

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICRON TECHNOLOGY, INC., MICRON SEMICONDUCTOR
PRODUCTS, INC., MICRON TECHNOLOGY TEXAS LLC,
DELL TECHNOLOGIES INC., DELL INC., and HP INC.,
Petitioner,

v.

UNIFICATION TECHNOLOGIES LLC,
Patent Owner.

IPR2021-00344
Patent 8,762,658 B2

Before JUSTIN T. ARBES, TERRENCE W. McMILLIN, and
CHRISTOPHER L. OGDEN, *Administrative Patent Judges*.¹

ARBES, *Administrative Patent Judge*.

JUDGMENT
Final Written Decision
Determining Some Challenged Claims Unpatentable
35 U.S.C. § 318(a)

¹ Katherine K. Vidal, Under Secretary of Commerce for Intellectual Property and Director of the United States Patent and Trademark Office (Director), is recused from this proceeding and took no part in this decision. See Director's Memorandum, Procedures for Recusal to Avoid Conflicts of Interest and Delegations of Authority (Apr. 20, 2022) (Recusal Procedure Memo), *available at* <https://go.usa.gov/xJjch>; Interim Process for Director Review (§ 20), *available at* <https://go.usa.gov/xJjce>.

I. INTRODUCTION

A. Background and Summary

Petitioners Micron Technology, Inc., Micron Semiconductor Products, Inc., Micron Technology Texas LLC, Dell Technologies Inc., Dell Inc., and HP Inc. (collectively, “Petitioner”) filed a Petition (Paper 4, “Pet.”) requesting *inter partes* review of claims 1–5, 8–12, and 22–26 of U.S. Patent No. 8,762,658 B2 (Ex. 1001, “the ’658 patent”) pursuant to 35 U.S.C. § 311(a). On July 9, 2021, we instituted an *inter partes* review as to all challenged claims on all grounds of unpatentability asserted in the Petition. Paper 9 (“Decision on Institution” or “Dec. on Inst.”). Patent Owner Unification Technologies LLC subsequently filed a Patent Owner Response (Paper 21, “PO Resp.”), Petitioner filed a Reply (Paper 28, “Reply”), and Patent Owner filed a Sur-Reply (Paper 34, “Sur-Reply”). An oral hearing was held on April 13, 2022, and a transcript of the hearing is included in the record (Paper 37, “Tr.”).

We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a). For the reasons that follow, we determine that Petitioner has shown by a preponderance of the evidence that claims 1–5 and 8–12 of the ’658 patent are unpatentable, and we cannot reach a decision on the merits with respect to whether Petitioner has established the unpatentability of claims 22–26.

B. Related Matters

The parties indicate that the ’658 patent is the subject of the following district court cases: *Unification Technologies LLC v. Dell Technologies, Inc.*, Case No. 6:20-cv-499-ADA (W.D. Tex.), *Unification Technologies LLC v. HP Inc.*, Case No. 6:20-cv-501-ADA (W.D. Tex.), and *Unification*

Technologies LLC v. Micron Technology, Inc., Case No. 6:20-cv-500-ADA (W.D. Tex.) (“the district court case”). Pet. 66; Paper 6, 2–3. Petitioner also filed petitions challenging claims of patents related to the ’658 patent in Cases IPR2021-00343 and IPR2021-00345.

C. The ’658 Patent

The ’658 patent discloses techniques for “managing data in a storage device using an empty data segment directive.” Ex. 1001, col. 1, ll. 25–27. “Typically, when data is no longer useful it may be erased. In many file systems, an erase command deletes a directory entry in the file system while leaving the data in place in the storage device containing the data,” such that the storage device is unaware that the data is now invalid. *Id.* at col. 1, ll. 29–33. “Another method of erasing data is to write zeros, ones, or some other null data character to the data storage device to actually replace the erased file,” but doing so is inefficient because “valuable bandwidth is used while transmitting the data” and “space in the storage device is taken up by the data used to overwrite invalid data.” *Id.* at col. 1, ll. 33–39. The ’658 patent attempts to overcome these issues by having the storage device “receive a directive that data is to be erased” and store a “data segment token” that represents erased data, rather than performing either of the typical erase methods. *Id.* at col. 1, ll. 60–65.

Figure 1A of the '658 patent is reproduced below.

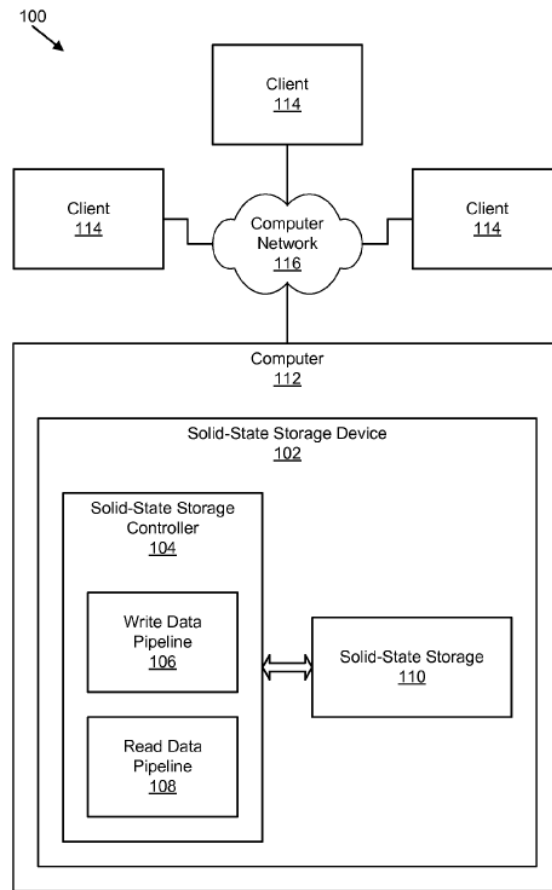


FIG. 1A

Figure 1A depicts clients 114 in communication over computer network 116 with computer 112 having solid-state storage device 102.² *Id.* at col. 7, ll. 29–36. Solid-state storage device 102 comprises solid-state storage 110 (e.g., flash memory) and solid-state storage controller 104 for writing to solid-state storage 110 (via write data pipeline 106), reading from solid-state storage 110 (via read data pipeline 108), and performing other operations on

² A solid-state storage device is a type of non-volatile memory that stores data in pages within blocks, where each page is identified by a unique physical address. Data in a solid-state storage device cannot be directly overwritten with new data, but instead must first be erased (at the block level) and then written (to pages). *See* Ex. 1001, col. 1, ll. 40–47.

solid-state storage 110. *Id.* at col. 7, ll. 37–44. When a “data packet is stored and the physical address of the data packet is assigned,” the solid-state storage controller creates an entry in an index that maps a “logical identifier” of the object to “one or more physical addresses corresponding to where the storage controller” stored the data packet and any object metadata packets. *Id.* at col. 12, ll. 38–44.

Write data pipeline 106 includes garbage collector bypass 316 that “receives data segments from the read data pipeline 108 as part of a data bypass in a garbage collection system.” *Id.* at col. 27, l. 65–col. 28, l. 1, Fig. 3. According to the ’658 patent,

[a] garbage collection system typically marks packets that are no longer valid, typically because the packet is marked for deletion or has been modified and the modified data is stored in a different location. At some point, the garbage collection system determines that a particular section of storage may be recovered. This determination may be due to a lack of available storage capacity, the percentage of data marked as invalid reaching a threshold, a consolidation of valid data, an error detection rate for that section of storage reaching a threshold, or improving performance based on data distribution, etc. Numerous factors may be considered by a garbage collection algorithm to determine when a section of storage is to be recovered.

Id. at col. 28, ll. 1–13. The ’658 patent discloses an apparatus comprising a “request receiver module” and “storage module.” *Id.* at col. 2, l. 11–col. 4, l. 45.

D. Illustrative Claim

Challenged claims 1 and 22 of the '658 patent are independent. Claims 2–5 and 8–12 depend from claim 1 and claims 23–26 depend from claim 22. Claim 1 recites:

1. An apparatus for managing data stored on a non-volatile storage medium, comprising:

a non-volatile storage medium;

a request receiver module configured to receive a message comprising a logical identifier, the message indicating that data associated with the logical identifier has been erased, wherein the logical identifier is mapped to a physical storage location of the non-volatile storage medium; and

a storage module configured to store persistent data on the non-volatile storage medium in response to the indication, wherein the persistent data is configured to indicate that the data associated with the logical identifier is erased.

E. Evidence

The pending grounds of unpatentability in the instant *inter partes* review are based on the following prior art:

U.S. Patent No. 7,624,239 B2, filed Nov. 14, 2005, issued Nov. 24, 2009 (Ex. 1002, “Bennett”);

U.S. Patent No. 7,057,942 B2, issued June 6, 2006 (Ex. 1003, “Suda”); and

Eran Gal & Sivan Toledo, “Mapping Structures for Flash Memories: Techniques and Open Problems,” *Proceedings of the IEEE International Conference on Software – Science, Technology & Engineering (SwSTE’05)*, Aug. 2005 (Ex. 1010, “SwSTE’05”).

Petitioner filed a declaration from R. Jacob Baker, Ph.D., P.E. (Ex. 1004) with its Petition. Patent Owner filed a declaration from Vijay K. Madiseti, Ph.D. (Ex. 2011) with its Response. Also submitted as evidence are

transcripts of the depositions of Dr. Baker (Ex. 2013) and Dr. Madiseti (Ex. 1038).

F. Asserted Grounds

This *inter partes* review involves the following grounds of unpatentability:

Claims Challenged	35 U.S.C. §	Reference(s)/Basis
1–5, 8–12, 22–26	103(a) ³	Bennett ⁴

³ The Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) (“AIA”), amended 35 U.S.C. §§ 103 and 112. Because the challenged claims of the ’658 patent have an effective filing date before the effective date of the applicable AIA amendments, we refer to the pre-AIA versions of 35 U.S.C. §§ 103 and 112. *See* Pet. 4.

⁴ Petitioner asserts that the challenged claims are unpatentable over (1) Bennett, (2) Suda, and (3) Suda and SwSTE’05, each “in view of a [person of ordinary skill in the art’s] knowledge.” Pet. 8–9. As explained in the Decision on Institution, we do not include the general knowledge of a person of ordinary skill in the art in listing the grounds themselves, recognizing that such knowledge is considered in every obviousness analysis. *See* Dec. on Inst. 6 n.3; 35 U.S.C. § 311(b) (*inter partes* review “only on the basis of prior art consisting of patents or printed publications”); *Koninklijke Philips N.V. v. Google LLC*, 948 F.3d 1330, 1337 (Fed. Cir. 2020) (“Although the prior art that can be considered in *inter partes* reviews is limited to patents and printed publications, it does not follow that we ignore the skilled artisan’s knowledge when determining whether it would have been obvious to modify the prior art. . . . Regardless of the tribunal, the inquiry into whether any ‘differences’ between the invention and the prior art would have rendered the invention obvious to a skilled artisan necessarily depends on such artisan’s knowledge.”); *Randall Mfg. v. Rea*, 733 F.3d 1355, 1362 (Fed. Cir. 2013) (“[T]he knowledge of [an ordinarily skilled] artisan is part of the store of public knowledge that must be consulted when considering whether a claimed invention would have been obvious.”); *Dow Jones & Co. v. Abblaise Ltd.*, 606 F.3d 1338, 1349 (Fed. Cir. 2010) (“[The obviousness] analysis requires an assessment of the . . . ‘background

Claims Challenged	35 U.S.C. §	Reference(s)/Basis
1–5, 8–12, 22–26	103(a)	Suda
2–5, 10–12, 23, 25	103(a)	Suda, SwSTE’05

II. ANALYSIS

A. Level of Ordinary Skill in the Art

In determining the level of ordinary skill in the art for a challenged patent, we look to “1) the types of problems encountered in the art; 2) the prior art solutions to those problems; 3) the rapidity with which innovations are made; 4) the sophistication of the technology; and 5) the educational level of active workers in the field.” *Ruiz v. A.B. Chance Co.*, 234 F.3d 654, 666–667 (Fed. Cir. 2000). “Not all such factors may be present in every case, and one or more of them may predominate.” *Id.*

Petitioner states that it assumes an effective filing date of December 6, 2006, for the challenged claims of the ’658 patent, and argues that a person of ordinary skill in the art at that time would have had “a Bachelor of Science degree in computer science or electrical engineering and at least two years of experience in the design, development, implementation, or management of solid-state memory devices.” Pet. 4–5 (citing Ex. 1004 ¶ 56). According to Petitioner, an ordinarily skilled artisan also

would have known, as background information: how flash memory erases data, how flash memory programs or writes data, how memory is used in a cache hierarchy, relative speeds of flash memory compared to other memory, how garbage collection is used with flash memory, how to use wear leveling to combat endurance limits of flash memory, how the [Flash Translation Layer (“FTL”)] works, and industry standards affecting flash

knowledge possessed by a person having ordinary skill in the art.” (citing *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 401 (2007))).

memory including the [Advance Technology Attachment (“ATA”)] standard.

