

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

KIOXIA AMERICA, INC. and KIOXIA CORPORATION

Petitioners,

v.

BITMICRO LLC, LLC,

Patent Owner.

Case No.: IPR2023-00742

U.S. Patent No. 8,010,740

DECLARATION OF R. JACOB BAKER, Ph.D., P.E.

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	QUALIFICATIONS	1
	A. Industry Experience.....	2
	B. Academic Experience.....	6
	C. Other Relevant Experience.....	8
III.	MATERIALS CONSIDERED	9
IV.	TECHNOLOGY BACKGROUND.....	12
V.	OVERVIEW OF THE '740 PATENT	14
	A. Specification of the '740 Patent	14
	B. Prosecution History for the '740 Patent	17
VI.	LEGAL PRINCIPLES USED IN ANALYSIS	18
	A. Prior Art.....	18
	B. Anticipation.....	18
	C. Obviousness.....	19
	D. Claim Construction	24
VII.	LEVEL OF ORDINARY SKILL IN THE ART.....	25
VIII.	CLAIM CONSTRUCTION	27
	A. “LBA”	27
	B. “PBA”	29
	C. “FDE” and “FDE identifier”	30
	D. “group identifier”	32
IX.	OVERVIEW OF THE PRIOR ART	33
	A. Sukegawa Discloses a LBA to PBA Mapping Table That Facilitates Optimized Operations	33
	B. Bennett Discloses a Logical-to-Physical Address Mapping Table That Enables Optimized Operations in Memory Systems	37
	C. Bruce Discloses Specific Access Parameters for Solid State Memory Systems, Each Having Unique Identifiers.....	41

X.	THE PRIOR ART DISCLOSES ALL OF THE CLAIMED ELEMENTS OF THE '740 PATENT.....	45
A.	Each of the Challenged Claims Is Anticipated or Rendered Obvious by Sukegawa.....	45
1.	Claim 1 Is Anticipated by Sukegawa.....	45
2.	Claim 9 Is Rendered Obvious by Sukegawa in View of Bruce	58
3.	Claim 10 Is Rendered Obvious by Sukegawa in View of Bruce	72
4.	Claim 11 Is Rendered Obvious by Sukegawa in View of Bruce	75
5.	Claim 12 Is Rendered Obvious by Sukegawa in View of Bruce	77
6.	Claim 13 Is Rendered Obvious by Sukegawa in View of Bruce	78
7.	Claim 14 Is Rendered Obvious by Sukegawa in View of Bruce	80
8.	Claim 15 Is Rendered Obvious by Sukegawa in View of Bruce	82
9.	Claim 32 Is Anticipated By Sukegawa.....	87
10.	Claim 34 Is Anticipated by Sukegawa.....	104
B.	Each of the Challenged Claims Is Anticipated or Rendered Obvious by Bennett.....	109
1.	Claim 1 Is Anticipated by Bennett.....	109
2.	Claim 9 Is Rendered Obvious by Bennett in View of Bruce	124
3.	Claim 10 Is Rendered Obvious by Bennett in View of Bruce	137
4.	Claim 11 Is Rendered Obvious by Bennett in View of Bruce	140
5.	Claim 12 Is Rendered Obvious by Bennett in View of Bruce	142
6.	Claim 13 Is Rendered Obvious by Bennett in View of Bruce	143
7.	Claim 14 Is Rendered Obvious by Bennett in View of Bruce	145
8.	Claim 15 Is Rendered Obvious by Bennett in View of	

	Bruce	147
9.	Claim 32 Is Anticipated by Bennett.....	152
10.	Claim 34 Is Rendered Obvious by Bennett in View of Bruce	169
XI.	CONCLUSION.....	172

I, R. Jacob Baker, Ph.D., P.E., declare as follows:

I. INTRODUCTION

1. I have been retained by KIOXIA Corporation and KIOXIA America, Inc. (“Petitioners”) as an independent expert in this *inter partes* review before the United States Patent and Trademark Office (“PTO”) regarding U.S. Patent No. 8,010,740 (“the ’740 patent”), Ex-1001. Specifically, I have been asked to consider whether certain references disclose or suggest the features recited in claims 1, 9-15, 32, and 34 of the ’740 patent (“challenged claims”). As explained below, it is my opinion that the challenged claims are invalid as anticipated and/or obvious in view of the prior art references discussed in this Declaration and the accompanying Petition.

2. I am being compensated at my ordinary consulting rate of \$655.00 per hour. No part of my compensation is dependent on the outcome of this proceeding or any other proceeding involving the ’740 patent or any of its related patents. I have no other interests in this proceeding.

