

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

INTEL CORPORATION,  
Petitioner,

v.

BiTMICRO LLC,  
Patent Owner.

---

IPR2023-00782  
Patent 8,010,740 B2

---

Before KEN B. BARRETT, TERRENCE W. McMILLIN, and  
KEVIN C. TROCK, *Administrative Patent Judges*.

McMILLIN, *Administrative Patent Judge*.

DECISION  
Denying Institution of *Inter Partes* Review  
35 U.S.C. § 314

## I. INTRODUCTION

### A. Background and Summary

Intel Corporation (“Petitioner”)<sup>1</sup> filed a Petition to institute an *inter partes* review of claims 1, 9–15, 32, and 34 (the “challenged claims”) of U.S. Patent 8,010,740 B2 (Ex. 1001, the “’740 patent”) pursuant to 35 U.S.C. § 311 *et seq.* Paper 2 (“Petition” or “Pet.”). BiTMICRO, LLC (“Patent Owner”)<sup>2</sup> filed a Preliminary Response. Paper 6 (“Preliminary Response” or “Prelim. Resp.”).

We have authority under 35 U.S.C. § 314, which provides that an *inter partes* review may not be instituted unless the information presented in the Petition and the Preliminary Response shows that “there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” 35 U.S.C. § 314(a) (2018). For the reasons discussed below, we deny institution.

### B. Related Proceedings

The parties identify the following related proceedings: *BiTMICRO LLC v. KIOXIA Am. Inc., et al.*, 6:22-cv-00331 (W.D. Tex.); and *BiTMICRO LLC v. Intel Corp.*, 5:23-cv-00625 (N.D. Cal.). Pet. 84; Paper 4, 2.

Patent Owner further identifies the following *inter partes* review petition challenging the ’740 patent: *KIOXIA Am. Inc., et al. v. BiTMICRO, LLC*, IPR2023-00742.<sup>3</sup> Paper 4, 2.

---

<sup>1</sup> Petitioner identifies Intel Corporation as the real party-in-interest. Pet. 84.

<sup>2</sup> Patent Owner identifies BitMicro, LLC as the real party-in-interest. Paper 4, 2.

<sup>3</sup> The Board instituted *inter partes* review of the ’740 patent in IPR2023-00742 on October 24, 2023. IPR2023-00742, Paper 8.

*C. The '740 patent*

The '740 patent is titled “Optimizing Memory Operations in an Electronic Storage Device.” Ex. 1001, code (54). The '740 patent explains that the “increased . . . adoption of . . . solid state storage devices due to certain advantages offered by these types of storage devices over other forms of storage devices, such as rotational drives” can be difficult “because many conventional computer devices, sometimes referred to as ‘hosts,’ use host operating systems, file systems, or both that are optimized for use with rotational drives rather than solid state storage devices.” *Id.* at 1:32–41. A “need exists for a solution that optimizes memory operations performed by or in an electronic storage device with a relatively minimal increase in device complexity and costs” and “include[s] increasing the likelihood that, in response to an I/O transaction initiated by a host, the operational load imposed on the storage device by these memory operations will be optimally distributed across different storage device resources, such as by interleaving or parallel memory operations, reducing memory operation latency, increasing operational device efficiency, or both.” *Id.* 2:11–21.

The '740 patent “relates to solutions for increasing the likelihood that the operational load imposed on the memory system by these memory operations, performed in response to an I/O transaction request, will be optimally distributed across resources used by the memory system, increasing operational efficiency, reducing memory operation latency or both.” *Id.* at 1:16–24. The '740 patent provides this solution using a “mapping table.” *See id.* at 2:25–49.

Figure 2, reproduced below, illustrates “a mapping table for an electronic storage device” in accordance to the '740 patent. *Id.* at 2:58–61.

9

| Logical Fields<br>26 | PBA Fields 28              |                  |                    |                        |
|----------------------|----------------------------|------------------|--------------------|------------------------|
|                      | Access Parameter Fields 30 |                  |                    | Memory Index Fields 32 |
|                      | Bus Id Fields 34           | FDE Id Fields 36 | Group Id Fields 38 |                        |
| 40<br>LBAs 00-07     | 8-0                        | 22-01            | 15-0               | 46<br>0x00             |
| 42<br>LBAs 08-15     | 8-0                        | 22-02            | 15-1               | 48<br>0x08             |
| 44<br>LBAs 16-23     | 8-1                        | 22-11            | 15-0               | 50<br>0x00             |
| 45<br>LBAs 24-31     | 8-1                        | 22-12            | 15-1               | 51<br>0x08             |
| ⋮                    | ⋮                          | ⋮                | ⋮                  | ⋮                      |
| ⋮                    | ⋮                          | ⋮                | ⋮                  | ⋮                      |

**FIG. 2**

Figure 2 depicts “Mapping table 9,” which “includes a set of fields, named logical fields 26, and a set of PBA [(physical block address)] fields 28.” *Id.* at 5:27–30.

