.2d 671 (C.C.P.A. 1973)	59

<i>Enfish, LLC v. Microsoft Corp.</i> , 822 F.3d 1327 (Fed. Cir. 2016)	16, 18
<i>Graham v. John Deere Co. of Kansas City</i> , 383 U.S. 1 (1966).....	66
<i>Intelligent Automation Design, LLC v. Zimmer Biomet CMF & Thoracic, LLC</i> , 799 F. App'x 847 (Fed. Cir. 2020)	42
<i>IPXL Holdings, L.L.C. v. Amazon.com, Inc.</i> , 430 F.3d 1377 (Fed. Cir. 2005)	43, 44, 45
<i>Martek Biosciences Corp. v. Nutrinova, Inc.</i> , 579 F.3d 1363 (Fed. Cir. 2009)	21
<i>Masimo Corp. v. Apple Inc.</i> , IPR2023-00745, Paper 44 (P.T.A.B. Oct. 15, 2024).....	67, 70
<i>Mastermine Software v. Microsoft Corp.</i> , 874 F.3d 1307 (Fed. Cir. 2017)	44
<i>Microbiotica, Limited v. Board of Regents, The University of Texas System</i> , PGR2023-00026, Paper 12 (P.T.A.B. Nov. 21, 2023)	21, 25, 26, 29
<i>Nautilus, Inc. v. Biosig Instruments, Inc.</i> , 572 U.S. 898 (2014).....	46
<i>Nearmap US, Inc. v. Eagle View Technologies, Inc.</i> , IPR2024-00716, Paper 9 (P.T.A.B. Oct. 9, 2024).....	68
<i>Netflix, Inc. v. DivX, LLC</i> , 84 F.4th 1371 (Fed. Cir. 2023)	68
<i>Niazi Licensing Corp. v. St. Jude Med. S.C., Inc.</i> , 30 F.4th 1339 (Fed. Cir. 2022)	32, 35, 37, 46
<i>Rain Computing, Inc. v. Samsung Elecs. Am., Inc.</i> , 989 F.3d 1002 (Fed. Cir. 2021)	41
<i>Sound View Innovations, LLC v. Facebook, Inc.</i> , 2017 WL 2221177 (D. Del. May 19, 2017)	41

<i>In re Stepan Co.</i> , 868 F.3d 1342 (Fed. Cir. 2017)	67
<i>StrikeForce Techs. Inc. v. PhoneFactor Inc.</i> , 2015 WL 5708577 (D. Del. Sept. 29, 2015).....	42
<i>Sysmex Corp. v. Beckman Coulter, Inc.</i> , 2021 WL 1259710 (D. Del. Apr. 6, 2021)	39
<i>Telcordia Techs., Inc. v. Cisco Sys., Inc.</i> , 612 F.3d 1365 (Fed. Cir. 2010)	38
<i>Typhoon Touch Techs., Inc. v. Dell, Inc.</i> , 659 F.3d 1376 (Fed. Cir. 2011)	42
<i>Va. Innovation Scis., Inc. v. Amazon.com, Inc.</i> , 2019 WL 4259020 (E.D. Tex. Sept. 9, 2019).....	40, 41
<i>Vervain, LLC v. Phison Electronics Corp.</i> , 1:24-cv-00259 (W.D. Tex.)	15
<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).....	38, 40
<i>Visual Memory LLC v. NVIDIA Corp.</i> , 867 F.3d 1253 (Fed. Cir. 2017)	18, 19
<i>Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.</i> , 200 F.3d 795 (Fed. Cir. 1999)	14
Statutes	
35 U.S.C. § 101	1, 16
35 U.S.C. § 103	46, 47
35 U.S.C. § 112	<i>passim</i>
35 U.S.C. § 112(a)	1, 20, 29
35 U.S.C. § 112(b)	1, 24, 31

35 U.S.C. § 112(f).....	38
35 U.S.C. § 312(a)(3).....	67
35 U.S.C. § 322(a)(3).....	70
35 U.S.C. § 323	1
35 U.S.C. § 324(a)	2

Other Authorities

37 C.F.R. § 42.22(a)(2)	67
37 C.F.R. § 42.207	1

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	Patent Office Subject Matter Eligibility Guidance Examples 1-36
2009	[RESERVED]
2010	[RESERVED]
2011	<i>Micron Technology, Inc., v. Vervain, LLC</i> , IPR2021-01550, Ex. 1009 (P.T.A.B. Sept. 21, 2021) (expert declaration challenging the validity of U.S. Patent No. 10,950,300)
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)

Exhibit No.	Description
2016	Micron’s Petition for <i>Inter Partes</i> Review of U.S. Patent No. 10,950,300, IPR2021-01550, Sept. 21, 2021 (“300 petition”)
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

Exhibit No.	Description
	<i>Vervain, LLC v. Kingston Technology Company., et al.</i> No. 1:24-cv-254-ADA.
2033	[RESERVED]
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

Patent Owner Vervain, LLC (“PO” or “Vervain”) submits this Preliminary Response in accordance with 35 U.S.C. § 323 and 37 C.F.R. § 42.207, responding to the Petition for Post Grant Review (PGR) (Paper 2) of U.S. Patent No. 11,967,369 (“369 Patent” or “the Challenged Patent”) filed by Phison Electronics Corporation (“Phison” or “Petitioner”). The Petition includes four grounds of unpatentability challenging claims 1-10 (“Challenged Claims”). The first three grounds include cursory challenges under 35 U.S.C. §§ 101, 112(a), and 112(b). The fourth 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 lac 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 retaining stored data in RAM, a feature that Gavens neither teaches nor suggests.

Moreover, *the Board already denied institution of petitions presenting similar grounds*. See Ex. 2039 (denying institution of PGR2024-00047); Ex. 2041 (denying institution of PGR2024-00048). As reflected in these prior decisions,

Petitioner has not established that it is more likely than not that it will prevail on any ground. *See* 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 November 7, 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

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, 2:50-57; Ex. 2001, ¶27.

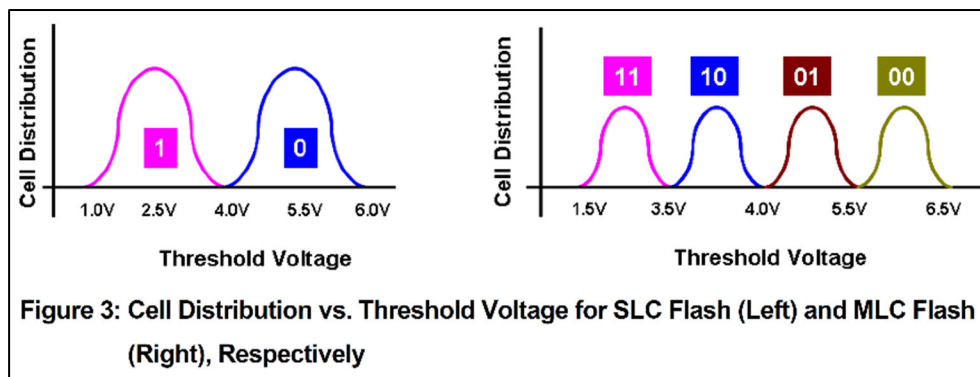
and lifetime (endurance).” *Id.*, 3:39-42. As a result, various hybrid systems combining SLC and MLC (among others) are taught in the Challenged Patent, to combine the benefits of both types of non-volatile flash storage at a low cost. *Id.*, 3:60-67.

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.*, 3:60-67. 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, 3:65-67.

A. SLC and MLC Flash

SLC memory stores 1 bit per cell, and MLC memory stores more than 1 bit per cell. Ex. 1001, 2:11-17; 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, 1:54-59. 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.*, 3:39-42.

SLC and MLC flash memories both use the same type of transistor called a floating gate transistor. *Id.*, 3:48-51. 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, 3:36-37. 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, 3:37-39.

B. Address Table

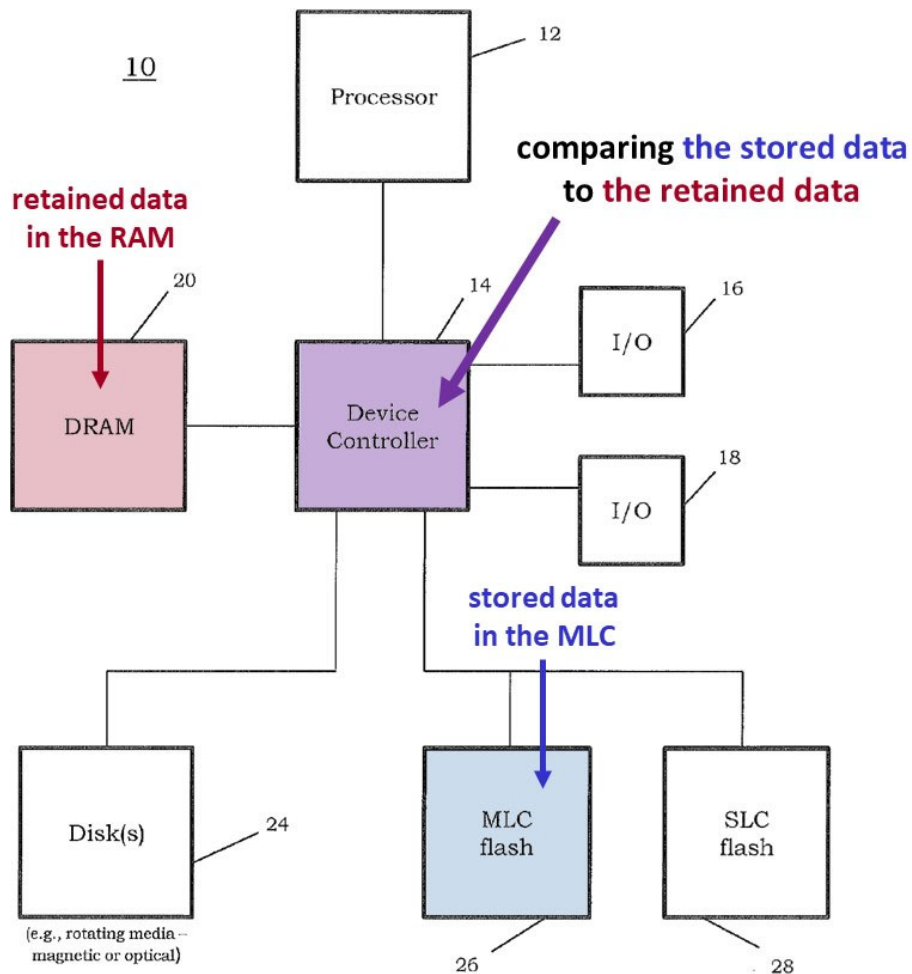
To provide wear leveling, garbage collection, and bad block management, a translation layer is used to map logical addresses to actual physical memory locations. Ex. 2003, 40-42; Ex. 1001, 3:1-33; *see also* Ex. 2001, ¶35. As part of this translation layer, “tables are widely used in order to map sectors and pages from logical to physical.” Ex. 2003, 40; *see also* Ex. 1001, 3:17-21. These tables map logical blocks to physical blocks. Ex. 2003, 40-42; Ex. 1001, 3:17-21. 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, 2:57-67. 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, 5:50-54. Dr. Rao further explained that memory is written and mapped on the granularity of a “quantum,” such as a block or page. *Id.*, 5:50-54; 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, 5:57-6:61. Alternately, MLC data can be remapped to other MLC blocks, and the address table is then modified accordingly. *Id.*, 3:11-33.

The Challenged Patent claims 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).