II. QUALIFICATIONS

3. My qualifications are generally set forth in my Curriculum Vitae, which is attached as Ex-1004. My Curriculum Vitae also includes a list of the publications I have authored and a list of the other cases in which I have testified during the last

four years.

4. I have been working as an Engineer since 1985 and have been teaching Electrical and Computer Engineering courses since 1991. I am currently a Professor of Electrical and Computer Engineering at the University of Nevada, Las Vegas (“UNLV”).

5. I received B.S. and M.S. degrees in Electrical Engineering from UNLV in 1986 and 1988, respectively. I received my Ph.D. in Electrical Engineering from the University of Nevada, Reno (“UNR”) in 1993.

6. My doctoral research, culminating in the award of a Ph.D., investigated the use of power MOSFETs in the design of very high peak power, and high-speed, instrumentation. I developed techniques to reliably stack power MOSFETs to switch higher voltages, that is, greater than 1,000 V and 100 Amps of current with nanosecond switching times. This work was reported in the paper entitled “Transformerless Capacitive Coupling of Gate Signals for Series Operation of Power MOSFET Devices,” published in the IEEE Transactions on Power Electronics. The paper received the Best Paper Award in 2000.

A. Industry Experience

7. I have done technical and expert witness consulting for over 200 companies and their subsidiaries since I started working as an Engineer in 1985.

From 1985 to 1993, I worked for EG&G Energy Measurements and the Lawrence Livermore National Laboratory designing nuclear diagnostic instrumentation for underground nuclear weapon tests at the Nevada test site. During this time, I designed, and oversaw the fabrication of, over 30 electronic and electro-optic instruments, including high-speed cable and fiber-optic receiver/transmitters, PLLs, frame and bit-syncs, data converters, streak-camera sweep circuits, Pockel's cell drivers, micro-channel plate gating circuits, charging circuits for battery backup of equipment for recording test data, and analog oscilloscope electronics.

8. My work during this time, as one example, had a direct impact on my doctoral research work using power MOSFETs, subsequent publishing efforts, and industry designs. In addition to the 2000 Best Paper Award from the IEEE Power Electronics Society, I published several other papers in related areas while working in industry. I hold a patent, Patent No. 5,874,830, in the area of power supply design, titled, "Adaptively biased voltage regulator and operating method," which was issued on February 23, 1999. I have designed dozens of linear and switching power supplies for commercial products and scientific instrumentation.

9. I am a licensed Professional Engineer and have extensive industry experience in circuit design, fabrication, and manufacture of Dynamic Random Access Memory (DRAM) semiconductor integrated circuit chips, Phase-Change

Random Access Memory (PCRAM) chips, and CMOS Image Sensors (CISs) at Micron Technology, Inc. (“MTI”) in Boise, Idaho. I spent considerable time working on the development of flash memory chips while at MTI. My efforts resulted in more than a dozen patents relating to flash memory. One of my projects at MTI included the development, design, and testing of circuit design techniques for a multi-level cell (MLC) flash memory using signal processing. This effort resulted in higher-density memories for use in solid-state drives and flash memory cards having an ATA interface that are ubiquitous in consumer electronics, including cameras and data storage systems. Further, the use of higher-density memory can result in fewer changes in the flash translation layer for logical-to-physical addressing, less need for garbage collection, and larger data segments that can improve a computing system’s performance.

10. Another project I worked on at MTI focused on the design of buffers for high-speed double-data rate DRAM, which resulted in around 10 U.S. patents in buffer design. Among many other experiences, I led the development of the delay locked loop (DLL) in the late 1990s so that MTI DRAM products could transition to the DDR memory protocol, used in mobile and non-mobile (server, desktop, cell phones, tablets, etc.) computing systems as main computer memory, for addressing and controlling accesses to memory via interprocess communications (IPC) with the

memory controller (MC). I provided technical assistance with MTI's acquisition of Photobit during 2001 and 2002, including transitioning the manufacture of CIS products into MTI's process technology. Further, I consulted at Sun Microsystems and then Oracle on the design of memory modules during 2009 and 2010. This work entailed the design of low-power, high-speed, and wide interconnection methods with the goal of transmitting data to/from the memory module and the MC at higher speeds.

11. I have extensive experience in the development of instrumentation and commercial products in a multitude of areas including: integrated electrical/biological circuits and systems, array (memory, imagers, and displays) circuit design, CMOS analog and digital circuit design, diagnostic electrical and electro-optic instrumentation for scientific research, CAD tool development and online tutorials, low-power interconnect and packaging techniques, design of communication/interface circuits (to meet commercial standards such as USB, firewire, DDR, PCIe, SPI, etc.), circuit design for the use and storage of renewable energy, and power electronics. For example, a part of my research at Boise State, for many years, focused on the use of Thru-Silicon-Vias (TSVs), aka Thru-Wafer Vias (TWVs), for high-density packaging. These packaging techniques were utilized in the memory module development work I did with Sun Microsystems and Oracle. As

another example, I designed circuitry for use in implementing Universal Serial Bus (USB) interface circuits while consulting at Tower Semiconductor. I designed PCI communication circuits for IPC between a Graphics Processor Unit (GPU) and memory while consulting for Rendition, Inc.