Id. at 5 (citing Ex. 1004 ¶¶ 57, 61). Patent Owner applies the same definition of the level of ordinary skill in the art. PO Resp. 17–18 (citing Ex. 2011 ¶¶ 47–52). Based on the full record developed during trial, including our review of the ’658 patent and the types of problems and solutions described in the ’658 patent and cited prior art, we agree with Petitioner’s proposed definition of the level of ordinary skill in the art and apply it for purposes of this Decision. *See, e.g.*, Ex. 1001, col. 1, ll. 22–56 (describing in the “Background of the Invention” section of the ’658 patent various write, read, and erase procedures for solid-state storage devices).

B. Claim Interpretation

We interpret the claims of the challenged patent

using the same claim construction standard that would be used to construe the [claims] in a civil action under 35 U.S.C. 282(b), including construing the [claims] in accordance with the ordinary and customary meaning of such [claims] as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

37 C.F.R. § 42.100(b) (2020). “In determining the meaning of [a] disputed claim limitation, we look principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 469 F.3d 1005, 1014 (Fed. Cir. 2006). Claim terms are given their plain and ordinary meaning as would be understood by a person of ordinary skill in the art at the time of the invention and in the context of the entire patent disclosure. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005) (en banc). “There are only two exceptions to

this general rule: 1) when a patentee sets out a definition and acts as his own lexicographer, or 2) when the patentee disavows the full scope of a claim term either in the specification or during prosecution.” *Thorner v. Sony Comput. Entm’t Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012).

After the Petition was filed, the district court issued a Claim Construction Order construing various terms of the ’658 patent:

Claim Term	Construction
“logical identifier”	An identifier that can be associated with a physical address on a storage device for identifying data stored at the physical address
“logical identifier [that/in the index] is empty”	Indefinite
“data associated with the logical identifier [has been/is] erased”	Not indefinite; plain and ordinary meaning
“marking module”	Not indefinite; not subject to §112(f); plain and ordinary meaning
“storage module”	Not indefinite; not subject to §112(f); plain and ordinary meaning
“instructions configured to . . . recording persistent data . . . in response the indication”	Not indefinite; not subject to §112(f); plain and ordinary meaning

Ex. 2006, 2–3.

We address three claim interpretation issues. First, in the Decision on Institution, based on the record at the time, we agreed with and adopted the district court’s construction of “logical identifier.” Dec. on Inst. 11–12. The parties agree with that construction. *See* PO Resp. 19; Reply 3. Based on the full trial record, we interpret “logical identifier” in the same manner as the district court.

Second, claim 22 recites “receiving an indication, comprising a logical identifier, that the logical identifier mapped to a physical storage location comprising data associated with the logical identifier in the index *is empty*” (emphasis added). Claim 25 recites “determining that the logical identifier *is empty* in response to the persistent data recorded on the non-volatile storage device while the data associated with the logical identifier remains on the physical storage location” and claim 26 recites “responding to a request pertaining to the logical identifier with an indication the logical identifier *is empty* while the data associated with the logical identifier remains on the physical storage location” (emphasis added). The district court’s Claim Construction Order lists “[i]ndefinite” as the final construction for the phrase “logical identifier [that/in the index] is empty.” Ex. 2006, 2 (alteration in original). Petitioner asserted in the Petition that we should instead adopt Patent Owner’s original proposed construction from the district court case of “data identified by the [logical identifier] that does not need to be preserved,” but Petitioner did not provide any explanation for why such an interpretation is justified. Pet. 6–7 (alteration in original). In the Decision on Institution, based on the record at the time, we explained why we were “unable to interpret the ‘empty’ phrases in claims 22, 25, and 26, and c[ould] not ascertain the scope of the claims with reasonable certainty for purposes of assessing patentability.” Dec. on Inst. 12–13. For example, we noted that, given the interpretation of “logical identifier” above, a logical identifier is simply “information identifying something else” and “either exists or does not exist”; it cannot be considered “empty.” *Id.* at 13. We encouraged the parties, to the extent they disagreed with that determination, to “provide an explanation and cite evidence in support of the proposed interpretation in their papers during trial.” *Id.*

Patent Owner argues that, based on our earlier determination, Petitioner has failed to prove that claims 22–26 are unpatentable. PO Resp. 25, 46, 50. Petitioner does not address claims 22, 25, and 26 in its Reply. Neither party provides any argument or evidence disputing our reasoning in the Decision on Institution or the district court’s conclusion that the claim phrases are indefinite. Based on the full trial record, we maintain our earlier determination and incorporate the previous analysis herein. *See* Dec. on Inst. 12–13.

Third, with respect to the “module” and “instructions” terms listed above, neither party argues that the terms are means-plus-function limitations under 35 U.S.C. § 112, sixth paragraph, and the district court construed them not to be means-plus-function limitations. *See* PO Resp. 18–19; Reply 3–4; Ex. 2006, 2–3. We presume that the terms are not means-plus-function limitations because they do not use the word “means” and find no basis on the record before us to conclude otherwise. *See Dyfan, LLC v. Target Corp.*, 28 F.4th 1360, 1365 (Fed. Cir. 2022) (“Because invoking § 112 ¶ 6 is typically a choice left to the claim drafter, we presume at the first step of the analysis that a claim limitation is subject to § 112 ¶ 6 when the claim language includes the term ‘means.’ . . . The inverse is also true—we presume that a claim limitation is not drafted in means-plus-function format in the absence of the term ‘means.’ . . . [T]his presumption is rebuttable [and] can be overcome if a challenger demonstrates that the claim term ‘fails to recite sufficiently definite structure.’”) (citing *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1348–49 (Fed. Cir. 2015)).

No other claim terms require interpretation to decide the issues presented during trial. *See Nidec Motor Corp. v. Zhongshan Broad Ocean*

Motor Co., 868 F.3d 1013, 1017 (Fed. Cir. 2017) (“Because we need only construe terms ‘that are in controversy, and only to the extent necessary to resolve the controversy,’ we need not construe [a particular claim limitation] where the construction is not ‘material to the . . . dispute.’” (citation omitted)).

C. Legal Standards

To prevail in its challenges to the patentability of claims 1–5, 8–12, and 22–26 of the ’658 patent, Petitioner must demonstrate by a preponderance of the evidence that the claims are unpatentable. 35 U.S.C. § 316(e). “In an [*inter partes* review], the petitioner has the burden from the onset to show with particularity why the patent it challenges is unpatentable.” *Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016). This burden of persuasion never shifts to Patent Owner. *Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015); *see also In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1376 (Fed. Cir. 2016) (“Where, as here, the only question presented is whether due consideration of the four *Graham* factors renders a claim or claims obvious, no burden shifts from the patent challenger to the patentee.”).

A claim is unpatentable for obviousness if, to one of ordinary skill in the pertinent art, “the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made.” *KSR*, 550 U.S. at 406 (quoting 35 U.S.C. § 103(a) (2006)). The question of obviousness is resolved on the basis of underlying factual determinations, including “the scope and content of the prior art”; “differences between the prior art and the

claims at issue”; and “the level of ordinary skill in the pertinent art.” *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966). Additionally, objective indicia of nonobviousness, such as “commercial success, long felt but unsolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented. As indicia of obviousness or nonobviousness, these inquiries may have relevancy.” *Id.* When conducting an obviousness analysis, we consider a prior art reference “not only for what it expressly teaches, but also for what it fairly suggests.” *Bradium Techs. LLC v. Iancu*, 923 F.3d 1032, 1049 (Fed. Cir. 2019) (citation omitted).

A patent claim “is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art.” *KSR*, 550 U.S. at 418. An obviousness determination requires finding “both ‘that a skilled artisan would have been motivated to combine the teachings of the prior art references to achieve the claimed invention, and that the skilled artisan would have had a reasonable expectation of success in doing so.’” *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1367–68 (Fed. Cir. 2016) (citation omitted); *see KSR*, 550 U.S. at 418 (for an obviousness analysis, “it can be important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does”). Also, “[t]hough less common, in appropriate circumstances, a patent can be obvious in light of a single prior art reference if it would have been obvious to modify that reference to arrive at the patented invention.” *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1361 (Fed. Cir. 2016).

“Although the *KSR* test is flexible, the Board ‘must still be careful not to allow hindsight reconstruction of references . . . without any explanation

as to *how* or *why* the references would be combined to produce the claimed invention.” *TriVascular, Inc. v. Samuels*, 812 F.3d 1056, 1066 (Fed. Cir. 2016) (citation omitted). Further, an assertion of obviousness “cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *KSR*, 550 U.S. at 418 (quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)); accord *In re NuVasive, Inc.*, 842 F.3d 1376, 1383 (Fed. Cir. 2016) (stating that “conclusory statements” amount to an “insufficient articulation[] of motivation to combine”; “instead, the finding must be supported by a ‘reasoned explanation’” (citation omitted)); *Magnum Oil*, 829 F.3d at 1380 (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”).

D. Obviousness Ground Based on Suda (Claims 1–5, 8–12, and 22–26)

1. Suda

Suda discloses a “memory management device for managing a nonvolatile semiconductor memory.” Ex. 1003, code (57).

Figure 1 of Suda is reproduced below.

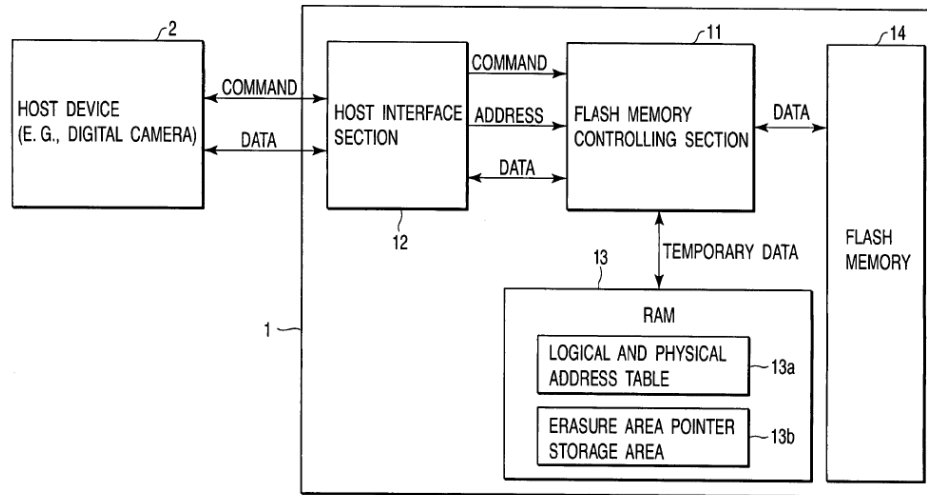


FIG. 1

Figure 1 depicts host device 2, which may be a digital camera, in communication with memory card 1 comprising host interface section 12, flash memory controlling section 11, and flash memory 14 (e.g., a NAND type nonvolatile memory). *Id.* at col. 2, ll. 58–66. “[F]lash memory controlling section 11 manages data erasure and a table indicating a relationship between logical blocks and physical blocks of the flash memory 14.” *Id.* at col. 3, ll. 13–15. Logical and physical address table 13a, stored in random access memory (RAM) 13, “manages logical addresses and physical addresses allocated to physical blocks in which data items are written, of the physical blocks in the flash memory 14, in association with each other.” *Id.* at col. 3, ll. 41–47.

According to Suda, when a subset of pages of a physical block are erased, “the time required for data erasure is long” because the non-erased pages must be read and “written to another physical block.” *Id.* at col. 4, ll. 60–67; *see also id.* at col. 1, ll. 19–23. Suda discloses an improved process that instead writes “erasure area pointer[s]” to erasure area pointer

storage area 13b in RAM 13 indicating that pages in a particular address range are in a “virtual erased state.” *Id.* at col. 5, ll. 14–23. “The virtual erased state is a state in which the flash memory controlling section 11 does not actually erase data items to be erased, i.e., they are subjected to virtual erasure, in response to an access command from the host device 2.” *Id.* at col. 5, ll. 23–27. Specifically, when host device 2 issues a data read command with a logical block address, flash memory controlling section 11 obtains the corresponding physical block address from logical and physical address table 13a, then looks to erasure area pointer storage area 13b to determine whether the requested data is within “an area indicated by the erasure area pointer or pointers”; if so, flash memory controlling section 11 outputs “initial-value” (i.e., empty) data. *Id.* at col. 9, ll. 52–62.

Figure 8 of Suda is reproduced below.