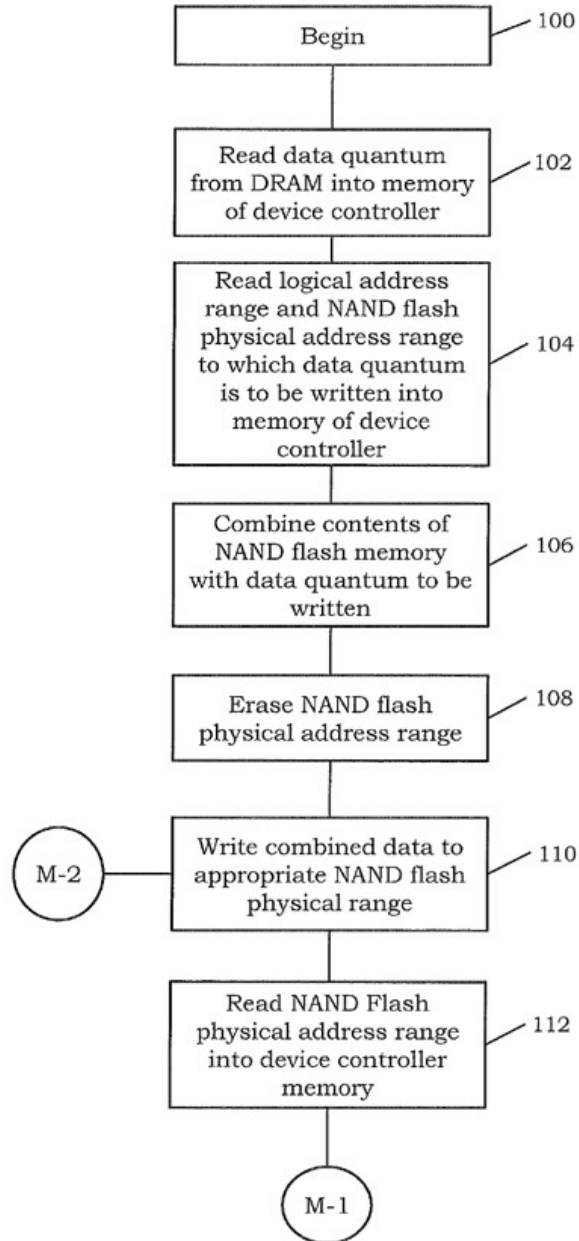
Ex. 1001, Fig. 1 (annotations added²), 5:30-31.

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, 5:55-57. If time permits, the controller performs a data integrity test on the data that was stored in the MLC. *Id.*, 5:57-60. If the

² Unless otherwise noted, PO added coloring to Figures.

stored 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.*, 5:60-5:63.

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.*, 6:13-16. In step 102, the data to be written to the MLC is read from the DRAM into the controller memory. *Id.*, 6:16-18. In step 104, the controller obtains the logical and physical address ranges for the data to be written to the MLC. *Id.*, 6:18-21. In step 106, the controller combines the data from the DRAM with data from the MLC. *Id.*, 6:21-23. In step 108, the controller erases the physical address range where the data is being written. *Id.*, 6:23-24. Finally, in step 110, the combined data is written to the MLC flash. *Id.*, Fig. 3A (step 110), 6:24-26.

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.*, 6:26-28. In step 114, the stored data is compared to the data that was retained in the DRAM (“the retained data”). *Id.*, 6:29-34.

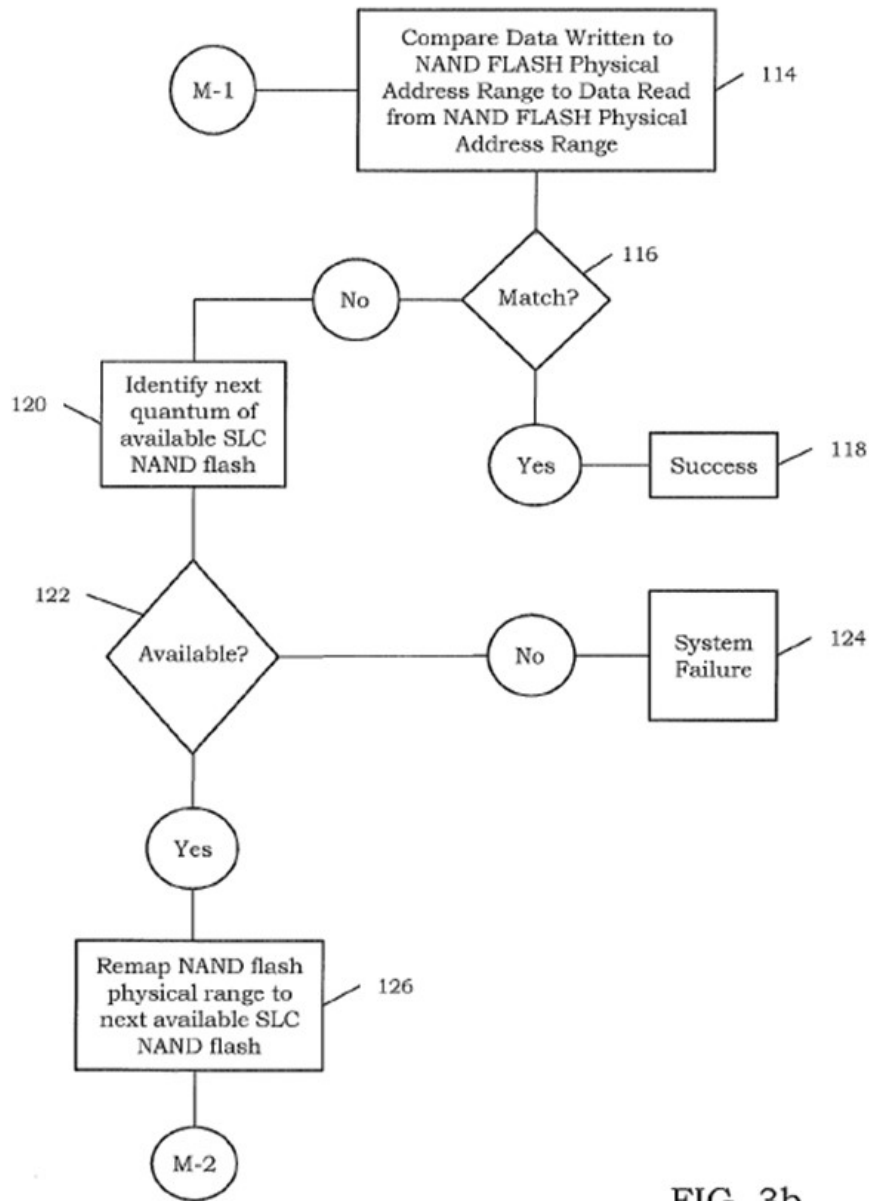


FIG. 3b

Id., Fig. 3B.

If there is a match, the data was successfully stored in the MLC. *Id.*, 6:34-36.

If not, the retained data is written to another location (*e.g.*, SLC). *Id.*, 6:36-40.

D. Claim 1

Claim 1 of the Challenged Patent requires a system for storing data comprising: (1) non-volatile MLC memory; (2) non-volatile SLC memory; (3) random access volatile memory (highlighted red below); and (4) controller memory. The random access volatile memory is a key feature of the invention and it permeates many of the elements of the claim ([1c], [1f]-[1g]). As discussed above, the random access volatile memory is particularly important to the “data integrity test” in [1h].

Claim 1	
[1Pre]	A memory system for storing data comprising:
[1a]	memory space containing logical and physical volatile memory space and nonvolatile memory space, wherein the logical and physical nonvolatile memory space includes both multi-level cell (MLC) logical and physical memory space and single level cell (SLC) logical and physical memory space;
[1b]	a bank of physical nonvolatile memory, including: [1b1] a plurality of physical MLC memory modules, each including at least one physical MLC nonvolatile memory element having a physical address within the physical MLC nonvolatile memory space that can be mapped to a logical address in the MLC logical nonvolatile memory space, and [1b2] at least one SLC memory module including at least one physical SLC nonvolatile memory element having a physical address within the

	physical SLC nonvolatile memory space that can be mapped into the SLC logical nonvolatile memory space;
[1c]	at least one random access volatile physical memory element that exists in the logical volatile memory space;
[1d]	at least one controller to operate physical memory elements and associated memory space for associated Write access operations to the physical memory elements and Read access operations from the physical memory elements, the controller having a controller memory associated therewith for storing received data therein;
[1e]	a flash translation layer (FTL), wherein the at least one controller, or the FTL, or a combination of both maintain an address table in one or more of the physical memory elements;
[1f]	[1f1] the controller controlling access of the physical MLC and SLC nonvolatile memory elements and the random access volatile physical memory elements for storage of the received data therein as stored data, [1f2] the controller, in at least a Write access operation transferring the stored received data from the controller memory to a given one of the physical MLC nonvolatile memory elements in an associated physical MLC memory module, [1f3] operable to store the received data in the given one of the physical MLC nonvolatile memory elements as stored data and retain such received data in the random access volatile physical memory element as retained data associated with stored data;
[1g]	the controller [1g1] performing a data integrity test on stored data in the given one of the physical MLC nonvolatile memory elements in the associated one of the physical MLC memory modules after the at least a

	Write access operation performed thereon by [1g2] reading the stored data to the controller memory and [1g3] comparing the stored data in the controller memory in the given one of the physical MLC nonvolatile memory elements to the retained data that was associated with the stored data in the random access volatile physical memory element by the controller during the Write access operation;
[1h]	wherein the address table maps logical addresses to physical addresses of the memory system, wherein the mapping is performed as necessitated by the memory system to maximize lifetime; and
[1i]	wherein a failure of the data integrity test performed on the stored data by the controller results in a remapping of logical address space to a different physical range of addresses in the physical address space and transfer of data corresponding to the retained data to those remapped physical addresses in the physical address space from those physical addresses in the physical address space determined to have failed the data integrity test.

In [1d], the controller stores received data in the controller memory. The data is transferred to two places: to the MLC non-volatile memory as “stored data” ([1f]) and to the random access volatile memory as “retained data” ([1f]).

After data is written to the MLC, the controller performs a data integrity test to determine whether the data has any errors ([1g]). To do so, the controller reads the “stored data” to the controller memory ([1g2]) and compares this data to the “retained data” in the random access volatile memory ([1g3]).

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, 27; Ex. 2001, ¶¶22-26.³

IV. CLAIM CONSTRUCTION⁴

Petitioner proposed construction of four terms: "MLC memory modules," "SLC memory module," "MLC memory element," "SLC memory element." Petition, 27-29. 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

³ PO submits the declaration of Dr. Sunil Khatri, an expert in the field of the Challenged Patent. Ex. 2001, ¶¶1-26.

⁴ PO addresses the indefiniteness grounds separately in Section VII below.

continuation. *Compare* Petition, 28 *with* PGR2024-00047, Paper 2 (Petition) at 24 (submitted here as Ex. 2029). However, neither of Petitioner’s constructions makes sense.

Further, 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. Specifically, the Court rejected Petitioner’s proposed constructions, instead adopting PO’s proposed constructions, for “MLC non-volatile memory,” “SLC non-volatile memory,” and “memory element.” *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. § 101

Petitioner asserts that the claims are directed towards patent-ineligible subject matter—namely, an allegedly abstract “memory space.” Petition, 29-32. This challenge fails. Moreover, the Board already considered—and rejected—substantially the same § 101 challenge in denying institution of PGR2024-00047. *See* Ex. 2039, 22-25, 23 (“[W]e understand ‘memory space’ as referencing physical memory rather than an abstract property.”).

A. The Petition Does Not Establish Patent Ineligibility Under § 101

Petitioner’s § 101 analysis is wrong regarding both steps of the *Alice* test. Ex. 2001, ¶¶46-58. In determining whether a claim is directed to patent-ineligible subject matter under the *Alice* framework, both the Supreme Court and Federal Circuit warn against characterizing the claim at “a high level of abstraction and untethered from the language of the claims” as such a characterization “all but ensures that the exceptions to § 101 swallow the rule.” *Enfish, LLC v. Microsoft Corp.*, 822 F.3d 1327, 1337 (Fed. Cir. 2016) (citing *Alice Corp. Pty. v. CLS Bank Int’l*, 573 U.S. 208, 217 (2014)). Petitioner’s characterization of the Challenged Claims impermissibly untethers the single element of a “memory space” from all the remaining claim language to assert that each and every one of the Challenged Claims

is directed exclusively to this single element. Petitioner also argues that each of the limitations of the Challenged Patent, other than the “memory space,” represent “well-understood, routine, and/or conventional” computer elements, without providing any analysis of what these other limitations require or how they are allegedly well-understood, routine, and/or conventional. Petition, 31. Petitioner’s conclusory arguments fail to show any likelihood of the Challenged Claims being found to be directed to ineligible subject matter.