Figure 1, reproduced below, illustrates “an electronic storage device that includes a memory system which uses a memory table” such as the mapping table illustrated in Figure 2. *Id.* at 2:53–57.

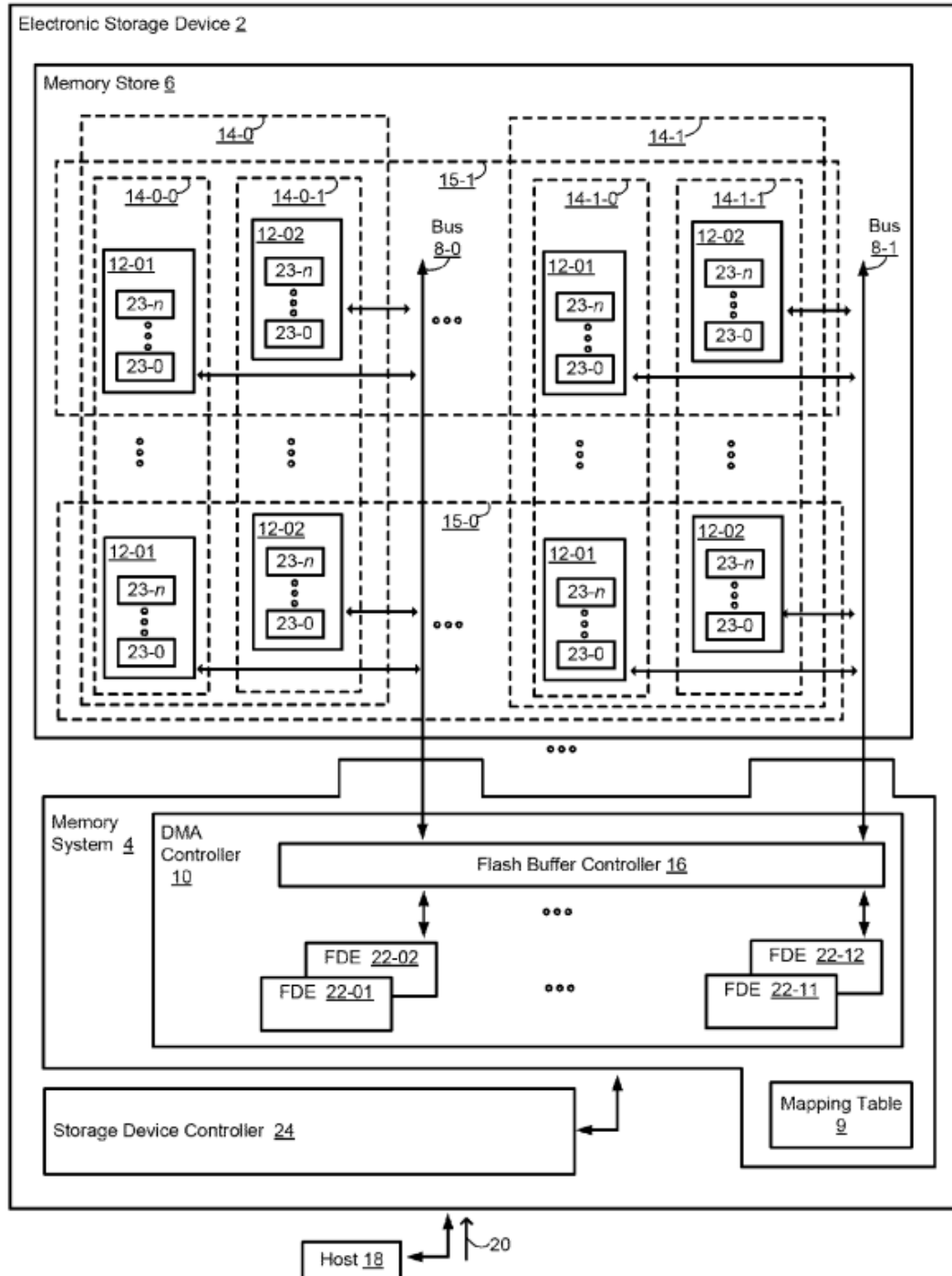


**FIG. 1**

Figure 1 depicts an electronic storage device with a “memory system 4” that “is coupled to a memory store 6 via a set of buses, such as buses 8-0 and 8-1,

and uses a mapping table 9” and that “also includes a DMA controller 10 that includes a flash buffer controller 16 and a set of direct memory access engines.” *Id.* at 3:25–30. “[T]hese direct memory access engines may be . . . referred to as FDEs,” which “represent[] a device that is capable of controlling a flash memory device and performing DMA operations on the flash memory device in response to commands generated by storage device 2 through storage device controller 24.” *Id.* at 3:32–39. “These DMA operations include facilitating high speed data transfers to and from memory devices that are included as part of memory store 6, such as flash memory devices 12-01 and 12-02,” and “[b]uses 8-0 and 8-1 respectively function as a bus interface that is used by an FDE to control and communicate with selected flash memory devices in memory store 6.” *Id.* at 3:39–45.