12. My current research work is focused in part on the design of integrated circuits for wireless sensing using LIDAR (Light Detection And Ranging). I have worked with several companies in the development of these circuits and systems, including Freedom Photonics, Aerius Photonics, and FLIR. In the early 1990s, I worked on wireless systems for wideband impulse radar while at Lawrence Livermore Laboratory. Further, part of my research for several years focused on the digitization of IQ channels using delta-sigma modulation. The knowledge and experience gained from this effort are reflected in my textbook *CMOS Mixed-Signal Circuit Design* and a presentation, which I have presented at several universities and companies, http://cmosedu.com/jbaker/papers/talks/BP_DSM_talk.pdf.

B. Academic Experience

13. I was an adjunct faculty member in the Electrical Engineering departments of UNLV and UNR from 1991-1993. From 1993 to 2000, I served on the faculty at the University of Idaho as an Assistant Professor and then as a tenured Associate Professor of Electrical Engineering. In 2000, I joined a new Electrical and

Computer Engineering program at Boise State University (“BSU”), where I served as department chair from 2004 to 2007. At BSU, I helped establish graduate programs in Electrical and Computer Engineering including, in 2006, the university’s second Ph.D. degree. In 2012, I re-joined the faculty at UNLV. Over the course of my career as a professor, I have advised more than 100 masters and doctoral students.

14. I have been recognized for my contributions as an educator in the field. While at Boise State University, I received the President’s Research and Scholarship Award (2005), Honored Faculty Member recognition (2003), and Outstanding Department of Electrical Engineering Faculty recognition (2001). In 2007, I received the Frederick Emmons Terman Award (the “Father of Silicon Valley”), which is bestowed annually upon an outstanding young electrical/computer engineering educator in recognition of the educator’s contributions to the profession. In 2011, I received the IEEE Circuits and Systems Education Award. I received the Tau Beta Pi Outstanding Electrical and Computer Engineering Professor Award every year it was awarded while I have been back at UNLV.

15. I have authored several books and papers in the electrical and computer engineering area. My published books include *CMOS Circuit Design, Layout, and Simulation* (Baker, R.J., Wiley-IEEE, ISBN: 9781119481515 (4th ed., 2019)) and

CMOS Mixed-Signal Circuit Design (Baker, R.J., Wiley-IEEE, ISBN: 9780470290262 (2nd ed., 2009) and ISBN: 9780471227540 (1st ed., 2002)). I co-authored *DRAM Circuit Design: Fundamental and High-Speed Topics* (Keeth, B., Baker, R.J., Johnson, B., and Lin, F., Wiley-IEEE, ISBN: 9780470184752 (2008)), *DRAM Circuit Design: A Tutorial* (Keeth, B. and Baker, R.J., Wiley-IEEE, ISBN: 0-7803-6014-1 (2001)), and *CMOS Circuit Design, Layout and Simulation* (Baker, R.J., Li, H.W., and Boyce, D.E., Wiley - IEEE, ISBN: 9780780334168 (1998)). I contributed as an editor and co-author on several other electrical and computer engineering books.

C. Other Relevant Experience

16. I have performed technical and expert witness consulting for more than 200 companies and their subsidiaries and given more than 50 invited talks at conferences, companies, and universities. Further, I am the author or co-author of more than 100 papers and presentations in the areas of electrical and computer engineering design, fabrication, and packaging.

17. I currently serve, or have served, as a volunteer on the IEEE Press Editorial Board (1999-2004); as editor for the Wiley-IEEE Press Book Series on Microelectronic Systems (2010-2018); as the Technical Program Chair of the 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS

2015); on the IEEE Solid-State Circuits Society (SSCS) Administrative Committee (2011-2016); as a Distinguished Lecturer for the SSCS (2012-2015); the Technology Editor (2012-2014) and Editor-in-Chief (2015-2020) for IEEE Solid-State Circuits Magazine; IEEE Kirchhoff Award Committee (2020-2023); and advisor for the student branch of the IEEE at UNLV (2013-present). These meetings, groups, and publications are intended to allow researchers to share and coordinate research. My active participation in these meetings, groups, and publications allows me to see what other researchers in the field have been doing.

18. In addition to the above, I am an IEEE Fellow for contributions to semiconductor memory design and a member of the honor societies Eta Kappa Nu and Tau Beta Pi.