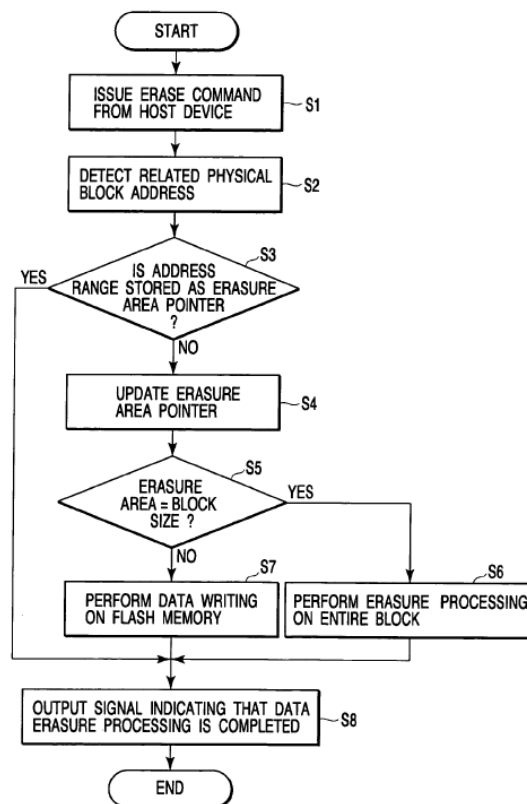


FIG. 8

Figure 8 is a flowchart depicting the disclosed erase process. *Id.* at col. 2, ll. 38–40. Host device 2 issues an erase command to erase particular pages of data stored in a physical block (step S1). *Id.* at col. 7, ll. 11–19. Flash memory controlling section 11 refers to logical and physical address table 13a and “detects the physical address of a physical block related to a logical block given an address design[at]ed in the erasure command” (step S2), then “determines whether an address range corresponding to an area in which the data items to be erased in response to the erase command are stored is already stored in the erasure area pointer storage area 13b” (step S3). *Id.* at col. 7, ll. 30–42. If not, it stores erasure area pointers (i.e., a “start pointer” for the first page address and “end pointer” for the last page address) in erasure area pointer storage area 13b (step S4). *Id.* at col. 5, ll. 36–46, col. 7, ll. 43–55, Figs. 3–5. Flash memory controlling section 11 then “determines whether or not the address range indicated by the erasure area pointer[s] . . . is coincident with the size of a physical block to be subjected to data erasure” (step S5). *Id.* at col. 7, ll. 56–63. If it is coincident (i.e., the entire physical block is already in a “virtual erased state”), the block may be put into an “unused state” by, for example, erasing the associated address information in logical and physical address table 13a (step S6). *Id.* at col. 7, l. 64–col. 8, l. 2. If it is not coincident, flash memory controlling section 11 writes the data items written to erasure area pointer storage area 13b to flash memory 14 so that “even if a power supply to the memory card 1 is turned off, the information of the erasure area pointer is maintained, and thus a virtual erased state is also maintained” (step S7). *Id.* at col. 8, ll. 3–12.

2. *Claim 1*

Petitioner argues that claim 1 is unpatentable over Suda⁵ under 35 U.S.C. § 103(a), relying on the testimony of Dr. Baker as support. Pet. 34–40 (citing Ex. 1004). Patent Owner makes various arguments in response, relying on the testimony of Dr. Madiseti. PO Resp. 26–37 (citing Ex. 2011); Sur-Reply 7–15.

a) Petitioner's Arguments

Petitioner argues that Suda teaches or renders obvious all of the limitations of claim 1. Pet. 34–40. Petitioner asserts that Suda teaches an “apparatus” (i.e., the memory device shown in Figure 1) for managing data stored on a “non-volatile storage medium” (i.e., flash memory 14) comprising a “request receiver module” (i.e., host interface section 12 and/or flash memory controlling section 11 performing various functions in the disclosed erase process) and “storage module” (i.e., flash memory controlling section 11 performing various functions in the disclosed erase process). *Id.* Specifically, with respect to the recited “request receiver module,” Petitioner argues that a person of ordinary skill in the art would have understood that “Suda teaches the recited ‘request receiver module’ in the form of a host interface section and/or the flash memory controlling section,” where both “components receive commands originating from the host device.” *Id.* at 35–36.

Claim 1 recites that the request receiver module is configured to “receive a message comprising a logical identifier, the message indicating

⁵ The three prior art references at issue in this proceeding (Suda, SwSTE’05, and Bennett) were not of record during prosecution of the ’658 patent. *See* Ex. 1001, code (56); Pet. 9.

that data associated with the logical identifier has been erased, wherein the logical identifier is mapped to a physical storage location of the non-volatile storage medium.” Petitioner relies on Suda’s host interface section 12 and flash memory controlling section 11 receiving an erase command from a digital camera (i.e., host device 2) connected to the memory device. *Id.* at 35–38. Petitioner argues that the erase command in Suda designates a logical block address, which constitutes a “logical identifier” and is mapped to physical block numbers in logical and physical address table 13a. *Id.* (citing Ex. 1003, col. 3, ll. 42–55, col. 7, ll. 11–18, 30–34, col. 8, l. 66–col. 9, l. 3, Fig. 7). Petitioner further contends that the digital camera sends an erase command with a logical block address to the memory device, which “processes the erase command to prevent users from later reading this erased data.” *Id.* at 36–37. “Thus, from the perspective of a user of Suda’s digital camera, the designated data has been deleted, and a [person of ordinary skill in the art] would have understood that these erase commands indicate that data has been erased by the host device” according to Petitioner. *Id.* at 37; *see* Ex. 1003, col. 7, ll. 11–19, 34–38 (describing an example of an “erase command to erase 16384-bytes data items (32 pages) of the data items which are managed when they are stored in a physical block given physical block address ‘3’ which is related to logical block address ‘0x40000’ by the logical and physical address table 13a”), Fig. 7.

Finally, claim 1 recites that the storage module is configured to “store persistent data on the non-volatile storage medium in response to the indication, wherein the persistent data is configured to indicate that the data associated with the logical identifier is erased.” Petitioner asserts that the erasure area pointers in Suda record that the range of data in the identified physical block is in a “virtual erased state.” Pet. 38–40. Petitioner argues

that the erasure area pointers are stored as “persistent data” because they are stored in flash memory 14 (as well as RAM 13). *Id.* at 39 (citing Ex. 1003, col. 2, ll. 65–66, col. 3, ll. 41–43, col. 8, ll. 3–16).

Petitioner provides the following annotated version of Figure 7 of Suda. *Id.* at 38.

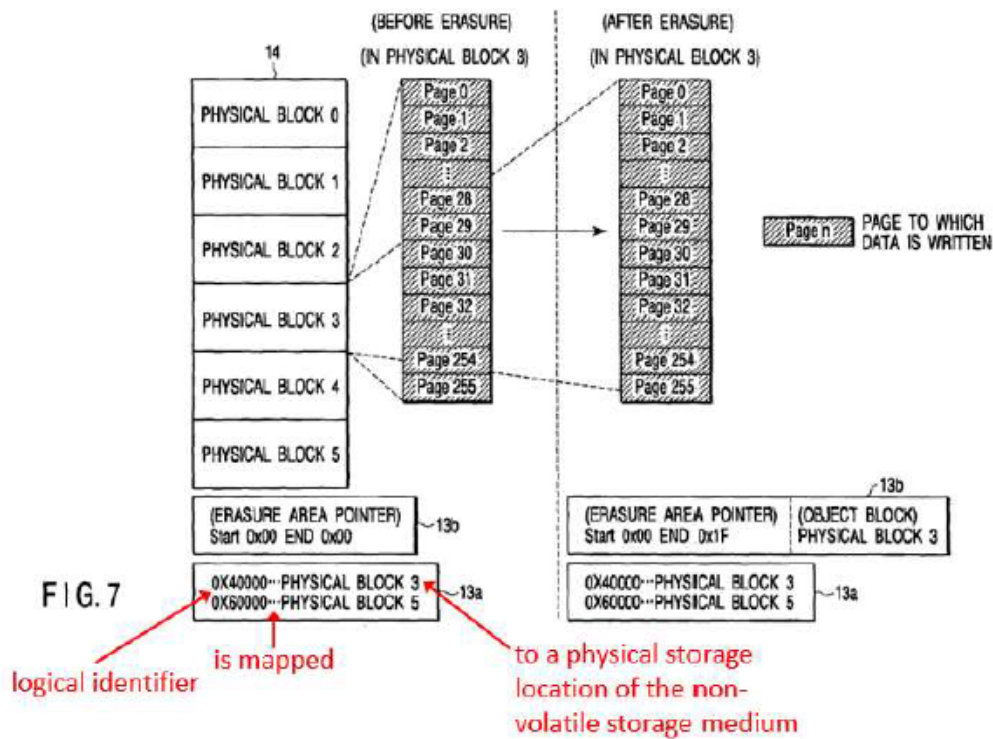


Figure 7, with Petitioner’s annotations in red, depicts erasure area pointer storage area 13b before and after the entry of erasure area pointers for pages 0 to 31 of physical block 3. *Id.*; see Ex. 1003, col. 7, ll. 5–55.

Petitioner argues that “[t]he system will physically erase a block once it fills up with virtually erased data, returning the block to an unused state.” Pet. 15.

b) Patent Owner’s Arguments

Patent Owner makes three arguments regarding claim 1. See PO Resp. 26–37; Sur-Reply 7–15. We address those arguments below.

(1) “Request Receiver Module”

First, Patent Owner disputes Petitioner’s contentions regarding the recited “request receiver module.” PO Resp. 26–33 (citing Ex. 2011 ¶¶ 82–89); Sur-Reply 7–11. Patent Owner argues that Petitioner identifies two different “request receiver module[s]” (i.e., host interface section 12 and flash memory controlling section 11) receiving two different “message[s] comprising a logical identifier” (i.e., the erase command sent to host interface section 12, which Patent Owner refers to as the “Initial Command” or “Initial Message,” and the command sent to flash memory controlling section 11, which Patent Owner refers to as the “Backend Command” or “Backend Instruction”), neither of which satisfies the language of the claim. PO Resp. 26–33. Patent Owner provides the following annotated version of Figure 1 of Suda. *Id.* at 29.

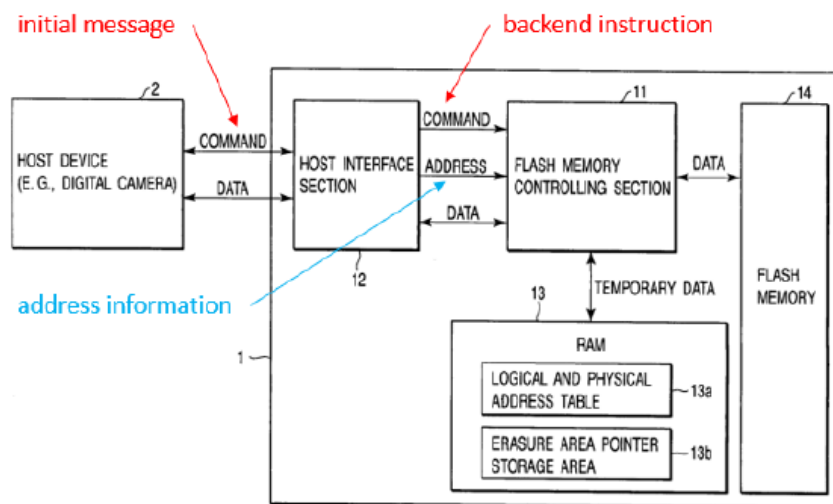


FIG. 1

Annotated Figure 1 depicts the alleged “Initial Message” (red) received by host interface section 12, and the alleged “Backend Instruction” (red) and “address information” (blue) received by flash memory controlling section 11. *Id.*

With respect to Petitioner’s theory that host interface section 12 is a “request receiver module,” Patent Owner argues that claim 1 recites that the storage module stores persistent data “in response to the indication,” but “no data is stored or recorded in response to the Initial Message (the alleged message indication).” *Id.* at 32–33; Sur-Reply 8. According to Patent Owner, even though “Suda uses the words ‘in response to an erase command,’” a person of ordinary skill in the art would have understood that flash memory controlling section 11 stores data in response to a different message (the Backend Command) sent by host interface section 12, not in response to the Initial Message, which “never reaches” flash memory controlling section 11. *Id.*

With respect to Petitioner’s theory that flash memory controlling section 11 is a “request receiver module,” Patent Owner argues that host interface section 12 extracts address information from the Initial Command and sends flash memory controlling section 11 the Backend Instruction separately from the address information over “separate buses.” PO Resp. 28–31 (citing Ex. 1003, col. 3, ll. 7–9, Fig. 1); Sur-Reply 8–10. Thus, “[b]ecause the Backend Instruction does not include address information,” flash memory controlling section 11 does not “receive a message comprising a logical identifier” as recited in the claim. PO Resp. 31–32.

We find that host interface section 12 is a “request receiver module” that receives a “message comprising a logical identifier,” as recited in claim 1. Suda discloses that the erase command, which “designate[s]” a logical block address, is “issued” by host device 2 to host interface section 12. Ex. 1003, col. 3, ll. 4–6, col. 4, ll. 18–26, col. 7, ll. 11–19, 30–35, col. 8, l. 66–col. 9, l. 3. The logical block address is mapped to physical block numbers in logical and physical address table 13a, such that the logical block

address is “related” to a physical block address. *Id.* at col. 7, ll. 11–19, 30–35. The logical block address designated in the erase command is a “logical identifier,” given the undisputed interpretation of the term specified above. *See supra* Section II.B. Indeed, Patent Owner acknowledges that the erase command in Suda “comprises address information” and is “receiv[ed]” by host interface section 12. *See* PO Resp. 27, 31; Sur-Reply 8; Ex. 2011 ¶¶ 82–83, 86. That is all the claim requires of the “request receiver module.” Claim 1 imposes no requirements on what is done with the message after it is received (apart from the storage module storing, “in response to the indication,” persistent data indicating that data associated with the logical identifier is erased, which we address below). *See* Sur-Reply 11 n.3. Thus, the fact that host interface section 12 extracts command and address information from the erase command and sends such information to flash memory controlling section 11 after the erase command is received is immaterial. *See* Ex. 1003, col. 3, ll. 7–9.