The ’740 patent discloses that “storage device 2 initializes mapping table 9 by creating an association between . . . LBA [(logical block address)] sets and PBAs so that adjacent LBA sets are mapped to PBAs, and any two adjacent PBAs differ by at least one access parameter.” *Id.* at 7:3–6. “This mapping association among LBA sets and PBAs increases the likelihood that memory operations resulting from an I/O transaction request will be optimized because if these memory operations involve data accesses associated with contiguous LBA sets these data accesses will in likelihood occur on different PBAs,” which “results in the likelihood that different storage resources will be used, such as different buses, different FDEs, different groups or any combination of these, which provides a form of interleaving by bus, FDE, flash memory device group or any combination of these.” *Id.* at 7:9–19.

#### *D. Challenged Claims*

Petitioner challenges claims 1, 9–15, 32, and 34 of the '740 patent.

Pet. 1–2. Of the challenged claims, claims 1 and 32 are independent claims.

Ex. 1001, 9:34–10:24, 13:29–14:5. Claim 1 recites:

1. A mapping table for optimizing memory operations performed by an electronic storage device in response to receiving an I/O transaction request initiated by a host, said mapping table comprising:

a set of logical fields, including a first logical field and a second logical field, and said logical fields respectively disposed for representing a plurality of LBA sets, including said first logical field disposed for representing a first LBA set and said second logical field disposed for representing a second LBA set, said **first and second LBA sets each representing a set of consecutive LBAS;**

a set of PBA fields, including a first PBA field and a second PBA field, said set of PBA fields respectively disposed for representing a set of PBAs, including a first PBA disposed for representing a first set of access parameters and a second PBA disposed for representing a second set of access parameters, said PBAS each associated with a physical memory location in a memory store, said set of logical fields and said set of PBA fields disposed to associate said first and second LBA sets with said first and second PBAs; and

wherein, in response to receiving the I/O transaction request, said mapping table causes the electronic storage device to perform optimized memory operations on memory locations respectively associated with said first PBA and said second PBA, if the I/O transaction request is associated with said first and second LBA sets.

*Id.* at 9:34–10:24 (disputed limitation highlighted).

#### *E. The Asserted Grounds*

Petitioner challenges claims 1, 9–15, 32, and 34 of the '740 patent based on the grounds set forth in the table below.

| Claims Challenged | 35 U.S.C. § | Reference(s)/Basis                |
|-------------------|-------------|-----------------------------------|
| 1, 14             | 103         | Bruce-006 <sup>4</sup>            |
| 1, 14             | 103         | Bruce-006, SBC <sup>5</sup>       |
| 1, 9–15, 32, 34   | 103         | Bruce-006, Bruce-251 <sup>6</sup> |
| 1, 9–15, 32, 34   | 103         | Bruce-006, Bruce-251, SBC         |

Pet. 2.

## II. ANALYSIS

### A. Legal Standards

A patent claim is unpatentable as obvious if the differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations including: (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of ordinary skill in the art; and (4) objective evidence of non-obviousness. *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966).

“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 Technology, Inc.*, 815 F.3d 1358, 1363 (Fed. Cir. 2016) (citing 35 U.S.C. § 312(a)(3) (requiring *inter partes* review petitions to identify “with particularity . . . the evidence that supports the grounds for the challenge to each claim”)). Petitioner cannot satisfy its

---

<sup>4</sup> US 6,000,006, issued Dec. 7, 1999 (Ex. 1005).

<sup>5</sup> US 5,822,251, issued Oct. 13, 1998 (Ex. 1006).

<sup>6</sup> *American National Standard for Information Technology–SCSI-3 Block Commands (SBC)* (Ex. 1007).

burden of proving obviousness by employing “mere conclusory statements.”  
*In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016).

*B. Level of Ordinary Skill in the Art*

With regard to the level of ordinary skill in the art, Petitioner contends a person of ordinary skill in the art “would have had at least a bachelor’s degree in electrical engineering or computer engineering and two years’ experience in memory system design, including solid-state memory, or equivalent combined education, work, and/or experience.” Pet. 5 (citing Ex. 1003 ¶ 22). Patent Owner does not address the level of ordinary skill in the art. *See generally* Prelim. Resp.