III. MATERIALS CONSIDERED

19. My opinions, as set forth in this Declaration, are based on the documents I have reviewed, my experience and background, and my knowledge and professional judgment. In forming these opinions, I have reviewed the '740 patent (Ex-1001), the prosecution history for the '740 patent (Ex-1002), the prior art discussed below, and any other material and information identified in this Declaration. I have also relied on my experience and knowledge in the field of computer memory and data storage devices.

20. In particular, my opinions regarding the invalidity of the challenged claims are based on the following prior art references, as informed by my knowledge in the field of computer memory and data storage devices.

Exhibit	Description
1005	U.S. Patent No. 5,572,466 (“Sukegawa”). It is my understanding that Sukegawa was issued on November 5, 1996. Petitioners’ counsel has informed me that Sukegawa is prior art under pre-AIA 35 U.S.C. § 102(b). See <i>infra</i> § VI.B for a more detailed description regarding my understanding of this category of prior art.
1006	U.S. Patent No. 5,822,251 (“Bruce”). It is my understanding that Bruce issued on October 13, 1998. Petitioners’ counsel has informed me that Bruce is prior art under pre-AIA 35 U.S.C. § 102(b). See <i>infra</i> § VI.B for a more detailed description regarding my understanding of this category of prior art.
1007	U.S. Patent No. 7,139,864 (“Bennett”). It is my understanding that Bennett was filed on December 30, 2003, and was published June 30, 2005. Petitioners’ counsel has informed me that Bennett is prior art under pre-AIA 35 U.S.C. § 102(a). See <i>infra</i> § VI.B for a more detailed description regarding my understanding of this category of

Declaration of R. Jacob Baker
Inter Partes Review of U.S. Patent No. 8,010,740

Exhibit	Description
	prior art.
Ex-1008	Claim Construction Order and Memorandum in Support Thereof, <i>BiTMicro LLC v. KIOXIA Am., Inc.</i> , Case No. 6:22-cv-00331-ADA, Dkt. No. 54 (W.D. Tex. Feb. 16, 2023).
Ex-1009	U.S. Patent No. 6,000,006 (“the ’006 patent”)
Ex-1010	U.S. Patent No. 7,506,098 (“the ’098 patent”)
Ex-1013	U.S. Pat. No. 5,404,485 (“Ban”)
Ex-1014	BRIAN DIPERT & MARKUS LEVY, DESIGNING WITH FLASH MEMORY (Annabooks 1994) (“Dipert”).
Ex-1015	H. Niijima, <i>Design of a Solid-State File Using Flash EEPROM</i> , IBM JOURNAL OF RESEARCH AND DEVELOPMENT, vol. 39, no. 5, pp. 531-545, Sept. 1995 (“Niijima”)
Ex-1016	Eran Gal et al., <i>Mapping Structures for Flash Memories: Techniques and Open Problems</i> , PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON SOFTWARE—SCIENCE, TECHNOLOGY & ENGINEERING, Herzlia, Israel, 2005, pp. 83-92, doi: 10.1109/SWSTE.2005.14 (“Gal”)

IV. TECHNOLOGY BACKGROUND

21. Before June 8, 2006, flash memory devices generally were widely used. Likewise, the concept of mapping between logical address used by a host and physical address in the memory, would have been implemented in flash memory systems and POSITAs would have reliably and predictably known how to do so

22. For example, Flash memory systems utilize a Flash Translation Layer (FTL) that allows for the mapping (also referred to as “translating,” “indexing,” or “converting”) of logical addresses to physical addresses and is implemented on the flash device itself. Ex-1016 at 6-9. Logical addresses are the addresses used by computer operating systems to identify certain data. Physical addresses indicate a physical storage location of the data. In platter hard drives, physical addresses would indicate a cylinder, sector, and/or head of a platter where data is stored. In flash drives, physical addresses indicate the block (and also usually pages) of flash memory where data is stored.

23. The FTL was patented by Ban in 1995 in U.S. Pat. No. 5,404,485 titled “Flash file system,” more than ten years before the priority date for the ’740 patent. Ex-1013. The FTL was also adopted as part of the PCMCIA standard before the priority date of the 740 patent. Ex. 1016 at 7. It is used in conjunction with both wear leveling and garbage collection in flash devices. Ex-1015 at 5-9.

24. An FTL is well-known to map the logical to physical addresses using various data structures, such as an “index,” “table,” or other similar structure. *See, e.g.*, Ex-1016 at 7-9; *Id.*, Fig. 1; Ex-1005, Figure 11; Ex.-1007, Figures 3B and 17B. The copy of the FTL table preserved on nonvolatile memory may be a direct copy (Ex. 1014 at 15-17), an inverse/reverse table (Ex-1015 at 5, Ex. 1016 at 7-9), or some other structure that allows the FTL table to be reconstructed. During powered operation, the FTL data structure maintaining the relationship between logical and physical addresses may be reconstructed in volatile memory, such as SRAM or DRAM, for faster access and updating. Ex-1015 at 5; Ex-1016 at 6-9.