To the extent Patent Owner’s arguments are premised on the functionality of the “storage module” recited later in claim 1, the claim only recites that the module is configured to “store persistent data on the non-volatile storage medium in response to the indication, wherein the persistent data is configured to indicate that the data associated with the logical identifier is erased.” Claim 1 does not recite that the storage module itself receives the message. To the contrary, the preceding limitation clearly recites that the request receiver module is what is “configured to receive a message . . . indicating that data associated with the logical identifier has been erased.” We see nothing improper about Petitioner mapping host interface section 12 to the “request receiver module” that receives the “message” and mapping flash memory controlling section 11 to the “storage

module” that stores information in response to host interface section 12 receiving the message.

We further find that Suda teaches recording that data stored at a physical address mapped to the logical identifier is erased “in response to the indication.” The processing at steps S2–S8 of Figure 8 only takes place once host interface section 12 receives the erase command from host device 2 at step S1. *See id.* at col. 7, l. 11–col. 8, l. 20. Suda states numerous times that flash memory controlling section 11 stores erasure area pointers for a particular physical block “in response to” the erase command being received. *Id.* at col. 1, l. 60–col. 2, l. 8 (“a setting unit configured to set an address range of data to be erased *in response to* an erase command in a block in which the data to be erased is written” (emphasis added)), col. 5, ll. 38–46 (“flash memory controlling section 11 designates as a start pointer a page address . . . *in response to* an erase command from the host device 2” then “designates as an end pointer a page address” (emphasis added)), col. 6, ll. 60–63 (“erasure area pointer storage area 13b stores the data items of a physical block in which data items to be erased *in response to* an erase command from the host device 2 are stored”) (emphasis added)), col. 7, ll. 11–19 (“when the host device 2 issues an erase command, . . . the processing is performed *in response to* the erase command” (emphasis added)), 43–53 (“flash memory controlling section 11 performs rewriting processing to change data written to the erasure area pointer storage area 13b . . . in order that the area in which the data items to be erased *in response to* the erase command are stored” (emphasis added)). Given these express disclosures, we disagree with Patent Owner’s position that flash memory controlling section 11 records data in response to the Backend Command

rather than the Initial Message (i.e., the erase command).⁶ *See* PO Resp. 32–33; Sur-Reply 8.

We are persuaded by Petitioner’s arguments, supported by the testimony of Dr. Baker, which we credit, mapping host interface section 12 to the “request receiver module” and mapping flash memory controlling section 11 to the “storage module” of claim 1. *See* Pet. 35–40; Reply 10–11; Ex. 1004 ¶¶ 200–214.

Regardless, though, even if host interface section 12 alone could not be considered a “request receiver module,” we also are persuaded by Petitioner’s argument regarding the combination of host interface section 12 and flash memory controlling section 11. *See* Pet. 35–36 (arguing that “Suda teaches the recited ‘request receiver module’ in the form of a host interface section *and/or* the flash memory controlling section” (emphasis added)); Reply 10–11. We see no reason why the “request receiver module” recited in the claim cannot comprise more than one component. Indeed, Patent Owner proposed in the district court case that the term “module” should be construed to mean “a hardware circuit and/or programmable hardware and/or software implemented within a storage controller,” and acknowledged during the hearing that it is not advocating a different interpretation in this proceeding. *See* Ex. 2007, 16; Tr. 56:20–57:3; Ex. 1001, col. 6, ll. 13–46 (describing “modules” broadly as implementable

⁶ Patent Owner argued during the hearing that Suda does not set erasure area pointers “directly” in response to an erase command. *See* Tr. 65:1–4, 71:19–72:15. Claim 1, however, only recites storing persistent data “in response to the indication.” Patent Owner has not proposed an interpretation of the claim language or otherwise explained why it would be proper to add a “directly” requirement to the claim, and we see no basis on the record before us to do so. *See* Dec. on Inst. 24–25.

in, for example, “programmable hardware devices” or “software for execution by various types of processors”).

Suda does not disclose explicitly what host interface section 12 and flash memory controlling section 11 are, but describes what functions they perform in the disclosed memory management device. Host interface section 12 receives commands from host device 2, “extracts” information from the received commands, and sends command and address information to flash memory controlling section 11. Ex. 1003, col. 3, ll. 4–9. Flash memory controlling section 11 is “connected to the host device 2 by the host interface section 12” and “connected to a RAM 13 and the flash memory 14.” *Id.* at col. 2, l. 66–col. 3, l. 3. It “operates based on” the information received from host interface section 12 and “manages data erasure” using tables 13a and 13b stored in RAM 13. *Id.* at col. 3, ll. 9–15. Under Petitioner’s alternative theory, the combined portions (i.e., hardware and/or software) of host interface section 12 and flash memory controlling section 11 responsible for receiving and processing an erase command would qualify as a “request receiver module,” and the portions of flash memory controlling section 11 responsible for recording erasure information in tables 13a and 13b would qualify as a “storage module.” Contrary to Patent Owner’s argument, they are not the same. *See* Sur-Reply 10–11.

For the reasons explained above, we find that host interface section 12 alone, or, alternatively, the combined portions of host interface section 12 and flash memory controlling section 11 described above, constitute a

“request receiver module” receiving a “message comprising a logical identifier,” as recited in claim 1.⁷

(2) “*Message*”

Second, Patent Owner argues that Suda does not teach or suggest receiving a “message indicating that data associated with the logical identifier has been erased,” as recited in claim 1. PO Resp. 33–35 (citing Ex. 2011 ¶¶ 90–96); Sur-Reply 11–13. Patent Owner contends that the perspective of the user of the digital camera “is irrelevant to whether the message indicates the data associated with the logical identifier has been erased.” PO Resp. 34–35. Patent Owner further disputes Petitioner’s and Dr. Baker’s analysis, arguing that “the erase command is only processed after the ‘message’ is received by the Host Interface section” and the fact that “Suda prevents users from later reading the data . . . does not show a ‘message indicating that data associated with the logical identifier has been erased.’” *Id.* at 33–34. “Instead, it would show that the actions taken as a result of the message (i.e., preventing subsequent reads) indicate the data has been erased. But such actions are not part of the message and do not occur until after the message is received.” *Id.* at 34 (quoting Ex. 2011 ¶ 93).

As an initial matter, we disagree with Patent Owner’s view that the perspective of the digital camera user is irrelevant. Claim 1 does not explicitly state from what perspective data “has been erased,” and thus does not expressly preclude the erasing being assessed from the perspective of the host device sending the message. Petitioner’s understanding, though, is the

⁷ Accordingly, we need not address Petitioner’s other argument that flash memory controlling section 11 alone is a “request receiver module.”

most natural reading of the claim. Claim 1 recites an apparatus with a request receiver module that “receive[s] a message . . . indicating that data associated with the logical identifier has been erased.” In response to that indication, the storage module of the apparatus stores persistent data on the non-volatile storage medium “configured to indicate that the data associated with the logical identifier is erased.” Therefore, nothing has been stored on the non-volatile storage medium to indicate that data is “erased” when the message is received; such storage occurs “in response to” the message indicating that data “has been erased.” This is consistent with Petitioner’s view of Suda, where nothing has been stored on flash memory 14 to indicate that data is erased at the time the erase command is received, and storage of erasure area pointers occurs in response to the erase command. *See* Pet. 35–40.

What we must determine is how a person of ordinary skill in the art would have understood Suda’s disclosure—namely, whether a person of ordinary skill in the art would have understood Suda’s erase command to indicate that “data associated with the logical identifier has been erased.” *See Finisar Corp. v. DirecTV Group, Inc.*, 523 F.3d 1323, 1336 (Fed. Cir. 2008) (“[T]he meaning of a prior art reference requires analysis of the understanding of an artisan of ordinary skill.”). In doing so, we consider the testimony of the parties’ experts, who have different views. *See Acoustic Tech., Inc. v. Itron Networked Solutions, Inc.*, 949 F.3d 1366, 1373 (Fed. Cir. 2020) (“Expert testimony may shed light on what a skilled artisan would reasonably understand or infer from a prior art reference.”).

Suda discloses a digital camera as a host device that uses flash memory to store data. Ex. 1003, col. 2, l. 58–col. 3, l. 15, Fig. 1. The digital camera “issues an erase command to erase” particular pages of data stored in

the flash memory (identified by the provided logical block address), causing flash memory controlling section 11 to perform the disclosed erase process. *Id.* at col. 7, ll. 11–19, col. 8, l. 66–col. 9, l. 3. Suda teaches a specific type of host device (i.e., a digital camera) that was known to store photos in memory. *See* Ex. 1004 ¶ 207. Dr. Baker explains that

Suda teaches an example where the host device is a digital camera. A [person of ordinary skill in the art] would have known that digital cameras at the time allowed users to take and delete photos. When a user deletes a photo, the digital camera will issue an erase command to the memory device. This erase command specifies the logical address of a photo that the user wanted to erase. Then, the flash memory controlling section will process the erase command so that the user will no longer be able to read the erased data (the erased photo). Based on this process, a [person of ordinary skill in the art] would have understood that from the perspective of a user of the digital camera, the erase command indicates that the photo has been deleted, as claimed

Id. (citations omitted).

A person of ordinary skill in the art reading Suda would have understood the way in which file systems delete host data when sending an “erase command” to flash memory. *See Fleming v. Cirrus Design Corp.*, 28 F.4th 1214, 1223 (Fed. Cir. 2022) (“Since *KSR*, we have explained that it is appropriate to consider the knowledge, creativity, and common sense of a skilled artisan in an obviousness determination.”); *supra* Section II.A (the parties’ agreed definition of the level of ordinary skill in the art includes knowledge of “how flash memory erases data, how flash memory programs or writes data, how memory is used in a cache hierarchy,” etc.); Reply 11–12; Ex. 1004 ¶¶ 84–89. Dr. Baker describes the “chain of events when a user reads, writes, or deletes data in a computer system with flash memory” as follows:

A user interacts, through a user interface, with an application or an operating system to modify certain data. For example, the user may specify a file name (e.g., “C:\document.doc”) to be read/written/deleted. Operating systems then access a “file” that translates the file name into another logical address (e.g., a file indicating that “C:\document.doc” is stored at logical address 0x4000). The operating system then generates a command to read, write, or delete data stored at a logical identifier (e.g., at the logical address, or at the file name). The driver and the file system then work together to convert the operating system’s command into the specific electrical signals understood by the flash memory device in order to read, write, or delete data on the flash drive. As part of this process, some file system data in RAM may be modified, and a modification to corresponding file system data on the flash drive will be performed. For example, when sending erase commands, the computer updates its own system “file” in RAM to indicate that C:\document.doc is erased and no longer stored at logical address 0x4000, such as by erasing this file entry.

Ex. 1004 ¶ 89. The device receiving the erase command would “then operate as described” in Suda. *Id.* This is consistent with the ’658 patent itself, which states in the “Background of the Invention” section that “[t]ypically, when data is no longer useful it may be erased. In many file systems, an erase command deletes a directory entry in the file system while leaving the data in place in the storage device containing the data.” See Ex. 1001, col. 1, ll. 22–32; *Qualcomm Inc. v. Apple Inc.*, 24 F.4th 1367, 1376 (Fed. Cir. 2022) (“[A] patentee’s admissions about the scope and content of the prior art provide a factual foundation as to what a skilled artisan would have known at the time of invention.”); *Koninklijke Philips*, 948 F.3d at 1337; *Randall Mfg.*, 733 F.3d at 1362. We find Dr. Baker’s analysis persuasive as to how an ordinarily skilled artisan would have understood Suda’s description of an “erase command” sent by the digital camera.

Dr. Baker's view also is consistent with Suda's description of what happens after the erase command. In particular, in response to the erase command, flash memory controlling section 11 sets erasure area pointers in erasure area pointer storage area 13b, which prevents the data from being read. *See* Ex. 1003, col. 8, ll. 24–38 (disclosing that when a read request is submitted, flash memory controlling section 11 outputs “initial-value data” if the applicable page range is “included in the area indicated by the erasure area pointer”). Thus, the photo is deleted from the user of the digital camera's point of view and cannot be read. Reply 11–12. Petitioner is not relying on the later read command in Suda as an alleged “message,” but rather asserting that the erase command constitutes a “message” that “compris[es] a logical identifier” (i.e., logical block address) and “indicat[es] that data associated with the logical identifier has been erased” (i.e., the message indicates, from the perspective of the user of the digital camera, that a photo was deleted). Pet. 36–37; Reply 12 (arguing that a person of ordinary skill in the art “would understand Suda's device (digital camera) would be operated by a user selecting a photo on the camera interface that the user wanted to delete before sending the erase command”). In other words, what happens after the erase command is evidence of how a person of ordinary skill in the art would have understood Suda's description of the erase command (and what such a person would have understood the erase command to “indicat[e]”). Based on our review of the full trial record, we find that Dr. Baker's explanation as to what a person of ordinary skill in the art allegedly would have understood the erase command to indicate to the flash memory controlling section in the digital camera embodiment is most consistent with the disclosure of Suda. *See* Pet. 35–38; Ex. 1004 ¶ 207.