For purposes of this Decision, we adopt Petitioner’s definition of the level of ordinary skill in the art because it is consistent with the ’740 patent and the asserted prior art.

*C. Claim Construction*

In an *inter partes* review for a petition filed on or after November 13, 2018, “[claims] of a patent . . . shall be construed 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) (2018); *see also Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–14 (Fed. Cir. 2005). We shall construe only terms that are in controversy and then only to the extent necessary to resolve the controversy. *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999). “In determining the meaning of the 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) (citing *Phillips*, 415 F.3d at 1312–17). There is a “heavy presumption,” however, that a claim term carries its ordinary and customary meaning. *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002) (citation omitted).

Petitioner submits constructions for two claim terms which it applies for purposes of the Petition. Pet. 5–6. For other claim terms, Petitioner “appl[ies] plain meaning.” *Id.* at 5.

The two terms discussed in the Petition and the proposed constructions are:

|         |                                                                                                                    |
|---------|--------------------------------------------------------------------------------------------------------------------|
| LBA     | also referred to as a logical block address, an address that is part of a logical addressing system used by a host |
| LBA set | set of consecutive LBAs                                                                                            |

*Id.* at 6. Petitioner supports these constructions with citations to the Specification of the ’740 patent. *See id.*

Patent Owner proposes that we not engage in any claim construction. Prelim. Resp. 10. Patent Owner contends that “Petitioner fails to establish any ‘LBA set’ under Petitioner’s own proposed construction of this term, and fails to establish any ‘LBA sets each representing a set of consecutive LBAs’ as required by the challenged claims,” and “fails to establish the unpatentability of every challenged claim.” *Id.* at 10–11. Patent Owner does not argue that Petitioner’s proposed constructions are incorrect or flawed in any respect. *Id.*

Based on this record, we adopt the uncontested constructions proposed by Petitioner and use these constructions in evaluating Petitioner's contentions in the Petition.

*D. Summaries of Cited Art*

*1. Bruce-006 (Ex. 1005)*

Bruce-006 is titled “Unified Re-Map and Cache-Index Table with Dual Write-Counters for Wear-Leveling of Non-Volatile Flash RAM Mass Storage.” Ex. 1005, code (54). Specifically, Bruce-006 discloses a “flash-memory system [that] provides solid-state mass storage as a replacement to a hard disk,” using a “unified re-map table in a RAM . . . to arbitrarily re-map all logical addresses from a host system to physical addresses of flash-memory devices.” *Id.*, code (57). Bruce-006 discloses a “single lookup of the re-map table for a logical address produces the physical block address (PBA) of the data in flash memory, the write-counts for wear-leveling, and the cache index (pointer to the data in the cache) should the data currently be in the cache.” *Id.* at 5:33–37. “The wear-leveling swap . . . moves the data for the heavily-used LBA 011 to the lightly-used PBA 101, while the existing data in PBA 101 for LBA 010 is transferred to the cache and then to the heavily-written PBA 111.” *Id.* at 8:35–39. Bruce-006 also discloses that “[i]nterleaving is desirable since multiple accesses can occur in parallel to the flash devices, improving performance,” and “is accomplished by altering the physical address bits assigned to decoders for the flash devices and the physical address bits assigned to decode blocks and pages within a flash device.” *Id.* at 11:20–26.

## 2. *Bruce-251 (Ex. 1006)*

Bruce-251 is titled “Expandable Flash-Memory Mass-Storage Using Shared Buddy Lines and Intermediate Flash-Bus Between Device-Specific Buffers and Flash-Intelligent DMA Controllers.” Ex. 1006, code (54). Specifically, Bruce-251 discloses an “expandable” “flash-memory system” that uses “flash buffer chips” instead of “directly connecting individual flash-memory chips to a controller.” *Id.*, code (57). “Each flash buffer chip can connect to four banks of flash-memory chips,” enabling “individual chips in a bank [to be] generated from an address sent to the flash buffer chips.” *Id.* The use of “[t]wo flash-specific DMA controllers . . . each with four DMA state machines for controlling the four banks of flash-memory chips attached to a flash buffer chip . . . allows for four-way interleaving,” and “[t]wo flash buses connect the two DMA controllers to flash buffer chips.” *Id.* As a result, “[t]he data can be interleaved among the four banks for each flash buffer chip, and also interleaved among chips connected to the two flash busses.” *Id.* at 6:57–60. Because, “[e]ach bank can be separately accessed, . . . many flash operations [can] be performed in parallel.” *Id.* at 5:21–23.