25. The FTL mapping between logical and physical identifiers is updated in response to changes in the data. Ex-1016 at 9 (“data structures that map logical blocks to sectors are updated if necessary, to reflect the relocation”). For example, if the physical storage location of data is changed (such as due to wear leveling or garbage collection) then the logical-to-physical address mapping for that data will be updated to reflect the new physical storage location. Similarly, if data is erased altogether, then the corresponding logical-to-physical entry may either be deleted (typically for any mapping is stored in RAM, where erasures are easily performed) or marked as invalid/erased (more common if the mapping is stored in flash, where erasures are relatively slow). By using an FTL, an operating system can track the

location of data using only the logical address. This simplifies the management by the operating system, which can remain oblivious to where data is physically stored. By using FTLs and operating on logical addresses only, operating systems do not need to know how each and every unique storage device uniquely stores and uniquely updates data in different physical locations-such tracking is offloaded to the FTL.

26. In conclusion, well before the January 8, 2006, priority date of the '740 patent, there was a long history of using mapping tables and schemes to correlate logical host addresses with physical addresses in the memory

V. OVERVIEW OF THE '740 PATENT

A. Specification of the '740 Patent

27. I understand that the application for the '740 patent was filed on March 19, 2009, and claims priority to a parent application filed on June 8, 2006. Claim 1 of the '740 patent is an independent claim, while claims 9-15 depend from claim 1. Likewise, claim 32 is an independent claim, and claim 34 depends from claim 32.

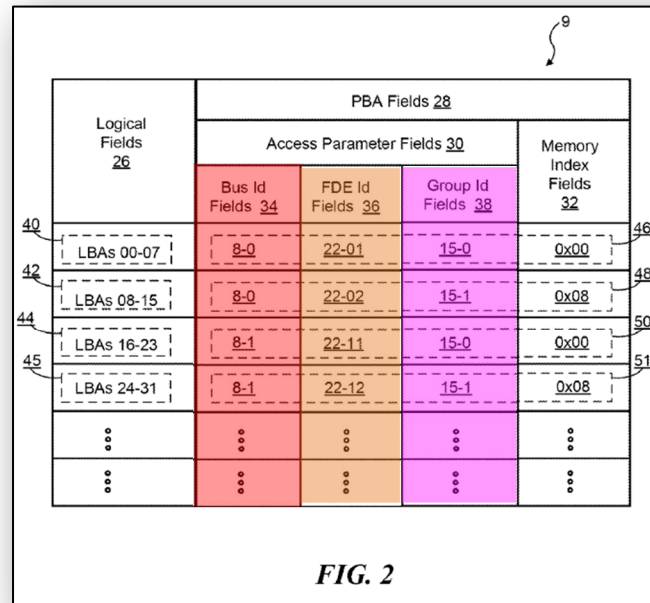
28. The '740 patent purportedly relates to “optimizing memory operations in a memory system suitable for use in an electronic storage device.” Ex-1001 at 1:16-18. As the '740 patent acknowledges, use of non-volatile solid state memory was “commonly known” at the time the '740 patent was filed. *Id.* at 1:28-32.

The '740 patent further recognizes that “flash memory related techniques that include read-modify-write transactions, wear leveling, bad block management or any combination of these,” as well as the use “flash translation layers,” were similarly well-known at the time of filing. *Id.* at 1:54-65. Nonetheless, the '740 patent purports to identify a need for “optimizing” memory operations by “increasing the likelihood that . . . the operational load imposed on the storage device by these memory operations will be optimally distributed across different storage device resources.” *Id.* at 2:11-21.

29. The '740 patent purports to address this alleged need through the use of a “mapping table” that is “disposed to associate” sets of logical block addresses (“LBAs”) from a host system to physical block addresses (“PBAs”) in a memory store. *Id.* at Abstract. According to the '740 patent, the PBAs—which “represent a unique addressable physical memory location” in a memory store—may be “represented” by a set of “access parameter fields.” *Id.* at 5:30-35. These access parameter fields may include a bus identifier, a flash DMA engine (or “FDE”) identifier, and a group identifier. *Id.* at 5:36-38. In other words, the '740 patent purports to identify a PBA by listing the bus, DMA engine, and group of flash devices associated with the PBA.

30. Annotated Figure 2 of the '740 patent, reproduced below, illustrates this

LBA to PBA mapping table. In particular, Figure 2 illustrates the access parameters comprising a **bus identifier**, **FDE identifier**, and **group identifier**.