We are persuaded that a person of ordinary skill in the art would have read the reference in the manner he describes. *See id.*

After reviewing the full trial record, we find that Petitioner has made a sufficient showing regarding the “message” of claim 1.⁸

(3) “*Storage Module*”

Third, Patent Owner argues that Suda does not teach or suggest a “storage module configured to store persistent data on the non-volatile storage medium in response to the indication, wherein the persistent data is configured to indicate that the data associated with the logical identifier is erased.” PO Resp. 36–37 (citing Ex. 2011 ¶ 97); Sur-Reply 13–15. Patent Owner contends that flash memory controlling section 11 “does not store persistent data because the persistent data or flash memory is separate from” flash memory controlling section 11. PO Resp. 36. According to Patent Owner, “it is RAM 13—which includes tables 13a and b—that interacts with the flash memory with respect to persistent data and not” flash memory controlling section 11. Sur-Reply 13–14.

We disagree. Suda discloses that the erasure area pointers are stored in both RAM and the flash memory itself (i.e., as “persistent data,” to

⁸ To the extent Petitioner relies on Patent Owner’s infringement contentions in the district court case (Ex. 1013) for the purpose of either showing that Suda’s logical block address is a “logical identifier” (Pet. 36) or that Suda teaches a similar “message” to what Patent Owner accuses of infringement (*id.*), we are not persuaded. Petitioner does not explain, and we do not see, how Patent Owner’s arguments regarding alleged infringement by Petitioner’s products are relevant to our inquiry here: determining whether Suda teaches or suggests the recited “message.” *See* Dec. on Inst. 34–35; Sur-Reply 11–12. We give no weight to the infringement contentions in conducting that analysis.

preserve the information when power is turned off), and flash memory controlling section 11 is the component responsible for processing, writing, and reading the erasure area pointers in both locations. *See, e.g.*, Ex. 1003, col. 3, ll. 13–15 (“The flash memory controlling section 11 manages data erasure and a table indicating a relationship between logical blocks and physical blocks of the flash memory 14.”), col. 8, ll. 3–17 (“[W]hen the *flash memory controlling section 11* determines in step S5 that the above address range is not coincident with the size of the physical block to be subjected to data erasure, it writes, in the flash memory 14 also, the data items written to the erasure area pointer storage area 13b in step S4 (step S7). . . . [W]hen the power supply to the memory card 1 is turned off, and is then turned on, the flash memory controlling section 11 has only to read information which is stored in the flash memory 14 in step S7, and write the information in the erasure area pointer storage area 13b.” (emphasis added)). We again credit the testimony of Dr. Baker, which is consistent with the disclosure of Suda and background knowledge of a person of ordinary skill in the art. *See* Pet. 16, 38–40; Ex. 1004 ¶¶ 65 (describing the known use of a “controller” to control flash memory), 85 (same), 210–214.

After reviewing the full trial record, we find that Petitioner has made a sufficient showing regarding the “storage module” of claim 1.

c) Conclusion

Petitioner has shown sufficiently that a person of ordinary skill in the art would have understood Suda to teach all of the limitations of claim 1. *See* Pet. 34–40; Ex. 1004 ¶¶ 196–214. Petitioner has proven, by a preponderance of the evidence, that claim 1 would have been obvious based on Suda under 35 U.S.C. § 103(a).

3. *Claims 2–5*

Claims 2–5 each depend, directly or indirectly, from claim 1 and each recite an “index reconstruction module.” For example, claim 2 recites (1) “an index reconstruction module configured to reconstruct mappings between logical identifiers and physical storage locations of the non-volatile storage medium from contents of the non-volatile storage medium,” and (2) “the reconstruction module is configured indicate that data of the logical identifier are erased based on the persistent data stored on the non-volatile storage medium.” As to the first limitation, Petitioner acknowledges that “Suda does not explicitly state to store information in the flash memory 14 for reconstructing the logical and physical address table 13a and then to reconstruct the logical and physical address table 13a in RAM 13,” but Petitioner contends that a person of ordinary skill in the art would have been motivated to modify Suda to do so because “Suda teaches reconstructing the storage and reconstruction of the erasure area pointer area after power-on events,” “flash-memory devices also use this same technique to preserve the logical-to-physical mappings in the flash memory when powered off,” and “both tables need to be reconstructed for the same reasons,” otherwise “the logical-to-physical mappings would be lost every time the power is turned off.” Pet. 40–41 (citing Ex. 1003, col. 8, ll. 6–16, Fig. 1; Ex. 1004 ¶¶ 215–220, 222, 224). As to the second limitation, Petitioner argues that Suda teaches storing persistent data (i.e., erasure area pointers) in flash memory 14 and “[a]fter each power-off-and-on cycle, the flash-memory controlling section reads this data from flash memory and writes the erasure area pointer storage area to RAM.” *Id.* at 41 (citing Ex. 1003, col. 8, ll. 13–17, Fig. 1; Ex. 1004 ¶ 225).

Regarding the first limitation, Patent Owner contends that Petitioner's arguments ignore that the claim requires the "index reconstruction module" to be "in non-volatile memory." PO Resp. 37–38 (citing Ex. 201 ¶¶ 98–101); Sur-Reply 15–16. We do not agree with Patent Owner's reading of the claim language. Claim 2 recites an "apparatus" comprising a "non-volatile storage medium," "request receiver module," "storage module," and "index reconstruction module." The "index reconstruction module" is one component of the "apparatus." The claim does not require it to be "in" the non-volatile storage medium as Patent Owner contends. Also, claim 2 recites that the index reconstruction module "reconstruct[s] mappings between logical identifiers and physical storage locations of the non-volatile storage medium from contents of the non-volatile storage medium." The index reconstruction module thus uses the contents of the non-volatile storage medium, but the claim does not require it to be part of the non-volatile storage medium itself.

Patent Owner also argues that Petitioner has not provided sufficient reasoning for why an ordinarily skilled artisan would have modified Suda's system to have an "index reconstruction module" as claimed. PO Resp. 37; Sur-Reply 16. After reviewing the full trial record, we find Petitioner's arguments persuasive. It is undisputed that Suda teaches storing the erasure area pointers in flash memory and reconstructing erasure area pointer storage area 13b from that persistent data when power is turned off and then back on. *See* Pet. 41. Suda discloses that

when the flash memory controlling section 11 determines in step S5 that the above address range is not coincident with the size of the physical block to be subjected to data erasure, it writes, in the flash memory 14 also, the data items written to the erasure area pointer storage area 13b in step S4 (step S7). When the step S7

is carried out, even if a power supply to the memory card 1 is turned off, the information of the erasure area pointer is maintained, and thus a virtual erased state is also maintained.

Furthermore, when the power supply to the memory card 1 is turned off, and is then turned on, the flash memory controlling section 11 has only to read information which is stored in the flash memory 14 in step S7, and write the information in the erasure area pointer storage area 13b.

Ex. 1003, col. 8, ll. 3–17. Petitioner’s position is that a person of ordinary skill in the art would have been motivated to modify Suda to also store the mappings between logical block addresses and physical block addresses in flash memory and reconstruct logical and physical address table 13a in “the same way” that Suda reconstructs erasure area pointer storage area 13b when power is turned off and then back on. Pet. 40–41; Reply 14–15. The motivation to do so would be “to actually preserve the data, and not lose it, when [the] device is powered off,” and the modification to Suda would have been “simply us[ing] a known technique to improve a similar device in the same way.” *See id.*; Ex. 1004 ¶¶ 218, 223. Petitioner has provided reasoned analysis, supported by evidence in the record, to show that a person of ordinary skill in the art would have been motivated to modify Suda to reconstruct mappings in logical and physical address table 13a in RAM 13 in the same way that Suda already reconstructs erasure area pointer storage area 13b. *See Arendi*, 832 F.3d at 1361; *Dow Jones*, 606 F.3d at 1350–51 (determining that claims would have been obvious where a person of ordinary skill in the art would have been motivated to add a “well known” feature to a prior art reference to meet the limitations).

Claims 3 and 4 recite similar “index reconstruction module” limitations, and claim 5 depends from claim 4. We have reviewed Petitioner’s contentions, which are supported by the testimony of Dr. Baker,

and are persuaded that the recited “index reconstruction module” would have been obvious based on the disclosure of Suda, and that Suda teaches the other limitations of claims 2–5 (which are not disputed by Patent Owner). *See* Pet. 40–44; Ex. 1004 ¶¶ 215–220, 224–237. Petitioner has proven, by a preponderance of the evidence, that claims 2–5 would have been obvious based on Suda under 35 U.S.C. § 103(a).

4. *Claim 8*

Claim 8 depends from claim 1 and recites “a read request module configured to return an indication that the data of the logical identifier are erased in response a request pertaining to the logical identifier received while the data associated with the logical identifier remains on the physical storage location of the non-volatile storage medium.” Petitioner argues that flash memory controlling section 11, “in response to a data-read command that includes a logical block address, determines whether a page is included in an erasure area and, if so, outputs initial-value data,” which is “data initially written in an unused physical block where ‘user data is not written,’ or in other words, empty data.” Pet. 44 (citing Ex. 1004 ¶ 110).

Initial-value data is returned “even though the actual data remains stored in the physical storage location,” as shown in the annotated Figure 7 above.

Id.; *see supra* Section II.D.2.a. Petitioner contends that a person of ordinary skill in the art would have understood that the components of flash memory controlling section 11, implemented as “hardware and software,” that process the read command and output the initial-value data (as shown in the read process of Figure 9) constitute a “read request module” as claimed. Pet. 44–45.

Patent Owner argues that Petitioner improperly relies on the same structure in Suda for the claimed “request receiver module” of claim 1 and the “read request module” of claim 8. PO Resp. 39 (citing Ex. 2011 ¶ 102); Sur-Reply 17. Petitioner, however, does not rely on the same components. As explained above, we agree with Petitioner that host interface section 12 alone, or, alternatively, the combined portions of host interface section 12 and flash memory controlling section 11 described in Petitioner’s analysis, constitute a “request receiver module.” *See supra* Section II.D.2.b.(1). And Petitioner identifies the components of flash memory controlling section 11, implemented as “hardware and software,” that process the read command and output the initial-value data (as shown in the read process of Figure 9) as a “read request module.” Pet. 44–45.

Patent Owner also argues that “Suda’s return of ‘initial-value data’ does not signal that the data of the logical identifier are erased. In Suda, the ‘initial-value data’ would be all ones and it would signal that the address ‘was never written to.’” PO Resp. 38 (quoting Ex. 2013, 89:3–13); Sur-Reply 17. According to Patent Owner, “simply because an address has not been written to, does not mean it has been erased.” PO Resp. 38. The record, however, supports Petitioner’s view. When a read request is submitted in Suda, flash memory controlling section 11 outputs “initial-value data” if the applicable page range is “included in the area indicated by the erasure area pointer” and outputs “applicable data to be read” if the applicable page range has not been virtually erased (i.e., has valid data). Ex. 1003, col. 8, ll. 24–41. Therefore, in the circumstance when a portion of a physical block has been virtually erased, the initial-value data that Suda returns would be “an indication that the data of the logical identifier are erased,” as recited in claim 8. This is consistent with how a

person of ordinary skill in the art would have understood the use of initial-value data. *See* Ex. 1004 ¶ 73 (“Standard flash memory ‘erases’ data in a block by changing any bits storing a ‘0’ in that block from a 0 to 1.”); Ex. 1011, 3 (“Erase operations are needed to change the value from 0 to 1.”).

Petitioner has proven, by a preponderance of the evidence, that claim 8 would have been obvious based on Suda under 35 U.S.C. § 103(a).

5. *Claim 9*

Claim 9 depends from claim 1 and recites “a marking module configured to record that contents of the physical storage location associated with the logical identifier no longer need to be retained on a non-volatile storage medium in response to the indication.” Petitioner argues that Suda’s “erasure area pointers indicate ‘data items to be erased’ later,” meaning that they “no longer need to be retained,” and a person of ordinary skill in the art would have understood that flash memory controlling section 11 is configured to set erasure area pointers in response to an erase command using a “hardware circuit and/or programmable hardware and/or software” (i.e., a “module”). Pet. 45–46. Patent Owner does not argue separately dependent claim 9. *See* PO Resp. 26–46; Sur-Reply 7–21. We disagree with Patent Owner’s arguments regarding parent claim 1 for the reasons explained above. *See supra* Section II.D.2. We have reviewed Petitioner’s contentions and supporting evidence, including the testimony of Dr. Baker, and are persuaded that Petitioner has proven, by a preponderance of the evidence, that claim 9 would have been obvious based on Suda under 35 U.S.C. § 103(a), for the reasons stated by Petitioner. *See* Pet. 45–46; Ex. 1004 ¶¶ 241–243.

6. *Claim 10*

Claim 10 depends from claim 1 and recites “a marking module configured to invalidate an index entry configured to associate the logical identifier with the physical storage location in response to the indication.” Petitioner argues that Suda teaches a “marking module” for the same reasons as presented for claim 9, and relies on Suda’s teachings of “canceling the relation between the logical block addresses and the physical addresses” (in the example of Figure 4, described in column 5 of Suda) and “eras[ing] address information” (in the example of Figure 6, physical block B, described in column 6 of Suda). Pet. 46 (quoting Ex. 1003, col. 5, l. 65–col. 6, l. 3, col. 6, ll. 35–41; citing Ex. 1004 ¶¶ 244–245); *see supra* Section II.D.5. Petitioner argues that both “occur in response to an erase command (the claimed indication)” and a person of ordinary skill in the art would have “known that the cancellation/erasure of the logical block address and the physical address in the logical-and-physical address table invalidates the association between those addresses.” Pet. 46. Alternatively, Petitioner cites Suda’s storage of erasure area pointers to indicate that physical storage is “virtually erased,” which Petitioner contends is an “invalid state.” *Id.* at 47–48 (citing Ex. 1003, col. 5, ll. 9–61, Figs. 3–7; Ex. 1004 ¶¶ 112, 211, 246).