1. Petitioner is Wrong About *Alice* Step One

The Challenged Claims are not directed to the abstract idea of a “memory space” as Petitioner contends. The Challenged Claims are instead directed to a “system for storing data” which includes device elements that are both physical and highly unconventional. *See* Ex. 2001, ¶¶46-58. These additional limitations further make it clear that the claims cannot be directed to a transitory signal, as no transitory signal could read on, among other limitations, “at least one controller to operate physical memory elements.” Challenged Claim 1, Limitation [1d].

The claimed subject matter of the Challenged Patent is very similar to an example in the USPTO Subject Matter Eligibility Guidelines of claims not directed to an abstract idea. *See* Ex. 2008 (USPTO § 101 Guidelines), 21 (example 27 reciting a method for storing BIOS in a system with volatile and non-volatile memory). This example recites a particular method for transferring and storing

BIOS in memory upon the local computer powering up. Similarly, here, the Challenged Claims recite a particular system that transfers retained data to a different physical address based on a data integrity test where there is “no apparent exception [*i.e.* abstract idea] recited in the claims,” thereby ensuring they are patent eligible. Ex. 2008, 22. Moreover, courts regularly find that claims relating to improvements in computer memory are directed to patent eligible subject matter. *See, e.g., Enfish*, 822 F.3d at 1346; *Visual Memory LLC v. NVIDIA Corp.*, 867 F.3d 1253 (Fed. Cir. 2017); *Ancora Techs., Inc. v. HTC Am., Inc.*, 908 F.3d 1343 (Fed. Cir. 2018), as amended (Nov. 20, 2018). Notably, in *Visual Memory*,⁵ the Federal Circuit overturned the district court’s finding of patent ineligibility, because the claimed

⁵ Claim 1 of the patent at issue in *Visual Memory*, which was found to be patent eligible by the Federal Circuit, recited in its entirety:

1. A computer memory system connectable to a processor and having one or more programmable operational characteristics, said characteristics being defined through configuration by said computer based on the type of said processor, wherein said system is connectable to said processor by a bus, said system comprising:

a main memory connected to said bus; and

a cache connected to said bus;

wherein a programmable operational characteristic of said system determines a type of data stored by said cache.

“computer memory system” was not directed to an abstract idea, and accordingly encompassed patent eligible subject matter, despite the claims at issue in that case containing significantly less detail defining the claimed memory system than those of the Challenged Patent. *Visual Memory*, 867 F.3d at 1262.

Similar to the system in *Visual Memory*, the system of the Challenged Claims improves computer functionality by allowing for remapping of address space based in part on a “data integrity test.” Ex. 1001 at 5:10-18; Ex. 2001, ¶¶52-58. This re mapping helps to optimize memory storage to improve system capacity, endurance, and access speed. Ex. 1001 at 4:28-39; Ex. 2001, ¶¶54-58. Moreover, the Challenged Claims recite significantly greater detail than the claims in *Visual Memory*, further confirming that the Challenged Patent does not tie up all ways of implementing any allegedly abstract idea.

2. Petitioner is Wrong About *Alice* Step Two

Second, even if the Challenged Claims were directed to an abstract idea, which PO does not concede, these claims would still be patent eligible under step 2B of the *Alice* framework, as they include inventive limitations that amount to “significantly more” than an abstract memory space *per se*. Ex. 2001, ¶¶48-49, 52-58. The Challenged Claims address a meaningful technical problem arising from the need to balance the various parameters of capacity, endurance, and accessibility in memory systems. Ex. 2001, ¶¶51-58. As Dr. Khatri explains, the Challenged

Claims embody a meaningful technical solution to this problem by, among other things, remapping logical address space to different physical range of addresses in the physical address space if data stored in a physical MLC nonvolatile memory element fails a data integrity test, which improves the performance of the memory system by, for example, ensuring that data is reliably stored in locations of memory where integrity is not in doubt. Petitioner also claims that every limitation of the Challenged Claims, other than the “memory space,” recites well-understood, routine, and/or conventional elements in a conventional arrangement, but fails to provide any analysis or evidence to support this unfounded argument. Petition, 31. To the contrary, the claims embody inventive techniques for improving memory operations as discussed above. Ex. 2001, ¶¶48-49, 52-58.

VI. GROUND 2: 35 U.S.C. § 112(a)

Petitioner asserts that seven terms lack written description support and/or enablement. Petition, 32-39. These challenges fail.

Moreover, the Board has already considered—and rejected—four of Petitioner’s five § 112(a) arguments in denying institution of PGR2024-00047. Ex. 2039, 25-30.

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. 2.A: “Memory Space”

“Memory space” is clearly disclosed in the specification and is fully enabled. *See* Ex. 2001, ¶¶63-64. 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 four sentences. Petition, 33; *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, 33. 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, 5:18. 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, ¶64. 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:

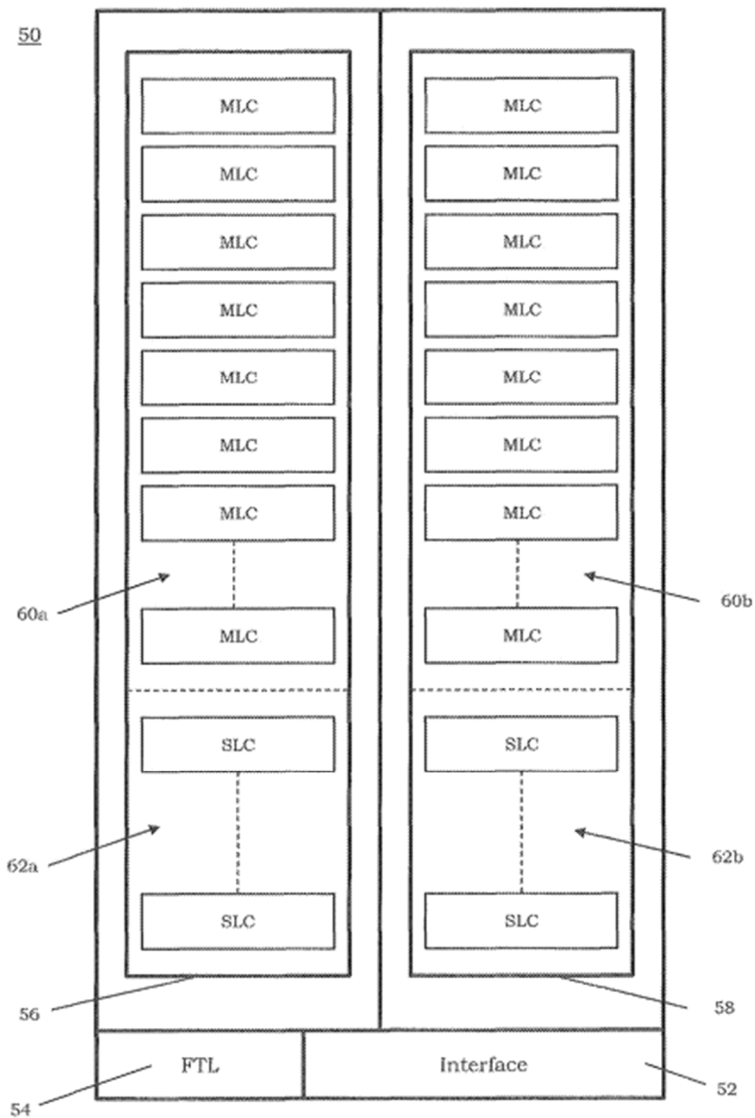


FIG. 4

Ex. 1001, Fig. 4.

2. 2.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,

¶65. In another threadbare argument, Petitioner suggests that “memory element” is not “mentioned...in the specification.” Petition, 33-34, 33. 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, ¶65. The specification states that cells of SLC or MLC flash memory may store a single bit or multiple bits per cell, respectively. Ex. 1001, 2:11-18. 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, 5:39-7:5. “Memory element” was a readily understood term in the art, and simply refers to an element of memory. *See* Ex. 2001, ¶65.

3. 2.C: “Mapping...As Necessitated By The System To Maximize Lifetime”

The Board has already determined that this term has written description support. *See* Ex. 2039, 32-33; *see also* Ex. 2006, 7 (district court previously rejecting a §112(b) challenge to this term). There is no merit to Petitioner’s challenge here—the specification clearly conveys possession of “mapping...as necessitated by the system to maximize lifetime,” and it was fully enabled. Ex. 2001, ¶66.

Petitioner's ground again should be rejected outright as ambiguous and inadequately supported. Petition, 34-36. Petitioner's argument is disjointed, beginning as an apparent written description challenge but later ambiguously referencing enablement. *See* Petition, 34 (beginning with the "Challenged claims lack any written description..."), 36 (ending with the "specification thus does not enable..."); *see also Microbiotica*, PGR2023-00026, Paper 12 at 24, 27 (denying institution of written description and enablement grounds when petitioner "d[id] not provide sufficient analysis of the full disclosure of the [challenged] patent" and included "enablement challenges repeat[ing] the written description arguments in one sentence, and provide[d] no more analysis, much less under the *Wands* factors").

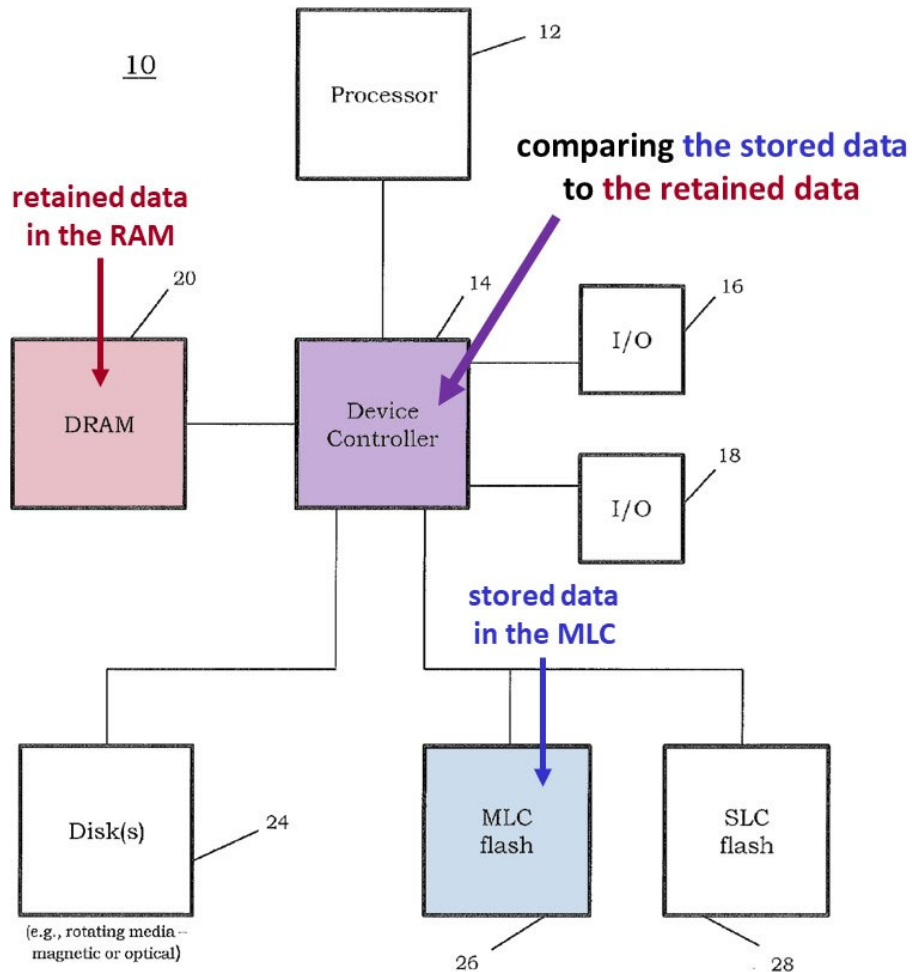
Even so, the Challenged Patent explains—as is claimed—that mapping is performed to increase the "reliability and lifetime" of the memory. Ex. 1001, 1:41-48. Particularly, Figs. 2A-2B and 3A-3B show how data is remapped to boost or prolong the lifetime of the system. If, for example, there is failure of a data integrity test, the data is transferred to a block with higher endurance. *Id.*, 5:60-63. Similarly, if a block is heavily used, the data is mapped to a block that is less-used, or a higher-performance block. *Id.*, 3:26-33 ("selects an available target block with the lowest overall erase count"), 6:55-61 ("on a periodic basis, such as, for example, every 100 writes, or every 10,000 writes, transfer the contents of those blocks into the SLC NAND flash..."). In each case, the lifetime of the memory is maximized. *Id.*

4. 2.D: “RAVM, Controller Memory And Their Respective Relationships To Stored And Retained Data”

In another conclusory ground, Petitioner suggests that the “recited RAVM and controller memory and their relationship to ‘stored data’ and ‘retained’ data” lacks written description support and is not enabled. Petition, 36-38. Petitioner is wrong—the Challenged Patent includes clear disclosure supporting the claims, including for these features. Ex. 2001, ¶¶67-70.