## 3. *SBC (Ex. 1007)*

SBC is an “American National Standards Institute” (ANSI) standard for “Small Computer System Interface” “SCSI-3 Block Commands (SBC).” Ex. 1007, Title. The “standard defines the command set extensions to facilitate operation of SCSI block devices.” *Id.* at 1. An objective of the standard is to “[p]ermit an application client to communicate with a logical unit that declares itself to be a direct-access device, write-once device, and optical memory device in the device type field of the INQUIRY command

response data over an SCSI service delivery subsystem.” *Id.* A “block device” is a “device that is capable of containing data stored in blocks that have a unique logical block address.” *Id.* at 3. SBC defines “commands for direct-access block devices,” including READ(10) and WRITE(10). *Id.* at 28–29. For both the READ(10) and WRITE(10) commands, the “TRANSFER LENGTH field specifies the number of contiguous logical blocks of data that shall be transferred.” *Id.* at 41, 57.

*E. Discussion of Challenges to the Claims*

Patent Owner opposes institution on the basis that Petitioners have failed to establish that the cited art teaches or suggests all the limitations of the challenged claims. Prelim. Resp. 8. Specifically, Patent Owner argues:

[I]n each challenge,<sup>[7]</sup> Petitioner incorrectly relies upon Bruce-006 to disclose the claimed “said first and second *LBA sets* each representing *a set of consecutive LBAs*” in the claimed “mapping table” of claim 1. (emphasis added). Bruce-006 fails to disclose this feature, as Bruce-006 uses a single LBA for an entire block, and then uses a second offset address that is not part of Bruce-006’s re-map table to identify a page within a flash block or byte within a page for a memory transaction. Bruce-006 therefore fails to disclose an[y] LBA sets, much less sets of consecutive LBAs as claimed, and Petitioner fails to establish any “LBA set” even under Petitioner’s own proposed construction of this term. Claim 32 recites substantially similar features as claim 1 (“a mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said

---

<sup>7</sup> The only discussion of the disputed claim elements in the Petition is on pages 14–18 with respect to ground 1 and claim 1. For the other grounds and claim 32, Petitioner relies on and cites back to the alleged showing for ground 1 and claim 1. *See* Pet. 35 (ground 2, claim 1), 49 (ground 3, claim 1), 67 (ground 3, claim 32), 76 (ground 4, claim 1), 78 (ground 4, claim 32). Thus, for every ground and challenged independent claim, Petitioner relies on Bruce-006 for the disputed claim elements.

first and second LBA sets including respective sets of consecutive LBAs”) and Bruce-006 fails to disclose or suggest these features as well for the same reason.

*Id.* at 8–9. Accordingly, we consider whether the Petition establishes that Bruce-006 teaches or suggests “said first and second *LBA sets* each representing *a set of consecutive LBAs*” as recited in claim 1 of the ’740 patent and as similarly recited in additional challenged independent claim 32. Ex. 1001, 10:6–7, 13:37–39.

Petitioner first argues that Bruce-006 uses “somewhat different terminology” for “LBA” (logical block address) and “LBA set” than that used (and recited in the claims) of the ’740 patent. Pet. 15. Petitioner contends that, for “LBA” or logical block address as used in the ’740 patent, Bruce-006 uses “logical address” and, for “LBA set” as used in the ’740 patent, Bruce-006 uses “LBA.” *Id.* Thus, in Petitioner’s view, the claim requirement said “first and second LBA sets each representing a set of consecutive LBAs” is taught by Bruce-006 disclosing first and second LBAs (substituting LBAs for LBA sets) each representing a set of consecutive logical addresses (substituting logical addresses for LBAs). Stated differently, Petitioner contends that, in Bruce-006, an LBA is a set of consecutive logical addresses. In support of these contentions, Petitioner cites the following passage in Bruce-006:

Host requests for access to flash storage are first re-mapped by unified re-map table 20. These host requests contain a logical address which includes a logical-block address (LBA) and an offset address that identifies a page within a flash block and perhaps a byte within a page for a read access.

*Id.* at 15 (citing Ex. 1005, 6:12–16). This passage in Bruce-006 does not support these contentions of Petitioner. This passage does not show that, in Bruce-006, a logical address is a logical block address (LBA) or an LBA is a set of logical addresses. This passages states that, in Bruce-006, a logical address “that identifies a page within a flash block” (a physical address) “includes a logical-block address (LBA) and an offset address.” Ex. 1005, 6:14–16. And, as stated in the detailed description of re-map table 20 in Bruce-006, each LBA entry is for a single logical block address (LBA). Ex. 1005, 6:55–57 (“Unified re-map table 20 contains of plurality of entries 50, each for a different logical block address (LBA).”).