Patent Owner responds that Suda does not teach “invalidat[ing] an index entry.” PO Resp. 39–44 (citing Ex. 2011 ¶¶ 103–105); Sur-Reply 17–20. Patent Owner argues that Suda does not use the word “invalid” at all and in fact “suggests just the opposite” of invalidation because it “discloses checking to ensure that the physical addresses associated with both erase requests and read requests do not fall within an erasure area,” citing Suda’s

description of Figures 8 and 9. PO Resp. 39–40. As to Figure 8 depicting an erase process, Patent Owner contends that prior to performing steps S4–S7, Suda in step S3 “check[s] whether the ‘data items to be erased in response to the erase command’ are ‘already stored in the erasure area pointer storage area.’” *Id.* at 40–41 (quoting Ex. 1003, col. 7, ll. 38–42). As to Figure 9 depicting a read process, Patent Owner contends that Suda likewise “refers to the erasure area pointer storage area 13b, and determines whether a page range in which data items to be read is included in the area indicated by the erasure area pointer.” *Id.* at 41–42 (quoting Ex. 1003, col. 8, ll. 30–34). According to Patent Owner and Dr. Madisetti, “[t]he fact that Suda recognizes erase and read requests can involve physical addresses that are within the erasure area pointer, means that the purported cancellation does not invalidate the entry.” *Id.* at 42–43 (quoting Ex. 2011 ¶ 104). Patent Owner reasons that

[i]f the virtual erase state invalidated the index entry, that would mean that logical addresses would no longer map to the physical address containing the invalid data. Suda’s disclosure that the physical block address must be checked against the erasure area pointers in read and erase requests means that Suda does not disclose invalidating the index entry.

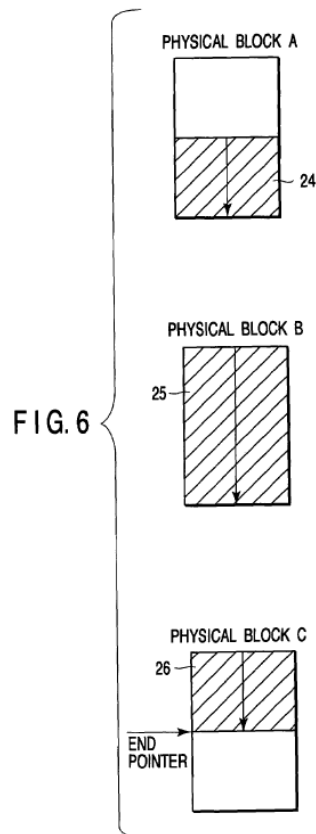
Id. at 43 (quoting Ex. 2011 ¶ 104); Sur-Reply 17–18. Also, “[i]f cancellation amounted to invalidation, it would not be necessary for either erase or read requests to confirm the requested materials are outside the erasure area pointers.” PO Resp. 43 (quoting Ex. 2011 ¶ 104).

Finally, Patent Owner points out that the portions of Suda relied upon by Petitioner for the alleged “message” (i.e., erase command) with respect to parent claim 1 are “not the same as the erasure area pointers section of Suda that addresses cancelling the relation between logical and physical

addresses” with respect to dependent claim 10, and the alleged invalidation is not “in response to the indication.” *Id.* at 43–44; Sur-Reply 20.

After reviewing the full trial record, we find that a person of ordinary skill in the art would have understood that Suda’s description of “eras[ing] address information” in the example of Figure 6 constitutes “invalidat[ing] an index entry configured to associate the logical identifier with the physical storage location in response to the indication,” as recited in claim 10.⁹

Figure 6 of Suda is reproduced below.



⁹ Because we find Petitioner’s showing with respect to “eras[ing] address information” in Suda sufficient, we need not address whether Suda’s other descriptions of “canceling the relation between the logical block addresses and the physical addresses” and storing erasure area pointers likewise constitute “invalidat[ing] an index entry configured to associate the logical identifier with the physical storage location in response to the indication.” *See* Pet. 46–47.

Figure 6 depicts exemplary uses of erasure area pointers to denote virtual erasure areas of physical blocks. Ex. 1003, col. 6, ll. 15–16. Suda provides

an explanation of processing in the case where *an erase command* to erase data items written to a number of physical blocks in the flash memory 14 *is issued from the host device 2*.

To be more specific, the processing will be explained by referring to the case where *an erase command* to erase data items written to three physical blocks (physical blocks A, B and C) *is issued from the host device 2*. . . .

With respect to the physical block B, the data items written to the entire area of the physical block B are to be erased. Thus, the flash memory controlling section 11 . . . *erases address information of the physical block B and a logical block address related to the address information of the physical block B from the logical and physical address table 13a*, thereby setting the entire area (area 25) of the physical block B in an unused state.

Id. at col. 6, ll. 18–41 (emphasis added). The scenario described with respect to physical block B occurs when the full block contains data to be virtually erased (unlike, for example, physical blocks A and C where only part of the physical block is to be virtually erased). *See id.* at col. 6, ll. 34–36; Pet. 15–16 (“When erasing the block, the corresponding logical and physical address entry is removed.”); Ex. 1004 ¶ 245.

Erasing address information from logical and physical address table 13a reflecting the association between a logical block address and physical block address in Suda constitutes “invalidat[ing] an index entry configured to associate the logical identifier with the physical storage location.” *See* Pet. 46–47. It also is performed “in response to the indication.” The excerpt of Suda quoted above discloses that the processing begins with the erase command to erase data items written to physical block B being issued from host device 2; address information is erased in response to that command,

returning the full physical block to an unused state. *See* Ex. 1003, col. 6, ll. 18–41.

We also disagree with Patent Owner’s arguments regarding Figures 8 and 9. Figure 8 depicts a “procedure for erasing data written to the memory card 1,” and Figure 9 depicts a “procedure for reading data written to the memory card 1,” in Suda’s “second example.” *Id.* at col. 2, ll. 38–43. Both, therefore, assume that there is “data written to the memory card 1.” *See id.* In other words, there is a valid mapping to data written in physical blocks. Indeed, Suda discloses that for Figure 8, “the structures of physical blocks not yet subjected to data erasure and the contents of information to be managed by the logical and physical address table 13a are the same as those in the first example” (i.e., Figure 2, showing two entries in logical and physical address table 13a), suggesting that Figure 8 is not describing the scenario of Figure 6 (physical block B) where address information has been removed from logical and physical address table 13a. *See id.* at col. 3, ll. 16–19, col. 7, ll. 19–25. Also, step S2 of Figure 8 and step A1 of Figure 9 both involve detecting a physical block address related to a logical block address in logical and physical address table 13a. *Id.* at col. 7, ll. 30–35, col. 8, ll. 24–30. Logically, therefore, the scenario described in reference to Figures 8 and 9 would “occur[] when data in only a part of a block was subjected to an erase command, causing erasure area pointers to mark that part of the block invalid, without invalidating the address mappings” (like what would occur for a full physical block, as described in reference to physical block B of Figure 6). *See* Reply 17 (emphasis omitted).

We find that Suda teaches “a marking module configured to invalidate an index entry configured to associate the logical identifier with the physical storage location in response to the indication,” for the reasons stated by

Petitioner. *See* Pet. 46–47; Ex. 1004 ¶¶ 244–245. Petitioner has proven, by a preponderance of the evidence, that claim 10 would have been obvious based on Suda under 35 U.S.C. § 103(a).

7. *Claim 11*

Claim 11 depends from claim 1 and recites “a storage recovery module configured to recover a storage division comprising the physical storage location in response to the indication.” Petitioner identifies as a “storage recovery module” the components of flash memory controlling section 11, flash memory 14, and RAM 13, implemented as “circuitry and/or software/firmware,” that “erase the contents of blocks in a storage-recovery process whenever an erase command triggers the erasure area pointers to indicate a block has filled with virtually erased data.” Pet. 47; *see* Ex. 1004 ¶ 250. Petitioner asserts that a person of ordinary skill in the art would have understood Suda’s description of physical block B of Figure 6 (shown above) to mean that “block B will have its contents physically erased in a storage-recovery process so that block B can then write new data again.”¹⁰ Pet. 47–48. Patent Owner responds that “[s]torage recovery is not the same as erasing” and Petitioner fails to explain how the cited portions of Suda describing erasure constitute “recover[ing] a storage division comprising the physical storage location.” PO Resp. 44–45 (citing Ex. 2011 ¶ 106); Sur-Reply 20–21.

¹⁰ Petitioner also refers to Patent Owner’s infringement contentions in arguing that Suda teaches the additional limitation of claim 11. Pet. 47 (citing Ex. 1013). For the reasons explained above, we give no weight to the infringement contentions when determining whether the reference teaches the limitations of the claim. *See supra* Section II.D.2.b.

We are persuaded that Suda teaches the limitation of claim 11 because Suda's process returns a physical block (i.e., a "storage division" comprising a "physical storage location") to a point where it can be written to again (i.e., "storage recovery"). Suda discloses that a virtually erased block can be "set in an unused state," and "initial-value data" is written to "unused" physical blocks. Ex. 1003, col. 3, ll. 56–63, col. 5, l. 65–col. 6, l. 3; *see* Pet. 47–48; Reply 19–20. An "unused" physical block "can be used" again by storing an entry in logical and physical address table 13a indicating a relation between the physical block address and a logical block address. Ex. 1003, col. 3, ll. 41–67 ("[A] physical block the physical block address of which is not related to a logical block address is an unused physical block. In the unused physical block, initial-value data is written. . . . The unused physical block can be used when its physical block address is related to a logical block address in accordance with the control of the flash memory controlling section 11."). Given these disclosures, we credit the testimony of Dr. Baker that a person of ordinary skill in the art would have understood that Suda's process of having block "contents physically erased during the storage recovery process so that the blocks can write new data," a form of "garbage collection," constitutes "recover[ing] a storage division comprising the physical storage location." *See* Ex. 1004 ¶¶ 250–251. The physical storage is "recover[ed]" because it is in a state in which it can be written to again. *See* Pet. 47–48; Reply 19–20.

Petitioner has proven, by a preponderance of the evidence, that claim 11 would have been obvious based on Suda under 35 U.S.C. § 103(a).

8. *Claim 12*

Claim 12 depends from claim 1 and recites “an erase module configured to erase the physical storage location in response to the indication.” Petitioner identifies as an “erase module” the components of flash memory controlling section 11, implemented as “circuitry and/or software/firmware,” that “erase the contents of blocks in a storage-recovery process whenever the erasure area pointers indicate a block has filled with virtually erased data in response to an erase command.” Pet. 48 (citing Ex. 1004 ¶ 254). Patent Owner argues that Petitioner improperly relies on the same disclosures in Suda for both the “storage recovery module” of claim 11 and the “erase module” of claim 12. PO Resp. 45 (citing Ex. 2011 ¶ 107); Sur-Reply 21. According to Patent Owner, a “storage recovery module” recovers a storage division comprising a physical storage location, “not that data is actually erased at that point,” as “recovering data is the opposite of erasing data.” Sur-Reply 21.

Petitioner does not rely on exactly the same components of Suda as teaching the two modules. Petitioner identifies the components of flash memory controlling section 11, flash memory 14, and RAM 13, implemented as “circuitry and/or software/firmware,” that “erase the contents of blocks in a storage-recovery process whenever an erase command triggers the erasure area pointers to indicate a block has filled with virtually erased data” as a “storage recovery module” (claim 11), and identifies the components of flash memory controlling section 11, implemented as “circuitry and/or software/firmware,” that “erase the contents of blocks in a storage-recovery process whenever the erasure area pointers indicate a block has filled with virtually erased data in response to an erase command” as an “erase module” (claim 12). Pet. 47–48; Reply 20.

Suda describes, in reference to physical block B of Figure 6, that “an erase command to *erase* data items written to a number of physical blocks in the flash memory 14 is issued from the host device 2” and “the data items written to the entire area of the physical block B are to be *erased*.”

Ex. 1003, col. 6, ll. 18–21, 34–41 (emphasis added). Suda, therefore, teaches “eras[ing] the physical storage location in response to the indication,” as recited in claim 12.

Petitioner has proven, by a preponderance of the evidence, that claim 12 would have been obvious based on Suda under 35 U.S.C. § 103(a).

9. Claims 22–26

Independent claim 22 recites “receiving an indication, comprising a logical identifier, that the logical identifier mapped to a physical storage location comprising data associated with the logical identifier in the index *is empty*” (emphasis added). Claims 23–26 depend from claim 22. Claims 25 and 26 also similarly recite that “the logical identifier *is empty*” (emphasis added). Petitioner in its asserted ground applies Patent Owner’s proposed construction from the district court case that the adjective “empty” refers to data identified by the logical identifier, not the logical identifier itself, arguing that Suda’s erase command “indicat[es]” that the identified data does not need to be preserved. Pet. 50–53. We disagree with that interpretation for the reasons explained above. *See supra* Section II.B; Ex. 2006, 2 (concluding that the “logical identifier [that/in the index] is empty” claim language is indefinite). Petitioner does not address claims 22–26 in its Reply.