Notwithstanding, Petitioner’s argument should again be rejected outright as ambiguous and inadequately supported. Petition, 36-37. Again disjointed, Petitioner’s argument begins as an apparent “written description” challenge but ends with an unclear assertion that the claims “are not enabled.” *Id.* But Petitioner’s challenge includes no analysis of the Challenged Patent’s specification itself, instead citing only the claim language, and includes no analysis “under the *Wands* factors.” *See Microbiotica*, PGR2023-00026, Paper 12 at 24, 27.

Still, Petitioner’s argument—to the extent it can be understood—lacks merit. The Challenged Patent includes a clear description of the claimed random access volatile memory and controller memory, and how they relate to stored data and retained data. Particularly, as explained in Section I.C above and as seen in Figure 1 below, the controller (purple) controls a DRAM (red).



Ex. 1001, Fig. 1 (annotations added), 5:30-31.

The DRAM is used during a write access operation to the MLC flash (highlighted blue). Data to be written to the MLC flash is maintained (or retained) in the DRAM. Ex. 1001, 5:55-57 (“Data to be written to NAND flash is *maintained in DRAM 20*[.]” (emphasis added)); Ex. 2001, ¶69.

Further, Figures 3A and 3B show how the DRAM and controller memory may be used during a data integrity test. “In step 102, [a] quantum of data is read *from DRAM into memory within the device controller (which acts as the memory*

controller).” Ex. 1001, 6:16-17 (emphasis added). The data is then written to the “NAND flash” (steps 104 through 110), after which it is “*read [back] into the device controller memory.*” Ex. 1001, 6:17-27 (emphasis added). “In step 114 [of Figure 3B] the NAND flash physical address range that was read into device controller memory is compared with the retained data representing the combination of the previous contents of the physical address range and the quantum of data to be written. In step 116, if the retained data matches the newly stored data in the NAND flash memory, the write was a success....” Ex. 1001, 6:29-36; Ex. 2001, ¶70.

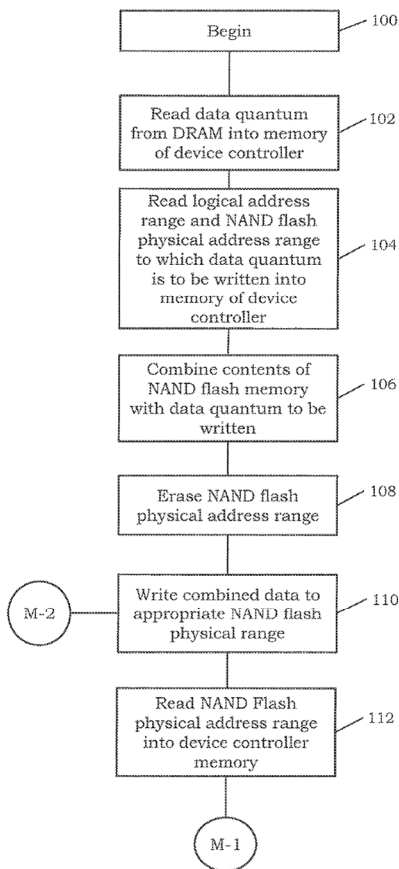


FIG. 3A

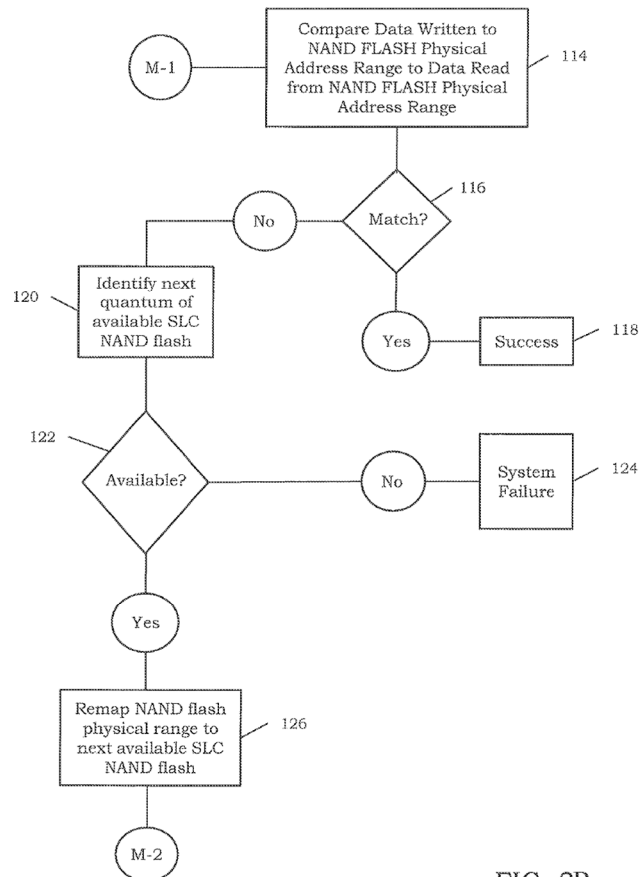


FIG. 3B

Ex. 1001, Figs. 3A and 3B.

Thus, the Challenged Patent includes a clear disclosure of the claimed controller memory and random access volatile memory, and their relationship to stored data and retained data, and Petitioner fails to show otherwise in its conclusory ground. *Contra* Petition, 37 (Petitioner, without citing to the Challenged Patent’s specification itself, conclusorily alleges that “there is no disclosure in the claim or specification where or how the ‘stored data’ in the controller memory is compared to the ‘retained data’”—this is, however, clearly disclosed, as explained above).

5. 2.E: “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, ¶¶71-72. 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). The lack of adequate explanation and the failure to present a §112(a) analysis is alone a basis to reject this ground. *See Microbotica*, PGR2023-00026, Paper 12 at 24, 27.

In any event, the specification provides a clear, enabling description of the “*at least one physical MLC nonvolatile* memory element” ([1b1]) and “*at least one*

physical SLC nonvolatile memory element” ([1b2]). 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, ¶71.

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, ¶72. But the Petition attempts to argue that a contrived, single page system would not be useful yet technically meet the claim language. Petition, 37-39, 38 (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.

In addition, Petitioner’s argument is facially false. Petitioner suggests that the claims of the Challenged Patent can be “met with *just one* SLC memory element and *one* MLC memory element....” Petition, 38. That is not what claim 1 requires. Instead, [1b] requires at least *two* physical MLC nonvolatile memory elements ([1b1], requiring “a plurality”) and at least one physical SLC nonvolatile memory element ([1b2]).

VII. GROUND 3: 35 U.S.C. § 112(b)

Petitioner asserts that six claim terms are indefinite. Petition, 39-46. 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 369 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 369 Patent, again rejecting similar indefiniteness challenge).

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

1. 3.A: “Memory Space”

Petitioner suggests in a cursory and ambiguous manner that “memory space” is indefinite because “of the recitation that RAVM ‘physical memory element’ ‘exists in the logical volatile memory space,’ and the controller memory is no part of the space at all.” Petition, 39-40 (emphasis omitted). 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, ¶¶73-77. To the contrary, claim 1 of the Challenged Patent requires “memory space containing logical and physical volatile memory space and nonvolatile memory space, wherein the logical and physical nonvolatile memory space includes both multi-level cell (MLC) logical and physical memory space and single level cell (SLC) logical and physical memory space” (limitation [1a]) where “at least one random access volatile physical memory element...exists in the logical volatile memory space” (limitation [1c]). Claim 1 also requires wherein “the controller having a controller memory associated therewith” (limitation [1d]). Particularly in light of this context, a POSA would find no ambiguity regarding what “memory space” refers to. Ex. 2001, ¶¶75-76.

“Memory space” simply refers to where data is stored. Ex. 2001, ¶74. 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, ¶75. 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, 7:41-52. Furthermore, the specification makes it clear that the “total non-volatile memory storage space=MLC+SLC.” Ex. 1001, 5:18. 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, ¶76.

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, ¶77.

2. 3.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 four 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, ¶¶78-83. “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, 1:29-37 (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, 40. 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. 3.C: “Mapping...As Necessitated By The System To Maximize Lifetime”

Both the Board and District Court have already rejected Petitioner’s assertion that “mapping...as necessitated by the system to maximize lifetime” is indefinite. Ex. 2039, 58-59; Ex. 2032, 50:9-54:13; *see also* Ex. 2006, 7 (earlier litigation, not involving the Challenged Patent but involving its parent patents, in which the court

also determined that this term is not indefinite). There is no reason for the Board to change its construction here. *See* Petition, 39-41.

A POSA would readily understand what it means to perform “mapping...as necessitated by the system to maximize lifetime.” Ex. 2001, ¶¶84-89. The point of the Challenged Patent is to increase the “reliability and lifetime” of the memory, and the specification is replete with examples of how to do this. Ex. 1001, 1:41-48. A POSA reading the specification would understand that this term refers to transferring data from MLC to SLC, or from heavily used MLC to less used MLC, in order to prolong or otherwise boost the lifetime of the memory. Ex. 2001, ¶86. Specifically, Figs. 2A-2B and 3A-3B of the Challenged Patent show how the data is remapped to boost or prolong the lifetime of the system. If, for example, there is failure of a data integrity test, the data is transferred to a block with higher endurance. Ex. 1001, 5:60-63. Similarly, if a block is heavily used, the data is transferred to a block that is less used, or a higher-performance block. *Id.*, 3:26-33 (“selects an available target block with the lowest overall erase count”), 6:55-61 (“on a periodic basis, such as, for example, every 100 writes, or every 10,000 writes, transfer the contents of those blocks into the SLC NAND flash...”). In each case, the lifetime of the memory is maximized. *Id.*

Further, previous petitioner Micron’s IPR against the 300 Patent (of which the Challenged Patent is a continuation) further confirms this term is not indefinite.

When Micron’s expert was faced with this claim term (which also appears in claim 1 of the 300 Patent), he had no problem determining its claim scope and opined that it relates to rewriting data to more robust SLC cells to boost lifetime. *See* Ex. 2011, ¶85 (alleging in the “Technological Background” that “[i]n order to maximize the lifetime of devices, it was known to write frequently updated blocks...to SLC due to its higher erase cycle capacity and faster speed”), ¶¶183-186, 184 (citing the 300 Patent at 5:14-20 and 1:45-52, explaining that lifetime is increased by using MLC and SLC). Additionally, in its prior petition, Micron confirmed that this term does “not require further construction and can be afforded [its] plain and ordinary meaning.” Ex. 2016, 28.

Here, Petitioner Phison instead contends that this term is indefinite because claim 1 does not specify “what mapping is implicated” to maximize lifetime. Petition, 41-42. Petitioner’s argument, in effect, is that the claim is too broad unless a specific “strategy/configuration” of mapping is claimed. *Id.*, 41. Again, Petitioner mistakes breadth for indefiniteness. *See Niazi*, 30 F.4th at 1347. The claim does not need to be limited to a specific type of mapping to be patentable.