The Petition provides the following argument in an attempt to show that Bruce-006 teaches “first and second LBA sets each representing a set of consecutive LBAs” as recited in claim 1 of the ’740 patent:

In Bruce-006, “[h]ost requests for access to flash storage are first re-mapped by unified re-map table 20. These host requests contain a logical address which includes [what Bruce-006 calls] a logical-block address (LBA) and an offset address that identifies a page within a flash block.” (INTEL-1005,<sup>[8]</sup> 6:12-17.) However, only “blocks rather than pages are re-mapped by the [Bruce-006] re-map table.” (INTEL-1005, 5:63-64.) Thus, an LBA entry in Bruce-006’s re-map table corresponds to and represents the grouping of logical addresses for the consecutive pages in the re-mapped physical block. (INTEL-1005, 6:19-20 (“Unified re-map table 20 contains entries preferably for all LBA’s.”); INTEL-1003,<sup>[9]</sup> ¶116-17.)

---

<sup>8</sup> “INTEL-1005” is Petitioner’s Exhibit 1005 or Bruce-006.

<sup>9</sup> “INTEL-1003” is Petitioner’s Exhibit 1003 or the Declaration of Dr. Carl Sechen, Ph.D. We have considered Dr. Sechen’s Declaration as it relates to the disputed elements of the challenged independent claims. We determine that Dr. Sechen’s testimony is contradicted by the cited passages in Bruce-006.

As such, the Bruce-006 LBAs qualify as the claimed LBA sets (or groupings of consecutive logical addresses). (INTEL-1003, ¶¶116-17.) Annotated Figure 7A below shows a re-map table with “7 sequential logical block addresses (LBA’s).” (INTEL-1005, 8:1-6.)

| LBA | PBA | TOT_WR'S | INCR_WR'S |
|-----|-----|----------|-----------|
| 001 | 001 | 50,000   | 50,000    |
| 010 | 101 | 20,000   | 20,000    |
| 011 | 111 | 250,001  | 250,001   |
| 100 | 011 | 350,000  | 15,000    |
| 101 | 100 | 125,000  | 125,000   |
| 110 | 000 | 200,000  | 200,000   |
| 111 | 010 | 300,000  | 12,000    |

90

TOT\_THRESH = 250,000  
INCR\_THRESH = 25,000

FIG. 7A

(INTEL-1005, Annotated-Figure 7A.) Bruce-006 thus discloses first and second logical fields (highlighted orange and purple, respectively) that represent first and second LBAs. (INTEL-1003, ¶¶114-19.)

Pet. 16–17 (alterations in original).<sup>10</sup>

This argument is conclusory and fails to provide any adequate explanation supporting the statements that either an “LBA entry in Bruce-006’s re-map table corresponds to and represents the grouping of logical addresses for the consecutive pages in the re-mapped physical block” or that “the Bruce-006 LBAs qualify as the claimed LBA sets (or groupings of consecutive logical addresses).” We do not discern any support in the quoted and cited passages in Bruce-006 for these statements. Indeed, the quoted sentence from Bruce-006 that says, “Unified re-map table 20 contains entries preferably for all LBA’s” (Ex. 1005, 6:19–20) indicates that

<sup>10</sup> Any coloration shown in the depicted figures or quoted passages is reproduced from the Petition.

the entries for LBAs in re-map table 20 are for individual LBAs rather than sets of LBAs. And, there are additional passages in Bruce-006 that indicate that the entries for LBAs in re-map table 20 are for individual LBAs rather than sets of LBAs. *See id.* at 5:42–43 (“The re-map table includes a physical block address (PBA) for each possible logical block address.”), 5:66–6:1 (“FIG. 3 shows that logical block address (LBA) 000 is re-mapped to PBA 100, while LBA 001 is re-mapped to PBA 1001.”), 6:55–62 (“Unified re-map table 20 contains a plurality of entries 50, each for a different logical block address. . . . Entry 50 contains physical block address field 44, which contains the physical address of the flash memory device containing the data of the LBA.”), 8:3–6 (“The table shows 7 entries (rows) in the unified re-map table for 7 sequential logical block addresses (LBA’s). These 7 LBA’s are mapped to various physical blocks in flash memory by the PBA fields.”), 8:35 (“This block [PBA 101] is currently allocated to LBA 010.”).

Petitioner asserts that its conclusion that LBAs in Bruce-006 correspond to “LBA sets” as recited in the ’740 patent is supported in Bruce-006 by a teaching that “[f]or block-interleaving, ‘[t]he LBA is the address bits above the page within a block bits.’” Pet. 17 (citing Ex. 1005, 11:67–12:8). We do not find this persuasive. Here again, we do not discern how the relied-upon passage supports the conclusion drawn by Petitioner. This passage is describing the physical address bits which are associated with a single LBA (*see, e.g.*, Ex. 1005, 11:22–26) and does not support the conclusion that LBAs in Bruce-006 correspond to “LBA sets” as recited.