31. As explained below, however, logical to physical address mapping tables were well known before the priority date of the '740 patent. Both Sukegawa and Bennett disclose such tables and confirm that they had been in use long before the priority date of the '740 patent. Similarly, identifying a unique addressable physical memory location (or PBA) by the bus, DMA engine (or FDE), and group of flash devices the DMA engine is connected to was well known before the priority date of the '740 patent. Bruce, which was filed nearly a decade before the '740 patent, discloses such identifiers. Moreover, the use of mapping tables, as taught by

Sukegawa, represents a common design approach to improving memory system performance and allowing for optimization of memory operations, which would reasonably include the access parameters taught by Bruce. Accordingly, nothing recited in the claims of the '740 patent would have been novel or non-obvious to a person of ordinary skill in the art at the time the application for the '740 patent was filed.

B. Prosecution History for the '740 Patent

32. It is my understanding that, during prosecution, the Examiner did not apply any prior art references to reject the claims. Instead, I understand that, on October 25, 2010, the Examiner rejected the claims as failing to comply with the written description requirement. Ex-1002 at 3-4. In this rejection, the Examiner stated that the specification “does not describe how the mapping table causes the storage device to perform optimized memory operation[s] on memory location[s] in such a way that is understood by one having ordinary skill in the related art.” *Id.*

33. I understand that, in a response dated February 25, 2011, the applicant argued that the specification provided sufficient written description support for the claims, after which the Examiner issued a Notice of Allowance on April 13, 2011. *Id.* at 28-36. Following the Notice of Allowance, I understand that the applicant made clerical amendments to the claims on July 12, 2011, which were subsequently

entered by the Examiner.

VI. LEGAL PRINCIPLES USED IN ANALYSIS

34. I am not an attorney and I offer no opinions on the law. I have been advised by Petitioners' counsel of certain legal principles relevant to an invalidity analysis, as summarized below. I have conducted my analysis in accordance with these principles.

A. Prior Art

35. I understand that a patent or other publication must qualify as prior art before it can be used to invalidate a patent claim.

36. I understand that a U.S. or foreign patent qualifies as prior art to a challenged claim if the patent was issued prior to the invention date of the challenged claim. I also understand that a U.S. or foreign patent or published patent application qualifies as prior art to a challenged claim if it was published before the invention date of the challenged claim.

B. Anticipation

37. I understand that a prior art reference "anticipates" an asserted claim, thereby rendering the claim invalid, if all elements of the claim are disclosed in the reference, either expressly or inherently. I understand that an element is disclosed inherently in a prior art reference if it is necessarily present in the prior art reference or is the natural result flowing from the disclosure of the prior art reference. I have

been informed that a prior art reference does not need to use the same words as the challenged claim in order to anticipate the claim.

38. Petitioners' counsel informs me that the application for the '740 patent was filed before the America Invents Act ("AIA") was enacted. Accordingly, I have been informed that pre-AIA 35 U.S.C. § 102 defines the types of references that can be used as prior art against the challenged claims.

39. I understand that under pre-AIA 35 U.S.C. § 102(a), a challenged claim is anticipated if it was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the patent applicant.

40. I also understand that under pre-AIA 35 U.S.C. § 102(b), a challenged claim is anticipated if was patented or described in a printed publication in this or a foreign country, or in public use or on sale in this country, more than one year prior to the date of the patent application in the United States.

C. Obviousness

41. I also understand that a patent is invalid 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 to a person of ordinary skill in the pertinent art at the time the invention was made. I understand that a person of ordinary skill in the art

provides a reference point from which the prior art and claimed invention should be viewed. This reference point prevents one from using his or her own personal insight or hindsight in deciding whether a claim is obvious.

42. I understand that an obviousness determination includes the consideration of various factors, such as (1) the scope and content of the prior art, (2) the differences between the prior art and the asserted claims, (3) the level of ordinary skill in the pertinent art, and (4) the existence of secondary considerations such as commercial success, long-felt but unresolved needs, failure of others, etc.

43. I understand that an obviousness evaluation can be based on a combination of multiple prior art references. I understand that the prior art references themselves may provide a suggestion, motivation, or reason to combine, which does not need to be explicit. I further understand that any need or problem known in the field and addressed by the patent can provide a reason for combining the elements in the manner claimed. To determine whether there was an apparent reason to combine the known elements in the way a patent claims, I have been informed that I may look to interrelated teachings of multiple patents, to the effects of demands known to the relevant community or present in the marketplace, and to the background knowledge possessed by a person of ordinary skill in the art. I also understand that practical and common-sense considerations should guide a proper

obviousness analysis because familiar items may have obvious uses beyond their primary purposes. I further understand that a person of ordinary skill in the art looking to overcome a problem will often be able to fit together the teachings of multiple publications. I understand that an obviousness analysis therefore accounts for the inferences and creative steps that a person of ordinary skill in the art would employ under the circumstances.