4. 3.D: “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. But

claim 1 never use the word “means,” and Petitioner cannot possibly overcome the presumption that § 112, ¶6 does not apply.⁶ Ex. 2001, ¶¶90-100.

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, 41. 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)

⁶ Petitioner references 35 U.S.C. § 112(f) but this is not the applicable statute. *See* Petition, 42. The Challenged Patent is a continuation of an application filed before September 16, 2012, and, therefore, contrary to Petitioner’s argument, the applicable statute is pre-AIA 35 U.S.C. § 112, ¶6.

(“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, 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, 2:43-46 (“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.”); 2:47-50 (“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)”); 3:11-17 (“wear leveling algorithms are used with the flash devices (as firmware commonly known as FTL or managed by a *controller*)”); 3:18-21 (“the *controller* maintains a lookup table”); 3:21-33 (“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, ¶95; 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, 3:6-7 (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, 42-44. 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, ¶¶99-100. A POSA would have understood that the specification discloses algorithms for the claimed functionality. Ex. 2001, ¶¶99-100; *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, 4:59-64; *see* Ex. 2001, ¶99; *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.”).

5. 3.E: 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, 44-46). Rather, they explain how the controller is configured to operate in the claimed system. Ex. 2001, ¶¶101-103. 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:...
- at least **one controller to operate** physical memory elements...
- wherein the at least **one controller**, or the FTL, or a combination of both **maintain** an address table...
- the controller controlling** access of the physical MLC and SLC nonvolatile memory elements and the random

access volatile physical memory elements for storage of
the received data therein...

the controller...operable to store the received data in the
given one of the physical MLC nonvolatile memory
elements as stored data and retain such received data in
the random access volatile physical memory...

the controller performing a data integrity test...after the
at least a Write access operation...

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, ¶102. 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”).

6. 3.F: Whether And When The Claimed Comparison Test Is Required

There is nothing ambiguous about term 3.F (“performing a data integrity test...after the at least a Write access operation”), and a POSA would have readily understood its scope. Ex. 2001, ¶104. The Boardy already considered—and rejected—Petitioner’s argument in denying institution of PGR2024-00047. Ex. 2039, 62; *see* Petition, 46.

Claim 1 makes clear that the “data integrity test” is performed “after...a Write access operation.” And the specification provides further insight “whether and when” a “data integrity test” may be performed. Petition, 46. For example, the data integrity test may be performed “after every write.” Ex. 1001, 5:10-11 (“The controller conducts a data integrity test after every write.”); *see also id.*, 4:27-29. Alternately, the data integrity test may be performed only when “time permits.” *Id.*, 5:57-60 (“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, ¶104. But all claim 1 requires is that the controller perform a data integrity test after “a Write access operation.”

Petitioner suggests that claim 1 does not provide sufficient guidance regarding “whether and when” the data integrity test must be performed and, accordingly, is somehow indefinite. Petition, 46. Petitioner suggests that the claim must explicitly state whether the data integrity test is performed after “*every* ‘Write Access.’” *Id.*, 45 (emphasis in original). 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.’”).

VIII. GROUND 4: 35 U.S.C. § 103

A. Overview Of The Cited Prior Art

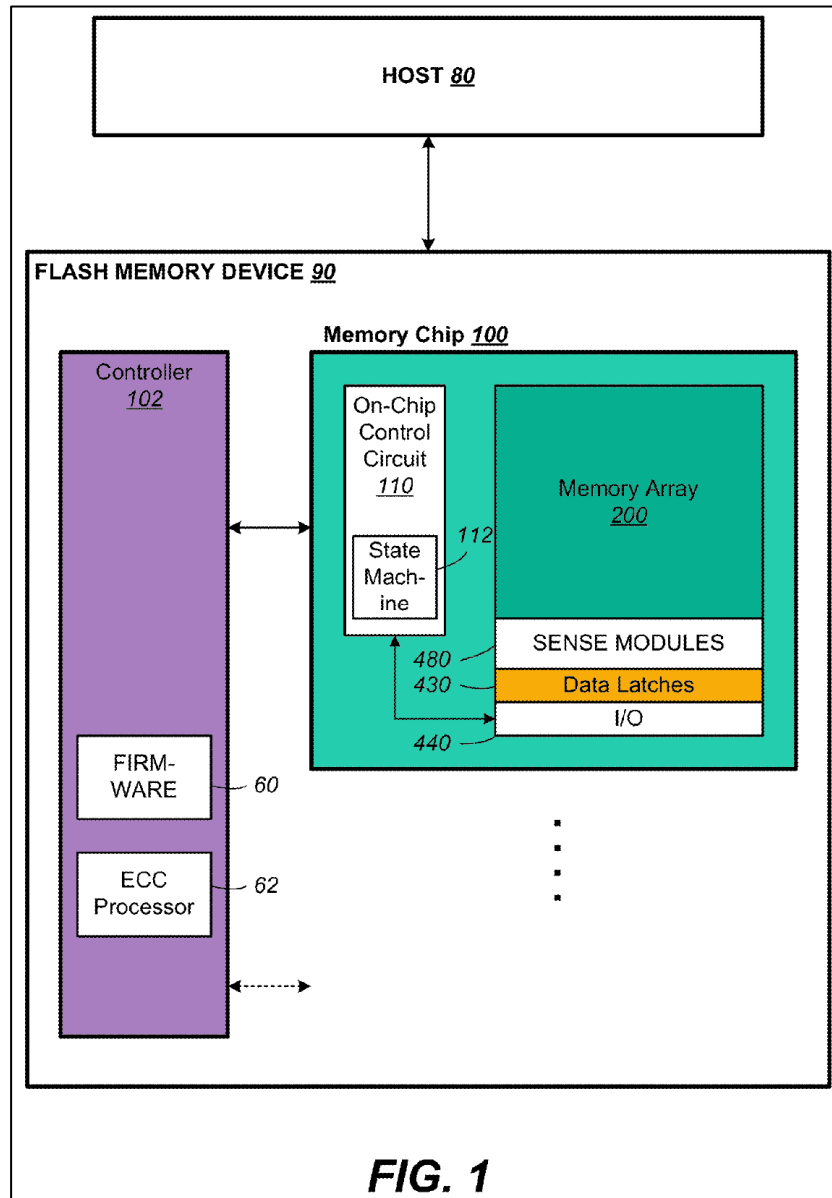
Petitioner asserts that the claims are obvious over Gavens (U.S. Patent No. 8,634,240) (Ex. 1045) in view of the knowledge of a POSA. Petition, 46-71.

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), U.S. Patent No. 5,936,971 (“Harari”) (Ex. 1057), and PCT Pub. No. WO 03/027828 (“Gorobets PCT”) (Ex. 1058). Petitioner also cites references unrelated to Gavens, including U.S. Patent Appl. Pub. No. 2009/0327591 (“Moshayedi”) (Ex. 1043).

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, ¶¶107-122. 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, ¶108. 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.*, 12:64-13:8.⁷

⁷ 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, ¶110 n.1.

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 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.

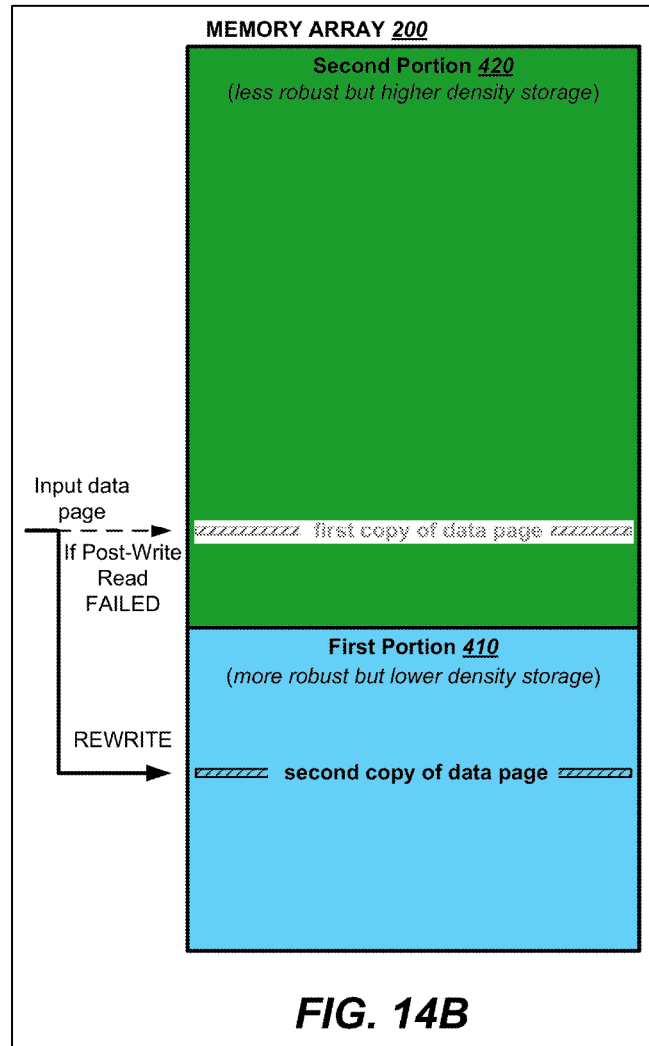
To address this problem, Gavens discloses the embodiments of Figures 14A-14B and 16A-16C, which describe the **memory array 200** of Figure 1 in greater detail. *See id.*, 8:13-15 (“Fig. 1 illustrates...a memory device in which the features of the present invention are embodied.”).

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, ¶¶114-116. 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, ¶114.

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.

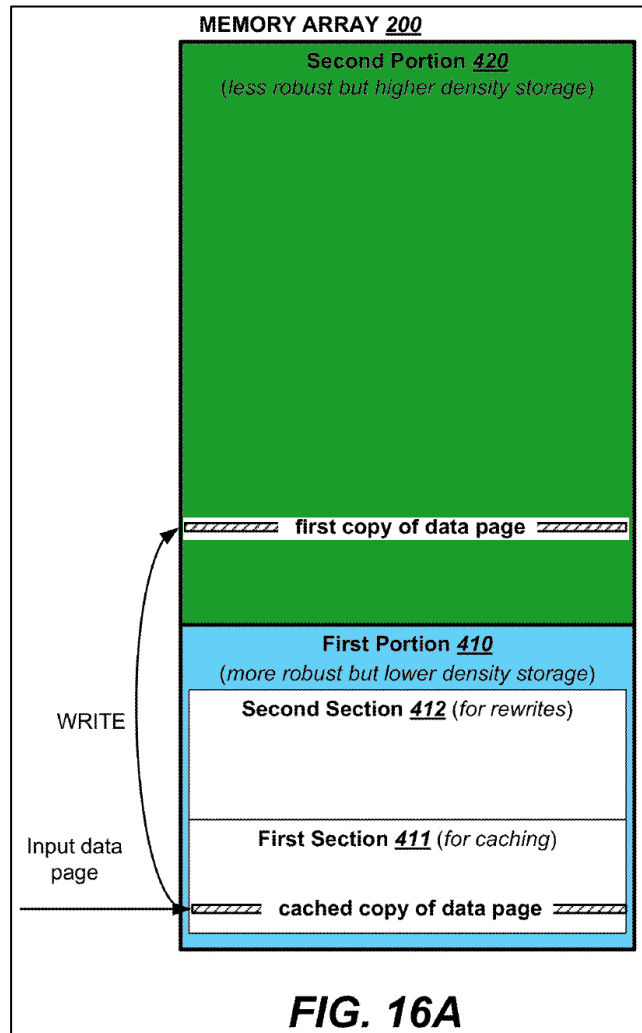
Once the error rate is checked, the device determines whether to keep the data in MLC or instead move it to SLC. If the stored data has an acceptable number of error bits, then the stored copy remains in the MLC and becomes the “valid” copy. *Id.*, 16:41-56. “If the error bits exceeded a predetermined 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, ¶¶117-122. 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, ¶124.