As noted above, Patent Owner disputes Petitioner’s showing as to “LBA sets . . . representing a set of consecutive LBAs” as recited in the ’740 patent claims. The Preliminary Response states:

[Petitioner] misunderstands Bruce-006's disclosure, which does not disclose or suggest any "LBA sets" "representing a set of consecutive LBAs," as claimed. Bruce-006 confirms that an LBA in Bruce-006's re-map table is not an "LBA set" as claimed because Bruce-006 discloses that the "re-map table includes a physical block address (PBA) for *each* possible logical block address." Bruce-006, 5:42-43 (emphasis added). This re-map table arrangement requires that every logical block address in Bruce-006 includes its own entry in the re-map table. Bruce-006 also states that "host requests contain a logical address which includes a logical-block address (LBA) and an offset address that identifies a page within a flash block and perhaps a byte within a page for a read access." Bruce-006, 6:13-17. As such, to identify a page within a flash block or even a byte within a page, a host request in Bruce-006 identifies an LBA, and then uses a separate offset address not included within Bruce-006's LBA to pinpoint the desired page or byte. The offset address used by Bruce-006 identifies a physical relationship and does not disclose a logical LBA "set." Further, the offset address is not included within Bruce-006's re-map table. Bruce-006's LBA (which Petitioner relies upon to disclose a claimed "LBA set") is not an LBA set, and does not include any sets of LBAs, much less a set of consecutive LBAs.

Petitioner also misrepresents what is disclosed in Bruce-006's column 6, lines 19-20. As noted above, Petitioner cites to this disclosure of Bruce-006 as support for arguing that "an LBA entry in Bruce-006's re-map table corresponds to and represents the grouping of logical addresses for the consecutive pages in the re-mapped physical block." Pet., 16. But Bruce-006's col. 6, lines 19-20 says nothing about "consecutive pages." Instead, it states that "Unified re-map table 20 contains entries preferably for all LBA's." Bruce-006, 6:19-20. This simply reinforces that a re-map table 20 includes an entry for each LBA used by the system, and does not support Petitioner's argument that "the Bruce-006 LBAs qualify as the claimed LBA sets (or groupings of consecutive logical addresses)." Pet., 16.

Further, Petitioner's argument incorrectly blurs the difference between "consecutive pages" in a "physical block"

with the concept of “consecutive LBAs,” which are not physical but rather logical. Per claim 1, an LBA set represents a “set of *consecutive* LBAs,” whereas Bruce-006’s LBA is not a “set.” Bruce-006’s re-map table uses a single LBA entry to identify a physical block having multiple pages, and does not use any “LBA set” or “set of consecutive LBAs.” Bruce-006’s LBA entry in the re-map table identifies a block, nothing more. *See* Bruce-006, 8:35 (“This block is currently allocated to LBA 010.”). As such, Bruce-006’s LBA fails to disclose any LBA set “representing a set of consecutive LBAs” as claimed.

Prelim. Resp. 12–14 (footnote omitted). We agree with Patent Owner. The Petition does not show that Bruce-006 teaches or suggests “LBA sets . . . representing a set of consecutive LBAs” as recited in the ’740 patent claims.

In considering this issue, we have reviewed all the passages in Bruce-006 that have been cited by the parties in their respective arguments. *See* Pet. 14–18; Prelim. Resp. 11–14. These passages in Bruce-006 provide:

The re-map table includes a physical block address (PBA) for each possible logical block address.

\* \* \*

Thus blocks rather than pages are re-mapped by the unified re-map table. A block is typically composed of 16 pages. A page is usually 512 bytes in size.

\* \* \*

Host requests for access to flash storage are first re-mapped by unified re-map table 20. These host requests contain a logical address which includes a logical-block address (LBA) and an offset address that identifies a page within a flash block and perhaps a byte within a page for a read access. Some embodiments only read a page or a block at a time and thus do not use offset address bits.

Unified re-map table 20 contains entries preferably for all LBAs.

\* \* \*

In FIG. 7A, the system-wide total threshold is set to 250,000 writes, while the system's incremental-write threshold is set to 25,000 writes. The table shows 7 entries (rows) in the unified re-map table for 7 sequential logical block addresses (LBA's). These 7 LBA's are mapped to various physical blocks in flash memory by the PBA fields.

\* \* \*

This block [PBA 101] is currently allocated to LBA 010.

\* \* \*

#### Interleaving of Flash Blocks or Pages

Interleaving can be combined with the invention. Interleaving is desirable since multiple accesses can occur in parallel to the flash devices, improving performance. Interleaving is accomplished by altering the physical address bits assigned to decoders for the flash devices and the physical address bits assigned to decode blocks and pages within a flash device.