We cannot ascertain the scope of claims 22, 25, and 26 with reasonable certainty for purposes of assessing patentability and thus cannot

reach a decision on the merits with respect to whether Petitioner has established the unpatentability of claims 22–26 over Suda under 35 U.S.C. § 103(a). *See Samsung Elecs. Am., Inc. v. Prisia Eng'g Corp.*, 948 F.3d 1342, 1353 (Fed. Cir. 2020) (“[T]he proper course for the Board to follow, if it cannot ascertain the scope of a claim with reasonable certainty for purposes of assessing patentability, is to decline to institute the [*inter partes* review] or, if the indefiniteness issue affects only certain claims, to conclude that it could not reach a decision on the merits with respect to whether petitioner had established the unpatentability of those claims under sections 102 or 103.”); *Marvell Semiconductor, Inc. v. Uniloc 2017 LLC*, IPR2019-01349, Paper 24 at 60 (PTAB Feb. 1, 2021) (final written decision reaching the same conclusion as to certain claims for which the Board could not ascertain their scope with reasonable certainty); *Cisco Sys., Inc. v. Bushnell Hawthorne, LLC*, IPR2019-00750, Paper 13 at 9–10 (PTAB Sept. 24, 2019) (citing prior cases where the scope of the challenged claims could not be determined and the petitioner “had not shown adequately how the prior art applied to the claims,” and denying institution based on a district court decision that all of the challenged claims were indefinite).

E. Obviousness Ground Based on Suda and SwSTE'05 (Claims 2–5, 10–12, 23, and 25)

1. SwSTE'05

SwSTE'05 is an IEEE journal article entitled “Mapping Structures for Flash Memories: Techniques and Open Problems.” Ex. 1010, 1. SwSTE'05 states that “sophisticated data structures and algorithms are required to effectively use flash memories” and provides a survey of “the data structures

and algorithms that have been developed for management of flash storage.”
Id.

One technique involves block mapping for improved wear-leveling, where “the block number presented by the host, called a *virtual block number*,” is mapped “to a physical flash address called a *sector*.” *Id.* at 2. “When a virtual block needs rewritten, the new data does not overwrite the sector where the block is currently stored. Instead, the new data is written to another sector and the virtual-block-to-sector map is updated.” *Id.* “[T]here are two kinds of data structures that represent such mappings. . . . [D]irect maps allow efficient mapping of blocks to sectors, and inverse maps allow efficient mapping of sectors to blocks.” *Id.* at 3. “Direct maps are stored at least partially in RAM” and “[i]nverse maps are stored on the flash device itself.” *Id.* “When a block is written to a sector, the identity of the block” is also stored “in a header immediately preceding the data” along with other data. *Id.* at 2–3. Figure 1 of SwSTE’05 is reproduced below.

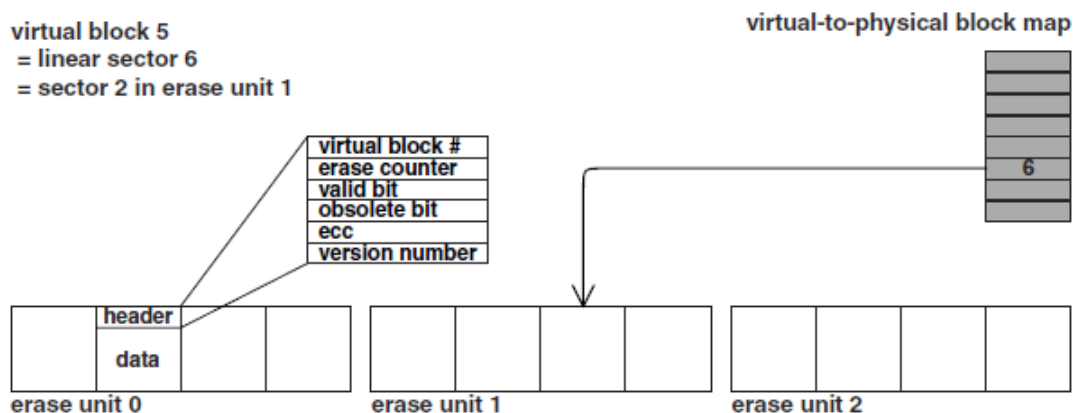


Figure 1 depicts data structures for block mapping in a flash device. *Id.* at 4.
SwSTE’05 discloses:

The gray array on the right [of Figure 1] is the virtual block to physical sector direct map, residing in RAM. Each physical sector contains a header and data. The header contains the index of the virtual block stored in the sector, an erase counter, valid

and obsolete bits, and perhaps an error-correction code and a version number. The virtual block numbers in the headers of populated sectors constitute the inverse map, from which a direct map can be constructed.

Id. The valid bit indicates whether a block has been written, such that “the sector is ready for reading.” *Id.* at 2.

SwSTE’05 also discloses a reclamation process, stating:

Over time, the flash device accumulates obsolete sectors and the number of free sectors decrease. To make space for new blocks and for updated blocks, obsolete sectors must be reclaimed. Since the only way to reclaim a sector is to erase an entire unit, reclamation (sometimes called *garbage collection*) operates on entire erase units.

Id. at 5. “Reclamation can take place either in the background (when the CPU is idle) or on-demand when the amount of free space drops below a predetermined threshold.” *Id.* The reclamation process involves the following steps:

One or more erase units are selected for reclamation.

The valid sectors of these units are copied to newly allocated free space elsewhere in the device. Copying the valid data prior to erasing the reclaimed units ensures persistence even if a fault occurs during reclamation.

The data structures that map logical blocks to sectors are updated if necessary, to reflect the relocation.

Finally, the reclaimed erase units are erased and their sectors are added to the free-sector reserve. This stage might also include writing an erase-unit header on each newly-erased unit.

Id.

2. Claims 2–5

Claim 2 recites, in part, “an index reconstruction module configured to reconstruct mappings between logical identifiers and physical storage

locations of the non-volatile storage medium from contents of the non-volatile storage medium.” Claims 3 and 4 recite similar “index reconstruction module” limitations, and claim 5 depends from claim 4. Petitioner argues that the subject matter of claims 2–5 would have been obvious over the combination of Suda and SwSTE’05,¹¹ relying on Suda for the limitations of parent claim 1, SwSTE’05 for the “index reconstruction module” limitations, and Suda for all other limitations of claims 2–5. Pet. 53–56. Petitioner contends that “[t]o the extent [Patent Owner] argues” that claims 2–5 would not have been obvious based on Suda alone, the claims would have been obvious based on the combination of Suda and SwSTE’05, citing SwSTE’05’s disclosure of storing an inverse map “on the flash device itself” and using the inverse map to “reconstruct a direct map during device initialization (when the device is inserted into a system or when the system boots).” *Id.* at 53–54 (quoting Ex. 1010, 3). Petitioner contends that a person of ordinary skill in the art would have been motivated to modify Suda’s memory device based on the teachings of SwSTE’05 to “store[] an inverse map of the logical and physical address table in the flash memory, and then reconstruct[] the logical and physical address table (the direct map) in RAM 13 during device initialization (a power-on event),” thereby “preserv[ing] the mappings through power-off events.” *Id.* at 54 (citing Ex. 1004 ¶¶ 223–224).

¹¹ Petitioner provides evidence supporting its contention that SwSTE’05 is a prior art printed publication under 35 U.S.C. § 102(b). *See* Pet. 9 (citing Ex. 1004 ¶¶ 59–60; Ex. 1027, 1–2; Ex. 1030 ¶¶ 46–49, 54; Ex. 1034). Patent Owner does not assert otherwise in its Response, and we agree that SwSTE’05 is prior art for the reasons stated by Petitioner.

Patent Owner makes two arguments. First, Patent Owner contends that a person of ordinary skill in the art would not have been motivated to make the asserted combination, relying on Dr. Madisetti's testimony that

Suda already includes protections against power off events, so the modification would not provide benefit. Further, mapping tables, such as the one disclosed in SwSTE'05, require frequent updating. While this is not as important when operating in volatile memory, it is significant in non-volatile flash memory. Flash memory has a limited number of write and erase cycles in its life before it becomes unreliable. Accordingly, if the inverse mapping table is stored in flash memory as Petitioner suggests, the memory will quickly wear out. For obvious reasons, it is not a good idea to store a mapping table in memory that is unreliable.

PO Resp. 47 (quoting Ex. 2011 ¶ 111); Sur-Reply 22–23.

We disagree. Dr. Madisetti does not identify specifically what “protections against power off events” Suda “already” has, or compare such protections to the asserted modification to explain why using SwSTE'05's inverse map would provide no benefit. *See* Ex. 2011 ¶ 111. Further, SwSTE'05 does not support Dr. Madisetti's opinion that storing the inverse map in flash memory would cause the memory to quickly wear out. *See id.* Suda describes known techniques for using flash memory that “support *efficient* not-in-place updates of data, reduce the number of erasures, and *level the wear* of the blocks in the device.” Ex. 1010, 1 (emphasis added). One of those techniques is an inverse map stored on the device itself, which “allow[s] efficient mappings of sectors to blocks.” *Id.* at 3. Suda, therefore, suggests that an inverse map was a common, efficient technique for using flash memory. *See Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120, 1125 (Fed. Cir. 2000) (evidence of a motivation to combine prior art references “may flow from the prior art references themselves”). Other evidence of record supports that understanding. *See*,

e.g., Ex. 1004 ¶ 82 (describing, as background knowledge of a person of ordinary skill in the art, the use of a FTL table to “map the logical to physical addresses,” where “[t]he data structure must be preserved on nonvolatile memory in case of power loss” and “[t]he copy of the FTL table preserved on nonvolatile memory may be a direct copy, . . . an inverse/reverse table, . . . or some other structure that allows the FTL table to be reconstructed”); Ex. 1011, 5 (disclosing that “information for reconstructing [an address translation table] should always be maintained [in non-volatile memory] in preparation for power-off” in “PC environments”).

Finally, we find Dr. Baker’s testimony persuasive that a person of ordinary skill in the art

would have found it obvious, in light of SwSTE’05, to modify Suda’s system to store an inverse map of the logical and physical addresses in the flash memory, which could then reconstruct the table in RAM during power-on. A [person of ordinary skill in the art] would have used this modification to preserve the mappings through power-off events and avoid losing the mappings every time the device powered off. A [person of ordinary skill in the art] would have successfully used this technique in the same way that Suda used it for preserving the erasure area pointers without any difficulty.

See Ex. 1004 ¶ 223. Logically, a system like Suda’s would need some mechanism to store address mappings in persistent memory to preserve the information when power is turned off; otherwise, “a computer will lose its logical to physical address mappings after a computer turns off, and users will no longer be able to read data using logical addresses.” *See* Reply 21.

Second, Patent Owner argues that SwSTE’05 does not teach an “index reconstruction module” that performs the other limitations of the claims—namely, “indicat[ing] that data of the logical identifier are erased based on

the persistent data stored on the non-volatile storage medium” (claim 2), “indicat[ing] that data associated with the logical identifier are erased based on the persistent data stored on the non-volatile storage medium” (claim 3), “record[ing] an indication that data associated with the logical identifier are erased based on the persistent data stored on the non-volatile storage medium” (claim 4), and “updat[ing] the index to indicate that the data associated with the logical identifier are erased” (claim 5). PO Resp. 48–50 (citing Ex. 2011 ¶ 115); Sur-Reply 24. Petitioner, however, relies on Suda, not SwSTE’05, as teaching these limitations. *See* Pet. 40–43, 55. In the asserted combination, Suda’s device, which already persistently stores and reconstructs erasure area pointers, would be modified based on the teachings of SwSTE’05 to also store an inverse map in flash memory and reconstruct the map on device initialization (such that the “based on the persistent data stored on the non-volatile storage medium” limitation in each claim is taught by the combination of Suda and SwSTE’05). *See* Pet. 40–43, 53–55; Reply 22–23; *supra* Section II.D.3; *Bradium*, 923 F.3d at 1050 (“A finding of obviousness . . . cannot be overcome ‘by attacking references individually where the rejection is based upon the teachings of a combination of references.’” (quoting *In re Merck & Co., Inc.*, 800 F.2d 1091, 1097 (Fed. Cir. 1986)); *In re Mouttet*, 686 F.3d 1322, 1332–33 (Fed. Cir. 2012) (holding that “the test for obviousness is what the combined teachings of the references would have suggested to those having ordinary skill in the art”). Petitioner has made a sufficient showing as to that combination.¹²

¹² As explained above, we are persuaded that claims 2–5 would have been obvious based on Suda alone. *See supra* Section II.D.3.