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.*

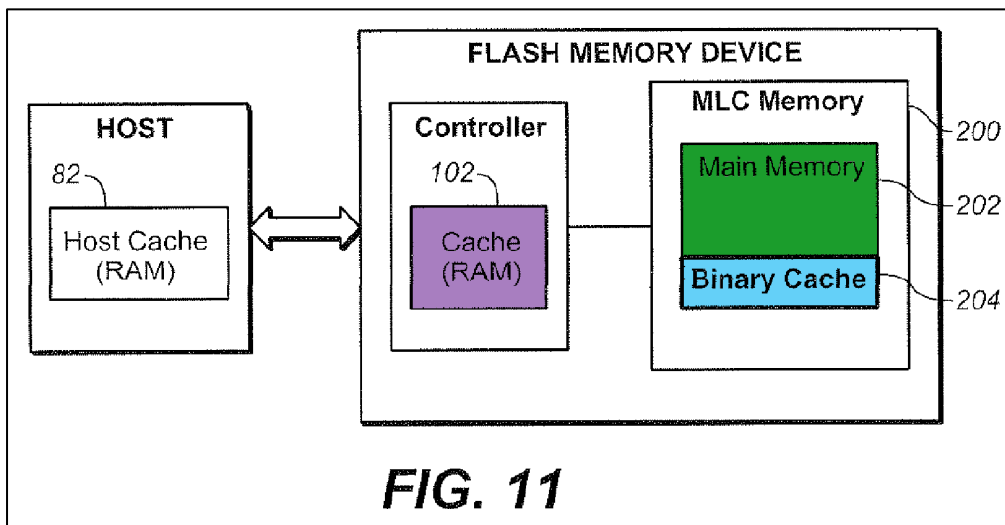
If there is an acceptable number of error bits, the MLC portion becomes the “valid” copy. *Id.*, 18:45-52, Fig. 16B. If there is an unacceptable number of error bits, the “cached copy” in the “first section” of the SLC is rewritten to the “second section” of the SLC. *Id.*, 18:53-67, Fig. 16C.

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*, ¶¶123-125.

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, ¶124.

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—in contrast to the Challenged Patent, data is not *retained* in volatile memory. *See id.*, ¶¶24, 109, 130, 144.

3. Paley (Ex. 1051)

Paley is closely related to Gorobets 912⁸ 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

⁸ 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, ¶¶126-128.

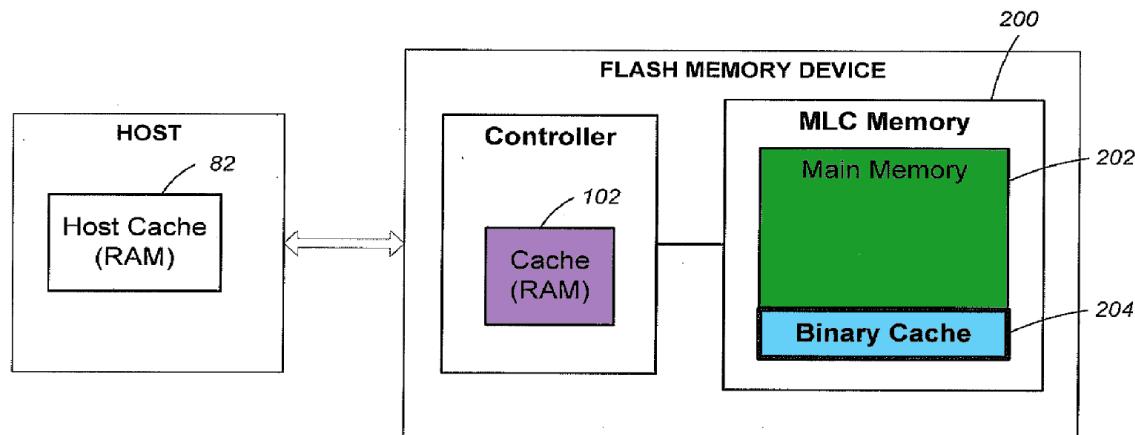


FIG. 15

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, ¶126.

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, unlike the Challenged Claims, both Gorobets 912 and Paley describe a memory system including a **RAM-based cache** that only *temporarily* stores data being passed between components. Ex. 2001, ¶¶127-128.

4. Harari (Ex. 1057)

Petitioner also cites Harari, which Gorobets 912 and Paley allegedly incorporate by reference according to Petitioner. Petition, 47-48 n.12. This assertion is false. Gorobets 912 and Paley merely state: “Using RAM in a write cache operating with a flash memory has been disclosed in U.S. Patent No. 5,936,971 to Harari et al.” Ex. 1051, ¶27; Ex. 1049, ¶24 (same). This is the only paragraph in which these references mention Harari. Neither reference clearly states that Harari 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).

That aside, Harari describes a RAM “cache buffer” “for *temporarily* holding write data files.” Ex. 1057, Fig. 8 (including “cache buffer 705”), 13:29-30 (emphasis added). It acts as a “buffer” because it holds data before it is written to a non-volatile memory. *See id.*, 14:1-38. When “the cache memory 705...start[s] to fill up[,]” it preferentially writes (“archive[s]”) infrequently written files to the flash non-volatile memory. *Id.*, 14:26-30. “In this way, only infrequently used data files are written into the Flash memory while frequently used data files are trapped in the

cache memory.” *Id.*, 14:22-25. Harari’s “cache buffer” therefore “reduce[s] the number of writes to the Flash...memory[.]” *Id.*, Abstract; *see also* Ex. 2001, ¶129.

5. Gorobets PCT (Ex. 1058)

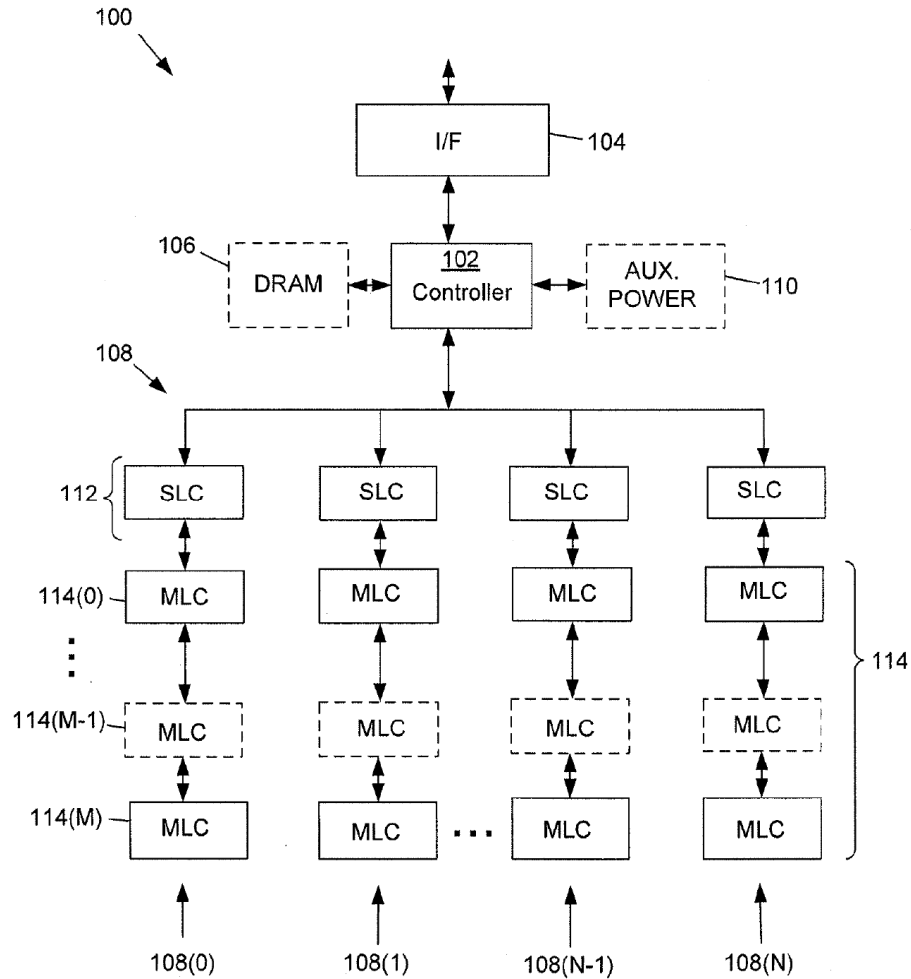
Petitioner also alleges that Gorobets 912 and Paley incorporate by reference Gorobets PCT. Petition, 47-48 n.12. This is not true. Gorobets 912 and Paley merely state—like with Harari—that “WO 03/027828 [(Gorobets PCT)] disclose[s] a memory system....” Ex. 1049, ¶19; Ex. 1051, ¶23 (same). These references do not state, with the required specificity, that Gorobets PCT is incorporated by reference. *See Callaway Golf*, 576 F.3d at 1331.

Notwithstanding, Gorobets PCT—just like Gorobets 912, Paley, and Harari—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, ¶130.

6. Moshayedi (Ex. 1043)

As shown in Figure 1 below, Moshayedi describes a flash memory having both SLC and MLC memory, with the described goals of “accommodat[ing] application memory needs at desirable prices, in addition to increasing read/write performance.” Ex. 1043, Abstract, ¶8. As is also shown in Figure 1 below,

Moshayedi's system includes "DRAM 106 [which] may be used to *buffer data* to be written into the flash array 108." *Id.*, ¶28; *see also id.*, Fig. 2 (including RAM 214, which is described in ¶¶36-37 as being used for "mapping" and a "buffer/list"); Ex. 2001, ¶¶131.



Ex. 1043, Fig. 1.

Thus, unlike the Challenged Claims, these tertiary references that Petitioner cites—Gorobets 912, Paley, Harari, Gorobets PCT, and Moshayedi—disclose

RAM-based caches that *temporarily* store data before it is written to flash. Ex. 2001, ¶132.

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

As discussed below, the Petition’s analysis is fatally flawed regarding multiple aspects of claim 1 of the Challenged Patent.

1. [1c]: “at least one *random access volatile physical memory element* that exists in the logical volatile memory space”

For limitation [1c], Petitioner fails to show how Gavens—alone or in combination with the numerous cited tertiary references (which are not even part of the unpatentability ground)—discloses or suggests the claimed “random access volatile physical memory element” (RAM). The Petition ambiguously mentions allegedly incorporated references such as Gorobets 912, Paley, Harari, and Gorobets PCT⁹, as well as unrelated references like Moshayedi, without clearly mapping any of them to this limitation. *See* Petition, 52-54. Gavens itself does not disclose RAM, and Petitioner fails to explain how the disparate RAM-related disclosures in the

⁹ As explained above in Sections VIII.A.4-VIII.A.5, Petitioner is incorrect in its assertion that Gorobets 912 and Paley incorporate by reference Harari and Gorobets PCT.

tertiary references apply to or could be combined with Gavens to satisfy the claim. See Ex. 2001, ¶¶134-138.

First, Petitioner does not allege that Gavens itself discloses the claimed RAM. Petition, 52-54. Nor can Petitioner do so, because Gavens simply does not disclose or suggest that its system includes RAM. 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 (Board 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’”).