A non-interleaved assignment of physical address bits is:  
3:0 Page within a block  
15:4 Block within a flash device  
31:16 Device within a bank; bank within the system,  
etc.

Interleaving of blocks across four devices has the bit assignment:

3:0 Page within a block  
5:4 Interleave among four devices  
17:6 Block within a flash device  
31:18 Device within a bank; bank within the system,  
etc.

With such block interleaving, a sequence of four blocks are interleaved among four flash devices. Performance is improved when sequential blocks are accessed and the flash devices are slow.

Another type of interleaving interleaves smaller pages rather than larger blocks among multiple devices:

0:1 Interleave among four devices

5:2 Page within a block

17:6 Block within a flash device

31:18 Device within a bank; bank within the system,  
etc.

Since pages are smaller than blocks, addresses are more likely to be interleaved with the page-interleaving scheme. However, wear-leveling and re-mapping can no longer occur for each block, since one block cannot be moved away from the other three blocks in the interleave group. Instead, re-mapping is performed for an interleave of four blocks. Each entry in the unified re-map table then corresponds to an interleave of four blocks, rather than just one block. This has the additional advantage of reducing the size of the re-mapping table. Another advantage is when different flash device capacities are used. Interleaving using the largest block-size can make all devices appear the same.

Interleaving can also be applied to the re-map table and any spare blocks. Different portions of the re-map table are spread out among the interleaves, allowing these portions to be read or written in parallel. Likewise, spare pages are spread out among the interleaves rather than just located at the end of the address space.

The interleaving used can be flexible by multiplexing different logical address bits to the unified re-map table. The LBA is the address bits above the page within a block bits, and includes the interleave bits for block interleaving, but does not include the interleave bits for page interleaving. Bits 31:4 are routed to the re-map table for block interleaving, but bits 31:6 are sent to the table for page interleaving. Bits 4 and 5 are routed to select the entry in the unified re-map for block interleaving but not for page interleaving. Different intended applications or uses for the flash-memory system can be

configured for page or block interleaving, or no interleaving, by the manufacturer.

\* \* \*

| LBA | PBA | TOT_WR'S | INCR_WR'S |
|-----|-----|----------|-----------|
| 001 | 001 | 50,000   | 50,000    |
| 010 | 101 | 20,000   | 20,000    |
| 011 | 111 | 250,001  | 250,001   |
| 100 | 011 | 350,000  | 15,000    |
| 101 | 100 | 125,000  | 125,000   |
| 110 | 000 | 200,000  | 200,000   |
| 111 | 010 | 300,000  | 12,000    |

90
TOT\_THRESH = 250,000  
INCR\_THRESH = 25,000

FIG. 7A

Ex. 1005, 5:42–43, 5:63–65, 6:12–20, 8:1–6, 8:35, 11:19–12:10, Fig. 7A.

These passages indicate that, in Bruce-006, the term LBA (logical block address) refers to a single LBA rather than a set of “consecutive LBAs” as recited in the claims of the ’740 patent.

For these reasons, we determine that Petitioner has not established that an “LBA” as in Bruce-006 is the equivalent of “LBA set” as in the ’740 patent claims. The term “LBA” in Bruce-006 refers to a single logical block address not a set of consecutive logical addresses. Petitioner has identified no teaching or suggestion in Bruce-006 of an “LBA set” as recited in the challenged independent claims of the ’740 patent. Petitioner has not shown that Bruce-006 teaches or suggests “first and second LBA sets each representing a set of consecutive LBAs” as recited in claim 1 or “first and second LBA sets including respective sets of consecutive LBAs” as recited in claim 32.

### III. CONCLUSION

On the record before us, we conclude that there is not a reasonable likelihood that the Petitioner will prevail with respect to showing that at least one of the claims challenged in the Petition is unpatentable. Therefore, we do not institute *inter partes* review on any claims or any challenge to the claims of the '740 patent.

### IV. ORDER

In consideration of the foregoing, it is hereby:

ORDERED that the Petition is *denied* and we do not institute *inter partes* review of any challenged claim of U.S. Patent No. 8,010,740 B2

IPR2023-00782  
Patent 8,010,740 B2

FOR PETITIONER:

Lori A. Gordon  
Tyler R. Bowen  
PERKINS COIE LLP  
gordon-ptab@perkinscoie.com  
bowen-ptab@perkinscoie.com

FOR PATENT OWNER:

Wayne M. Helge  
NOSSAMAN, LLP  
whelge@nossaman.com

James T. Wilson  
DAVIDSON BERQUIST JACKSON &  
GOWDEY, LLP  
jwilson@dbjg.com