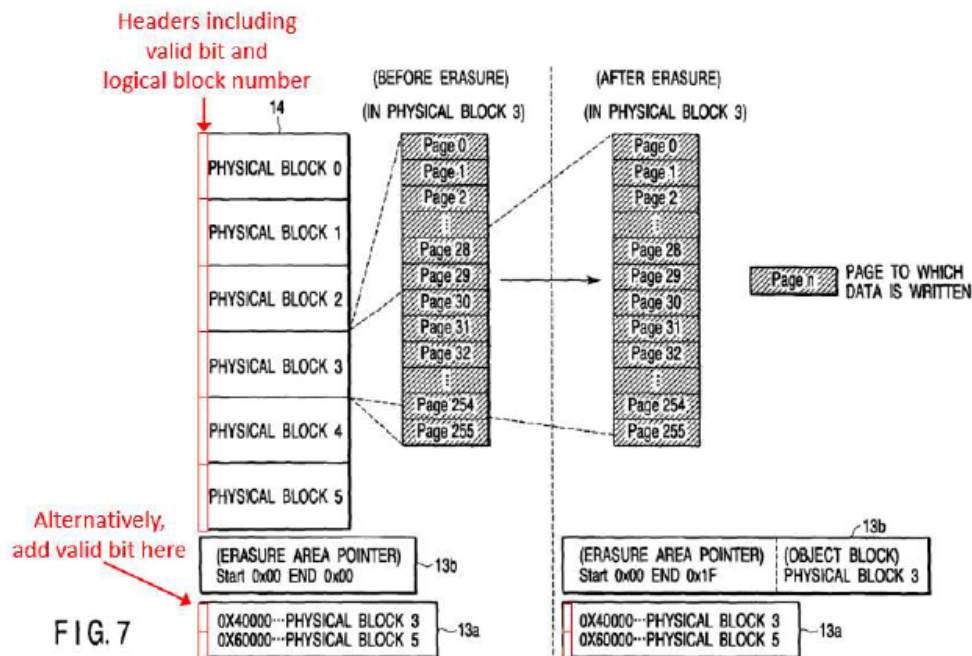
Petitioner has proven, by a preponderance of the evidence, that claims 2–5 would have been obvious based on Suda and SwSTE’05 under 35 U.S.C. § 103(a).

3. *Claim 10*

Claim 10 depends from claim 1 and recites “a marking module configured to invalidate an index entry configured to associate the logical identifier with the physical storage location in response to the indication.” Petitioner argues that the subject matter of claim 10 would have been obvious over the combination of Suda and SwSTE’05, relying on Suda for the limitations of parent claim 1 and SwSTE’05 for the additional limitation of claim 10. Pet. 56–57. Petitioner contends that “[t]o the extent that [Patent Owner] argues” that claim 10 would not have been obvious based on Suda alone,¹³ the claim would have been obvious based on the combination of Suda and SwSTE’05, citing SwSTE’05’s disclosure of a “valid bit” associated with each block, where “[a]n index entry pointing to a block with this valid bit set as invalid indicates that the mapping is to an invalid logical address.” *Id.* at 56 (citing Ex. 1010, Fig. 1). Petitioner argues that a person of ordinary skill in the art would have been motivated to include valid bits “in either headers of blocks or the index” in Suda and set them as invalid when “a block is designated for erasure, in addition to or instead of using erasure area pointers.” *Id.* at 16–17, 56–57.

¹³ As explained above, we are persuaded that Suda alone teaches the additional limitation of claim 10. *See supra* Section II.D.6.

Petitioner provides the following annotated version of Figure 7 of Suda. *Id.* at 57.



Annotated Figure 7 depicts where the valid bit would be stored in the asserted combination, i.e., either in the header of each physical block or in the entry for each physical block in logical and physical address table 13a. *Id.* According to Petitioner and Dr. Baker, combining the teachings of Suda and SwSTE'05 in this manner “would have been an obvious modification because both headers and valid bits were standard techniques,” and a person of ordinary skill in the art “would have had a reasonable expectation of successfully using a small fraction of Suda’s existing memory, whether RAM or flash memory or both, to store the valid bit associated with each logical block address.” *Id.* at 16–18, 57 (citing Ex. 1004 ¶¶ 247–248).

Patent Owner contends that a person of ordinary skill in the art would not have been motivated to make the asserted combination, relying on Dr. Madisetti’s testimony that

[v]alid bits are a less desirable approach as it requires marking each individual page rather than simply a range of pages. Given that Suda requires each page to be invalid before it performs the erase, marking each page is inefficient. This is also contrary to the speed goals of Suda.

PO Resp. 47–48 (quoting Ex. 2011 ¶ 112); Sur-Reply 23–24. We disagree. SwSTE’05 shows that the use of a valid bit to indicate whether a range of flash memory storage is valid or invalid was a well-known technique, and Petitioner has presented persuasive evidence that a person of ordinary skill in the art would have found it advantageous in Suda’s system. *See* Pet. 16–18, 56–57; Ex. 1010, 1–4 (presenting a survey of “data structures and algorithms that ha[d] been developed for management of flash storage” by 2005, including the use of a header containing a valid bit with a range of data storage), Fig. 1; Ex. 1004 ¶¶ 247–248. Also, we disagree that adding a single bit in a header for each physical block, or alternatively in each entry in logical and physical address table 13a, as Petitioner proposes, would have been significantly more inefficient than Suda’s existing erasure area pointer processes. *See* Ex. 1004 ¶ 248 (testifying that headers including a valid bit and other data provide “valuable information,” “are commonly used to provide metadata,” and only take up a “small fraction” of each block, where the “small fraction of storage would be greatly outweighed by the utility of the metadata”). “[A]n obviousness showing ‘does not require that a particular combination must be the preferred, or the most desirable, combination described in the prior art in order to provide motivation for the current invention.’” *Intel Corp. v. Qualcomm Inc.*, – F.4th –, No. 2020-2092, 2022 WL 880681, at *4 (Fed. Cir. Mar. 24, 2022). Petitioner needs to “show only that ‘there is something in the prior art as a whole to suggest the *desirability* . . . of making the combination,’ not

whether there is something in the prior art as a whole to suggest that the combination is the *most desirable* combination available.” *Id.*; *see also Intel Corp. v. Qualcomm Inc.*, 21 F.4th 784, 800 (Fed. Cir. 2021) (“[It is] not necessary to show that a combination is ‘the *best* option, only that it be a *suitable* option.’”). Patent Owner’s argument that using valid bits was a “less desirable approach” than other approaches of the time, therefore, is unavailing. *See* PO Resp. 48.

Petitioner has proven, by a preponderance of the evidence, that claim 10 would have been obvious based on Suda and SwSTE’05 under 35 U.S.C. § 103(a).

4. *Claims 11 and 12*

Claims 11 and 12 depend from claim 1 and recite, respectively, “a storage recovery module configured to recover a storage division comprising the physical storage location in response to the indication” and “an erase module configured to erase the physical storage location in response to the indication.” Petitioner argues that the subject matter of these claims would have been obvious over the combination of Suda and SwSTE’05, relying on Suda for the limitations of parent claim 1 and SwSTE’05 for the additional limitations of dependent claims 11 and 12. Pet. 58–59. For example, regarding claim 11, Petitioner argues that “Suda’s system operates as a ‘garbage collection’ process to reclaim erase units (Suda’s blocks).” *Id.* at 58. Alternatively, Petitioner points to the four-step process described in SwSTE’05 for erase unit reclamation, and argues that the last step of “physically erasing the erase unit” would recover a physical storage location as recited in claim 11. *Id.* at 58–59. According to Petitioner and Dr. Baker, a person of ordinary skill in the art would have been

motivated to apply SwSTE'05's process to Suda, "recognizing that the 'erase units' of SwSTE'05 are Suda's 'blocks,' and that the 'data structures that map logical blocks to sectors' of SwSTE'05 are Suda's tables 13a and 13b," and "would have had a reasonable expectation of success because garbage collection was well-known and had been a standard part of flash-memory management since the mid-1990s."¹⁴ *Id.* at 16–18, 59 (quoting Ex. 1010, 5; citing Ex. 1004 ¶ 252). Similarly, regarding claim 12, Petitioner argues that the last step of SwSTE'05's reclamation process is erasing a physical storage location as recited in claim 12. *Id.* at 59.

Patent Owner does not argue separately dependent claims 11 and 12. *See* PO Resp. 46–50; Sur-Reply 22–24. We disagree with Patent Owner's arguments regarding parent claim 1 for the reasons explained above. *See supra* Section II.D.2. We have reviewed Petitioner's contentions and supporting evidence, including the testimony of Dr. Baker, and are persuaded that Petitioner has proven, by a preponderance of the evidence, that claims 11 and 12 would have been obvious based on Suda and SwSTE'05 under 35 U.S.C. § 103(a), for the reasons stated by Petitioner.¹⁵ *See* Pet. 58–59; Ex. 1004 ¶¶ 250–256.

¹⁴ Petitioner also refers to Patent Owner's infringement contentions in arguing that the additional limitation of claim 11 is taught by Suda and SwSTE'05. Pet. 58 (citing Ex. 1013). For the reasons explained above, we give no weight to the infringement contentions when determining whether the references teach the limitations of the claim. *See supra* Section II.D.2.b.

¹⁵ As explained above, we are persuaded that Suda alone teaches the additional limitations of claims 11 and 12. *See supra* Sections II.D.7, II.D.8.

5. *Claims 23 and 25*

Claims 23 and 25 depend from claim 22. We cannot ascertain the scope of the claims with reasonable certainty for purposes of assessing patentability due to the “empty” phrases in parent claim 22 and dependent claim 25, and thus cannot reach a decision on the merits as to Petitioner’s asserted ground based on Suda and SwSTE’05. *See supra* Section II.D.9.

F. Obviousness Ground Based on Bennett (Claims 1–5, 8–12, and 22–26)

Petitioner contends that claims 1–5, 8–12, and 22–26 are unpatentable over Bennett under 35 U.S.C. § 103(a). Pet. 18–34. In the Decision on Institution, we determined based on the record at the time that Petitioner had not established a reasonable likelihood of prevailing on its asserted ground as to claims 1–5 and 8–12, and we could not ascertain the scope of claims 22–26 with reasonable certainty for purposes of assessing patentability due to the “empty” phrases in claims 22, 25, and 26, such that a meaningful review of Petitioner’s asserted ground could not be performed. Dec. on Inst. 29–36. As explained above, we conclude that claims 1–5 and 8–12 are unpatentable under the Suda-based grounds. *See supra* Sections II.D–E. As such, we need not address Petitioner’s alternative ground based on Bennett. *See Boston Sci. Scimed, Inc. v. Cook Grp. Inc.*, 809 F. App’x 984, 990 (Fed. Cir. Apr. 30, 2020) (non-precedential) (recognizing that “the Board need not address issues that are not necessary to the resolution of the proceeding” and, thus, agreeing that the Board has “discretion to decline to decide additional instituted grounds once the petitioner has prevailed on all its challenged claims”). With respect to claims 22–26, we cannot ascertain the scope of the claims with reasonable certainty and thus cannot reach a

decision on the merits as to Petitioner's asserted ground based on Bennett.
See supra Sections II.D.9, II.E.5.

III. CONCLUSION¹⁶

Petitioner has demonstrated, by a preponderance of the evidence, that claims 1–5 and 8–12 of the '658 patent are unpatentable. We cannot ascertain the scope of claims 22–26 with reasonable certainty for purposes of assessing patentability and thus cannot reach a decision on the merits with respect to whether Petitioner has established unpatentability of those claims under 35 U.S.C. § 103(a).

¹⁶ Should Patent Owner wish to pursue amendment of the challenged claims in a reissue or reexamination proceeding subsequent to the issuance of this Decision, we draw Patent Owner's attention to the April 2019 *Notice Regarding Options for Amendments by Patent Owner Through Reissue or Reexamination During a Pending AIA Trial Proceeding*. *See* 84 Fed. Reg. 16,654 (Apr. 22, 2019). If Patent Owner chooses to file a reissue application or a request for reexamination of the challenged patent, we remind Patent Owner of its continuing obligation to notify the Board of any such related matters in updated mandatory notices. *See* 37 C.F.R. §§ 42.8(a)(3), 42.8(b)(2).

In summary:

Claims	35 U.S.C. §	References/ Basis	Claims Shown Unpatentable	Claims Not Shown Unpatentable
1–5, 8–12, 22–26	103(a)	Suda	1–5, 8–12	
2–5, 10–12, 23, 25	103(a)	Suda, SwSTE’05	2–5, 10–12	
1–5, 8–12, 22–26	103(a)	Bennett ¹⁷		
Overall Outcome			1–5, 8–12 ¹⁸	

IV. ORDER

In consideration of the foregoing, it is hereby:

ORDERED that claims 1–5 and 8–12 of the ’658 patent have been shown to be unpatentable.

This is a final decision. Parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

¹⁷ As explained above, given our disposition of the grounds based on Suda and the combination of Suda and SwSTE’05, we do not reach Petitioner’s alternative ground asserting that claims 1–5 and 8–12 are unpatentable over Bennett. *See supra* Section II.F.

¹⁸ As explained above, we cannot ascertain the scope of claims 22–26 with reasonable certainty for purposes of assessing patentability and thus cannot reach a decision on the merits with respect to whether Petitioner has established unpatentability of claims 22–26 under 35 U.S.C. § 103(a). *See supra* Sections II.D.9, II.E.5, II.F.

IPR2021-00344
Patent 8,762,658 B2

FOR PETITIONER:

Louis L. Campbell
Michael Rueckheim
WINSTON & STRAWN LLP
llcampbell@winston.com
mrueckheim@winston.com

FOR PATENT OWNER:

Berry Bumgardner
Robert A. Delafield II
NELSON BUMGARDNER CONROY PC
barry@nbafirm.com
bobby@nbafirm.com