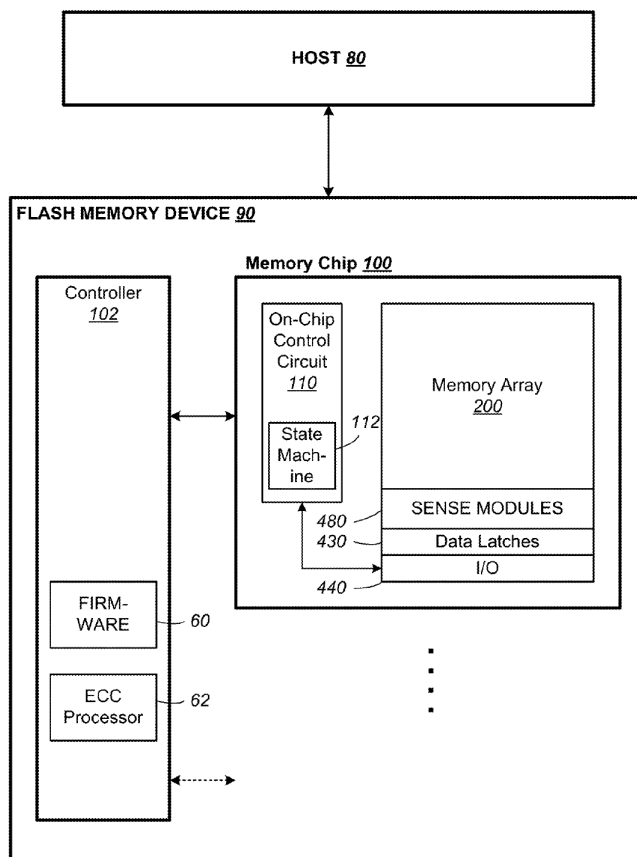


FIG. 1

Ex. 1045, Fig. 1; *see also* Ex. 2001, ¶135.

Second, apparently conceding that Gavens itself does not disclose the RAM required by limitation [1c], Petitioner points to disclosure from a multitude of disparate references—including Gorobets 912, Paley, Harari, Gorobets PCT, and Moshayedi. Petition, 52-54. Petitioner vaguely contends that these references’

distinct teachings support its claim that it was “well-known” “to use...RAM...in a NAND flash controller as a staging or cache location for data to be written to flash memory.” *Id.*, 52. 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, ¶136; *see also supra* Sections VIII.A.2-VIII.A.6 (summarizing the cited disclosure of these references). For one, as explained above, Gavens never discloses the use of RAM. Petitioner provides no explanation how the distinct disclosure of the allegedly incorporated references—let alone the references with no relation to Gavens, such as Moshayedi—apply to or 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”).¹⁰

¹⁰ Petitioner made the same error in PGR2024-00048, which challenged U.S. Patent No. 11,854,612—a parent patent of the 369 Patent challenged in this Petition. There, Petitioner cited Gorobets 912’s teaching of RAM for a similar limitation, and the

Finally, to the extent Petitioner is attempting to rely on an obviousness analysis for limitation [1c], that also fails.¹¹ Petitioner provides no coherent obviousness analysis. Petitioner does not explain which 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 state whether any of the tertiary references' RAM is being used to modify Gavens's system, and why a POSA would have been motivated to make any modification. *See* Petition, 52-54. 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). The Petition's obviousness analysis is insufficient and must be

Board found that, "As Patent owner correctly notes, 'Petitioner...does not state whether Gorobets [912]'s 'RAM' is being used to modify Gavens's system, and why a POSA would have made any modification.'" Ex. 2041, 43.

¹¹ Petitioner vaguely references what a "POSITA would...understand" based on the tertiary references but does not state that it is relying on an obviousness analysis for limitation [1c]. Petition, 53. Petitioner ambiguously concludes its ground with: "Gavens and incorporated references *disclose* or, in view of the POSITA's knowledge of NAND flash, claim element [1c]." Petition, 54 (emphasis added).

rejected; *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.”); Ex. 2001, ¶¶137-138; *see also* Ex. 2041, 43 (in denying institution of PGR2024-00048, the Board similarly determined that petitioner failed to provide the necessary obviousness analysis).

Thus, Petitioner has not demonstrated with particularity how the claimed “random access volatile physical memory element” of limitation [1c] is disclosed or suggested 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 the tertiary references without clearly identifying which *one* they rely on—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)).

2. [1d]: “at least one controller to operate physical memory elements and associated memory space for associated Write access operations to the physical memory elements and Read access operations from the physical memory elements, *the controller having a controller memory associated therewith* for storing received data therein”

The ambiguity and error in Petitioner’s mapping of limitation [1c] compounds further into its treatment of limitation [1d]. Here, Petitioner fails to identify with

particularity how Gavens discloses the claimed “controller memory.”¹² Although Petition references teachings from Gavens, Gorobets PCT, and Moshayedi, it provides no explanation how these distinct teachings map to—or can be combined to disclose—the “controller memory.” Petition, 54-55; Ex. 2001, ¶¶140-145.

Petitioner does not argue that Gavens teaches the claimed “controller memory.” Instead, Petitioner simply references the “controller 102” in Figure 1 of Gavens. Petition, 54. Petitioner then ambiguously cites *both* (1) the “in-controller RAVM for storing (caching) received data” purportedly taught by Gorobets PCT, and (2) the “off-controller DRAM for data caching” purportedly taught by Moshayedi. Petition, 54 (citing Ex. 1043 (Moshayedi), Fig. 1 (including DRAM 106), ¶28). This scattershot citation to both Gorobets PCT and Moshayedi fails for at least three reasons.

First, Petitioner does not identify what from Gorobets PCT and/or Moshayedi it is mapping to the “controller memory.” Petitioner cannot ambiguously cite things

¹² Petitioner does not rely on an obviousness analysis for limitation [1d]. Instead, Petitioner relies on what “Gavens and incorporated references *disclose*.” Petition, 54 (ambiguously concluding, “Gavens and incorporated references disclose or, in view of POSITA’s knowledge of NAND flash, claim element [1d].”) (emphasis added).

from both references and fail to map those things to the claims with the required particularity. *See* 35 U.S.C. § 322(a)(3); *Masimo*, IPR2023-00745, Paper 44, 36.

Second, Petitioner relies on the same things from Gorobets PCT and Moshayedi—the “in-controller RAVM” of Gorobets PCT and DRAM 106 of Moshayedi—for multiple claim elements. Petitioner appears to map both these things to the “controller memory” of limitation [1d]. Petition, 54. But for limitation [1c], ***Petitioner already relied on these things for the claimed “random access volatile physical memory element.”*** Petition, 53 (relying on the “controller SRAM buffer” of Gorobets PCT for the claimed RAM), 53 (*also* relying on “DRAM 106” of Moshayedi for the claimed RAM). Instead of providing a unique citation to Gorobets PCT for limitation [1d], Petitioner merely cites back to its ground for limitation [1c]. Petition, 54 (“As explained at [Section X(A)(3) *supra*],...Gorobets PCT...discloses...”). Thus, to the extent Petitioner is mapping ***both*** the claimed “controller memory” (limitation [1d]) and the “random access volatile physical memory element” (limitation [1c]) onto the same things in the prior art, this is improper. *See Becton, Dickinson & Co. v. Tyco Healthcare Grp., LP*, 616 F.3d 1249, 1254-55 (Fed. Cir. 2010); *see also* Ex. 2001, ¶¶141-143.

Third, Gavens, Gorobets PCT, and Moshayedi are different references, and describe different embodiments. Petitioner, however, provides no explanation how the distinct disclosure of Gorobets PCT and/or Moshayedi applies to or can be

combined with the previously cited disclosure of Gavens. Petition, 54; *see also* Ex. 2001, ¶144. Petitioner cannot cite, without explanation, the distinct disclosures of Gorobets PCT and Moshayedi as if they are part of the cited embodiment of Gavens. *See Captioncall*, No. IPR2020-01215, 2021 WL 278381, at *7.

Finally, the Petition includes no obviousness analysis for limitation [1d]. *See* Petition, 54. Still, to the extent Petitioner is attempting to rely on obviousness analysis for [1d], that also fails. Petitioner makes no mention of obviousness, and provides no basis or rationale for modifying Gavens’s system to arrive at limitation [1d].

Accordingly, Petitioner fails to demonstrate that Gavens discloses or suggests the “controller memory” of limitation [1d]. Petitioner’s ground improperly relies on the same things in the prior art for multiple claim elements and, notwithstanding, lacks any explanation how the distinct teachings of Gavens, Gorobets PCT, and Moshayedi could have been combined.

3. [1f]: “the controller...[1f3] operable to store the received data in the given one of the physical MLC nonvolatile memory elements as stored data and retain such received data in the random access volatile physical memory element as retained data associated with stored data”

Petitioner’s obviousness analysis for limitation [1f] is deficient at least because Petitioner fails to demonstrate how Gavens and the cited tertiary references (which are not even part of the unpatentability ground) disclose RAM that “retain[s]

such received data.” Petition, 59-60. Here, Petitioner asserts that Gavens’s disclosure of retaining an “original copy which may *cached*” (Ex. 1045, 16:46-49 (emphasis added)) would have been “*assumed*” to refer to a RAM-based cache. Petition, 60 (referring to “Gavens error management embodiment by ‘storing’ data in the MLC [1f3] while operating the system to maintain the ‘original copy which may be cached’ (16:46-49) as the ‘retained data’ in the *assumed, common on-controller or associated external RAVM* (or both)[.]” (cleaned up and emphasis added)). But, again, Petitioner provides no obviousness analysis to support this *assumption*. Gavens itself does not disclose that this “cache” is implemented in RAM. *See* Ex. 1045, 16:46-49. And a POSA would have instead understood that Gavens’s “cache[.]” that retains the “original copy” (Ex. 1045, 16:46-49) is a *non-volatile cache*. *See also* Ex. 1039, 77 (in denying institution of PGR2024-00047, agreeing that “‘the first portion’ [of Gavens] that ‘serves as a cache for incoming data’ is non-volatile memory and does not correspond to the claimed ‘random access volatile memory.’”).¹³ And the tertiary references Petitioner cites do not prove otherwise. *See* Ex. 2001, ¶¶146-159.

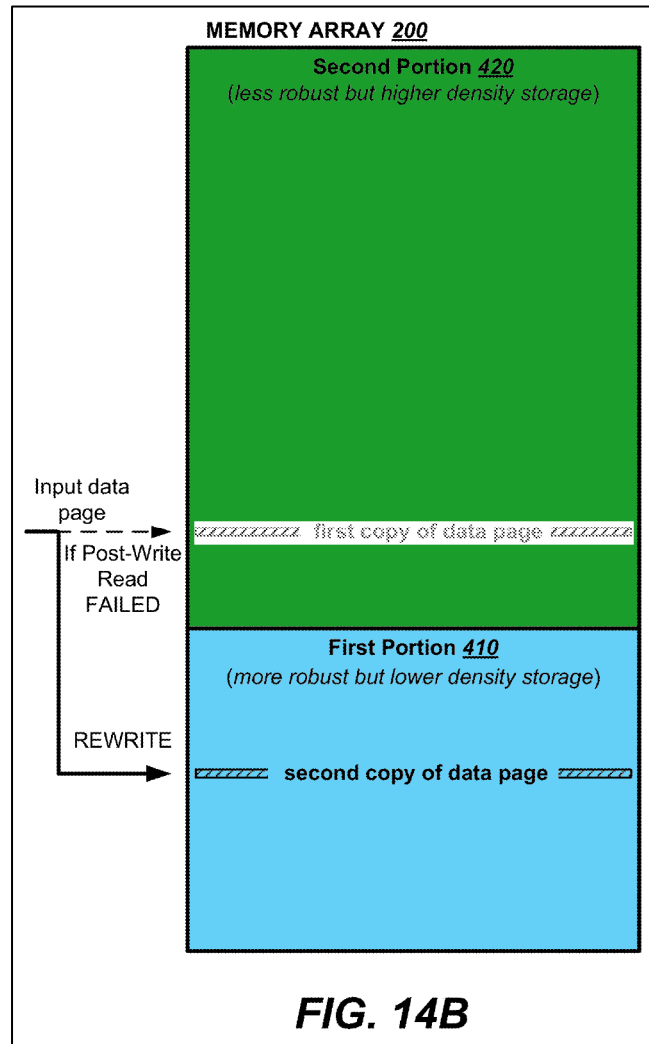
¹³ Petitioner relies on this same assumption for limitation [1g] and, similarly, fails to provide the required obviousness analysis. *See* Petition, 61 (conclusorily asserting that Gavens’s “original copy which may be cached” (Ex. 1045, 16:46-49) “may be

First, Petitioner cannot show that Gavens itself discloses a RAM-based cache used in 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 VIII.A.1; *see also* Ex. 2001, ¶¶147-148.