44. I understand that the nexus linking two or more prior art references can also be simple common sense. I further understand that the obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may be supplied by the direction of the marketplace.

45. I understand that if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill.

46. I understand that a particular combination may be proven obvious merely by showing that it was obvious to try the combination. For example, when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good

reason to pursue the known options within his or her technical grasp because the result is likely not the product of innovation, but of ordinary skill and common sense.

47. I understand that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. When a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different field. If a person of ordinary skill can implement a predictable variation, the patent claim is likely obvious.

48. As noted above, I understand that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

49. I understand that a claim can be obvious in light of a single reference, without the need to combine references, if the elements of the claim that are not found explicitly or inherently in the reference can be supplied by the common sense or knowledge of one of skill in the art.

50. I also understand that secondary indicia of non-obviousness may include: (1) a long-felt but unmet need in the prior art that was satisfied by the

invention of the patent; (2) commercial success of processes covered by the patent; (3) unexpected results achieved by the invention; (4) praise of the invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the invention; (7) failure of others to find a solution to the long-felt need; and (8) skepticism by experts.

51. I also understand that there must be a relationship between any such secondary considerations and the invention. I further understand that contemporaneous and independent invention by others is a secondary consideration supporting an obviousness determination.¹

¹ In this case, I am not aware of any such secondary considerations that would support the conclusion that the claims of the '740 patent are not obvious. To the contrary, it does not appear that there was any long-felt but unmet need for the alleged invention; as explained below, LBA-to-PBA mapping tables were well known by the priority date of the '740 patent, as was the use of access parameters to represent PBAs. I am also not aware of any evidence that Bit Microsystems Inc. (the original patentee) had commercial success marketing any product covered by the '740 patent. I am also unaware of any unexpected results achieved by the alleged invention; to the contrary, as described below, prior storage systems used mapping

52. In sum, it is my understanding that prior art teachings are properly combined where a person of ordinary skill in the art—having the understanding and knowledge reflected in the prior art and being motivated by the general problem facing the inventor—would have been led to make the combination of elements recited in the claims. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor at the time of the invention, can provide a reason for combining the elements of multiple prior art references in the claimed manner.

D. Claim Construction

53. I understand that, before an invalidity analysis can be performed, the scope and meaning of the challenged claims must be determined. It is my

tables to link logical host address to physical memory addresses within a memory array. Similarly, use of different access parameters to represent a PBA—such as bus identifiers, DMA engine identifiers, and other identifiers for a group of flash devices—were equally well known and used before the priority date of the '740 patent. I am also unaware any of any praise of the '740 patent by those skilled in the art, any deliberate copying of the '740 patent, any failure of others to find a solution, or any skepticism by experts regarding the systems recited in the '740 patent.

understanding that this is accomplished through the process of claim construction.

54. I have been informed and understand that claims are generally construed from the perspective of one of ordinary skill in the art at the time of the invention, which I have been asked to assume is the '740 patent's priority date of June 8, 2006. More specifically, I understand that the claims should be afforded a meaning in accordance with the ordinary and customary meaning of the claim, as understood by one of ordinary skill in the art at the time of the invention.

55. I have been advised that, in construing a claim term, one must first look to the "intrinsic" patent evidence, which includes the claims themselves, the patent specification, and the prosecution history of the patent. I have also been informed that "extrinsic" evidence, which is outside the patent and prosecution history, may be useful in construing patent claims. It is my understanding that such extrinsic evidence may include dictionaries, textbooks, treatises, and similar sources of information.

56. I understand that claims must be construed in an *inter partes* review using the same claim construction standard that would be used to construe the claim in a civil action before a district court.

VII. LEVEL OF ORDINARY SKILL IN THE ART

57. My opinions have been guided by my appreciation of how a person of

ordinary skill in the art (described below) would have understood the claims of the '740 patent as of the invention date, which I have been asked to assume is the '740 patent's priority date, June 8, 2006.

58. It is my understanding that a person of ordinary skill in the art is one who is presumed to be aware of all pertinent art, thinks along conventional wisdom in the art, and is a person of ordinary creativity. I further understand that the following factors may be considered in establishing the level of ordinary skill in the art in the relevant field: the type of problems encountered in the art; the prior art solutions to those problems; the rapidity with which innovations are made; sophistication of the technology; and the educational level of active workers in the field.