Specifically, for its ground, Petitioner cites teachings from the “first embodiment” of Gavens, including Figure 14; an overview of this embodiment is provided in Section VIII.A.1.a above. *See, e.g.*, Petition, 60 (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

in the RAVM of the controller memory or associated DRAM”). Accordingly, Petitioner’s analysis for limitation [1g] fails for the same reasons as those described in this Preliminary Response for limitation [1f].

cited by Petitioner (*see* Petition, 60), refers to this post-write error check of data written to the MLC portion.



Ex. 1045, Fig. 14B (annotated); *see also* Ex. 2001, ¶148.

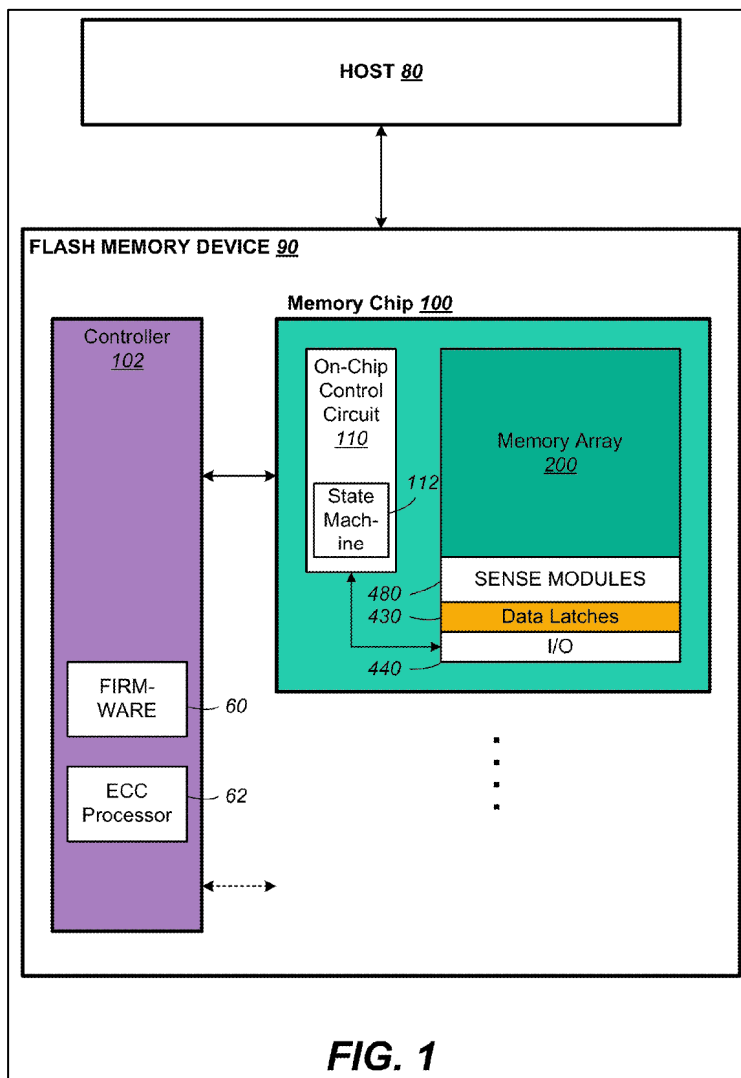
Gavens at 16:46-49 specifies that the post-write error check may be performed “by [a] comparison with the original copy which may be cached[.]” Ex. 1045, 16:46-49. At 16:46-49, Gavens does not specify how the “original copy” is cached, but Gavens later explains that “[i]n the preferred embodiment” (where 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; *see also id.*, 16:14-16. Thus, Gavens describes where the “original copy” is “cached” in non-volatile memory—not volatile memory, such as RAM, as Petitioner appears to contend. Ex. 2001, ¶149.

Additionally, Gavens further 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 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 VIII.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” (*i.e.*, the MLC 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; *see also* Ex. 2001, ¶150.

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 (the comparison)—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, ¶151.



Ex. 1045, Fig. 1 (annotated).

Thus, a POSA would not have understood Gavens itself to teach caching data in RAM, contrary to Petitioner's assumption. Gavens instead teaches that data is cached in the non-volatile SLC memory. Ex. 2001, ¶152.

Second, the teachings of the tertiary references—such as Paley (the only tertiary reference cited by Petitioner in its ground for limitation [1f] (*see* Petition, 59)) and Gorobets 912 (which, as explained in Section VIII.A.3 *supra*, is similar to

Paley)—do not justify Petitioner’s assumption regarding Gavens. *See also* Petition, 59-60 (for limitation [1f], Petitioner generally refers to its ground for limitation [1c] (“[Section X(A)(3)]”) which, as explained above in Section VIII.B.1, includes a scattershot citation to more than five tertiary references related to RAM); *see also* Ex. 2001, ¶¶153-158.

Petitioner’s ground for limitation [1f] relies only on what Gavens “in view of POSITA knowledge” ***discloses***—and Petitioner does not purport to rely on obviousness for this limitation. Petition, 60 (“Gavens thus discloses, in view of POSITA knowledge, claim limitation [1f].”). Petitioner, however, again cites the distinct teachings of Paley to argue how Petitioner believes a POSA would have “understood” Gavens. *See* Petition, 59. But, as was also addressed in countering Petitioner’s ground for limitation [1c] above, Petitioner provides no explanation how the unrelated disclosure of Paley applies to or can be combined with the previously cited disclosure of Gavens. Petitioner cannot cite the RAM-based cache taught by Paley—or any of the other tertiary references—as if they are part of Petitioner’s cited embodiment of Gavens. *See Captioncall*, 2021 WL 278381, at *7.

Notwithstanding, to the extent Petitioner was attempting to invoke obviousness—which it did not—Petitioner again fails to provide the requisite obviousness analysis to explain, for example, why Gavens would have been modified or re-interpreted based on the teachings of tertiary references (such as

Paley or Gorobets 912) to rely on a RAM-based cache for retaining the “original copy” (Ex. 1045, 16:46-49).

A POSA would have instead understood that Gavens’s non-volatile cache *should not* be replaced with the RAM-based caches taught by the tertiary references, such as Paley and Gorobets 912. Ex. 2001, ¶155. To start, unlike the “cache” of Gavens, which retains the “original data,” the tertiary references cited by Petitioner disclose RAM-based caches for buffering data temporarily.

For example, Petitioner cites ¶197 of Paley (*see* Petition, 38 (citing ¶197 of Paley)) but, in this paragraph, Paley explicitly states: “A cache is high-speed storage for temporarily storing data being passed between a high-speed and a slower-speed component of the system.” Ex. 1051, ¶197. Paley further suggests that a “high-speed volatile RAM” in a controller can be used as a cache to temporarily store data being passed to non-volatile memory, including SLC or MLC. *See* Ex. 1051, ¶197; Ex. 2001, ¶156.

Gorobets 912 includes the same disclosure. *See* Ex. 1049, ¶109 (“A Cache is high-speed storage for temporarily storing data being passed between a high-speed and a slower-speed component of the system.”). Gorobets 912 goes on to explain that this RAM, by design, is of “relatively limited capacity,” so once data is “transferred out” to the non-volatile memory, the contents of the RAM are cleared to “receive more data” being transferred—in other words, this RAM is not operable

to retain data that is written to non-volatile memory. *See* Ex. 1049, ¶144 (“As data has been transferred out of RAM at 721, this has opened up the RAM, which is relatively limited capacity, to receive more data from the host....”); Ex. 2001, ¶156.

Additionally, the Board already considered Gorobets 912 in denying institution of a related petition filed by Petitioner—PGR2024-00047, challenging U.S. Patent No. 11,830,546, which is a continuation of the 369 Patent challenged here—and determined, congruent with PO’s current arguments, that “Gorobets [912] ...relate[s] to using RAM to *temporarily* store data *before transferring the data* to non-volatile memory[.]” Ex. 2039, 78 (emphasis added).

Thus, Paley and Gorobets 912 make clear—contrary to Petitioner’s *assumption*—that their RAM-based caches only “temporarily” stores data when it is being “passed” to the non-volatile memory. Harari, Gorobets PCT, and Moshayedi—which are not explicitly cited in Petitioner’s ground for limitation [1f]—suffer from the same flaw and similarly disclose RAM-based caches/buffers for *temporarily* storing data. *See* Ex. 1057, Fig. 8, 13:29-30; Ex. 1058, 11:10-13; Ex. 1043, ¶28 *see also supra* Sections VIII.A.4-VIII.A.6 (describing the tertiary references’ disclosure of RAM-based caches for *temporarily* storing data); Ex. 2001, ¶157.

Accordingly, even assuming *arguendo* that a POSA would have combined a RAM-based cache for temporarily storing data—purportedly taught by each of

tertiary references—into Gavens’ cited system, Petitioner fails to demonstrate why a POSA would have understood the controller to be operable to *retain* data in that RAM when it is stored in MLC, as required by limitation [1f]. Ex. 2001, ¶¶153-159; *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]”).

Thus, Petitioner fails to show that Gavens discloses or suggests the features of limitation [1f].

4. [1g]: “the controller performing a data integrity test...and [1g3] *comparing the stored data in the controller memory in the given one of the physical MLC nonvolatile memory elements to the retained data that was associated with the stored data in the random access volatile physical memory element by the controller during the Write access operation*”

For limitation [1g], Petitioner again conclusorily assumes that the “cache” for retaining the “original copy” taught by Gavens would be a RAM-based cache. *See* Petition, 61 (asserting that Gavens “check[s] for errors...‘by comparison with the original copy which may cached’...which may be in the RAVM of the controller memory or associated DRAM” (quoting Ex. 1045, 16:46-49)). As explained above in Section VIII.B.3 (limitation [1f]), Gavens discloses only a non-volatile cache for

retaining the “original data,” and Petitioner fails to demonstrate why that non-volatile cache would instead be a volatile RAM-based cache. *See also* Ex. 2001, ¶¶160-164.

Notwithstanding that error—which is alone dispositive—Petitioner’s ground for limitation [1g] is also deficient because this limitation further requires that the “**controller**” compare stored data in MLC with retained data in RAM to determine data integrity (*i.e.*, to check for errors). Petitioner conclusorily contends that “[i]t would be obvious that the ‘stored data’ to be compared would be written to the controller memory...for comparison *using logic on the controller chip*[.]” Petition, 62 (emphasis added). In addition to the fact that the Petitioner provides no support for what it contends “would be obvious”—again, for example, providing no substantive explanation why a POSA would have come to this understanding—Petitioner’s contention is incorrect. A POSA would not have found it obvious that the Gavens’s “comparison” of the “original copy” would be performed “using logic on the controller chip” because *Gavens’s non-volatile cache is intentionally not in the controller to avoid using the controller for the comparison*. Ex. 2001, ¶162.

Specifically, as explained above, the non-volatile 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. Gavens teaches *against* using the controller for this post-write error-check. As explained in Sections VIII.A.1 (Gavens

background) and VIII.B.3 (limitation [1f]), Gavens explains that the data used for its comparison 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.”). Instead, Gavens uses *on-chip data latches* for the data comparison. Ex. 1045, 18:36-67. 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 the controller for the comparison (as required by Petitioner’s ground) would contradict the purpose of Gavens’s invention. Ex. 2001, ¶163.

Petitioner provides no rationale to overcome this clear teaching away. Thus, to the extent Petitioner presented an obviousness ground—which it did not—relying on the controller for Gavens’s comparison “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 810, 813 (C.C.P.A. 1959); *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 similar 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”).

Thus, Petitioner fails to show that Gavens discloses or suggests the features of limitation [1g].

C. The Petition Does Not Establish That Dependent Claims 2-10 Would Have Been Obvious Over Gavens And/Or Obvious Over Gavens In View Of Colligan

Petitioner’s challenges to dependent claims 2-10 also lack merit, at least because these claims depend from claim 1, for which Petitioner’s ground is deficient for the reasons stated above.

IX. 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 16,205 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/
Alan Whitehurst, Reg. No. 43,263

Dated: May 14, 2025