59. Based on the above factors, the materials and information I have reviewed, and my extensive experience with the subject matter of the '740 patent around the date of the alleged invention, it is my opinion that a person of ordinary skill in the art related to the technology of the '740 patent would have had at least a Bachelor's degree in electrical engineering, computer engineering, or related field, and at least two years of work experience in the computer memory or data storage industry. Someone with less technical education but more experience, or more technical education but less experience, would also meet this standard.

60. Based on my education and experience in the fields of electrical engineering, computer memory, and data storage, I believe I am well qualified to provide opinions about how one of ordinary skill in the art would have interpreted and understood the claims of the '740 patent and the prior art discussed below.

VIII. CLAIM CONSTRUCTION

61. I have set forth my understanding of relevant claim construction principles above, which I have followed in my analysis. Below, I discuss Petitioners' constructions of three claim terms. In my view, the remaining claim terms should be given their plain and ordinary meaning.²

A. "LBA"

62. Independent claim 1 of the '740 patent recites "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

² I understand that the parties identified the following three terms for construction in the co-pending district court case: "optimized memory operations"; "optimally distributed"; and "memory store." Ex-1008 at 26, 32, and 35. It is my understanding that the district court found that each term should be given its plain and ordinary meaning. *Id.* at 31, 34-35, and 41.

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.” Claim 1 further recites a “set of PBA fields disposed to associate said first and second LBA sets with said first and second PBAs.”

63. Independent claim 32 also recites “a mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said first and second LBA sets including respective sets of consecutive LBAs.”

64. I understand that Petitioners have proposed that the term “LBA,” also referred to as a “logical block address,” should be construed as “an address that is part of a logical addressing system used by a host.” Based on my review of the intrinsic record, together with the knowledge of a person of ordinary skill in the art, I agree with this construction.

65. The specification of the ’740 patent expressly states that “[t]he term ‘**LBA**’, which may also be referred to herein as a logical block address, is intended to represent *an address that is part of a logical addressing system (not shown) used by a host.*” Ex-1001 at 6:37-40 (emphasis added). Accordingly, based on this definition from the specification, together with the knowledge of a person of ordinary skill in the art, I agree that the term “LBA” should be construed as “an address that is part of a logical addressing system used by a host.”

B. “PBA”

66. Claim 1 of the ’740 patent recites “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.” Claims 10-14, which directly or indirectly depend from claim 1, further recite a first and second PBA having specific identifiers.

67. Likewise, claim 32 recites a mapping table “which maps a first LBA set to a first PBA, and a second LBA set to a second PBA . . . wherein said first PBA includes a first set of access parameters and said second PBA includes a second set of access parameters, and at least one difference between said first and second PBAs.” Claim 34, which depends from claim 32, further recites a “first memory operation performed using said first PBA and a second memory operation performed using said second PBA; said first and second PBAs respectively associated with a first set of access parameters and a second set of access parameters.”

68. I understand that Petitioners have proposed that “PBA” should be construed as a “unique addressable physical memory location.” Based on my review of the intrinsic record, together with the knowledge of a person of ordinary skill in the art, I agree with this construction.

69. In particular, the specification of the '740 patent specifically states that the memory store, illustrated in Figure 2, has a set of PBA fields “disposed to *represent a unique addressable physical memory location, named ‘PBA.’*” Ex-1001 at 5:30-33 (emphasis added). This is confirmed by U.S. Patent No. 7,506,098 (“the '098 patent”), to which the '740 patent claims priority. In particular, the '098 patent states that the PBA “contains the information for both the location of the flash device . . . and the address within the flash device.” Ex-1010 at 5:29-34.

70. Accordingly, based on the description in the specification, together with the knowledge of a person of ordinary skill in the art, I agree that the term “PBA” should be construed as a “unique addressable physical memory location.”

C. “FDE” and “FDE identifier”

71. Dependent claim 9 recites the “mapping table of claim 1, wherein said set of access parameters includes . . . an FDE identifier,” while dependent claims 10, 11, and 13 recite different combinations of FDE identifiers. Dependent claim 15 further recites a “set of buses and that are controlled by a set of FDEs” wherein the “access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively controlled by different FDEs from said set of FDEs.”

72. I understand that Petitioners have proposed that “FDE” should be

construed as “a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices.” Similarly, I understand that Petitioners have proposed that “FDE identifier” should be construed as “an identifier for a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices.” Based on my review of the intrinsic record, together with the knowledge of a person of ordinary skill in the art, I agree with this construction.

73. The specification of the '740 patent discusses direct memory access engines, and explains that “these direct memory access engines may be also herein referred to as FDEs.” Ex-1001 at 3:27-35. The '740 patent also states that an FDE represents “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.” *Id.* at 3:35-39. The '098 patent, to which the '740 patent claims priority, similarly states that the disclosed systems “comprises a number of Flash DMA, or Direct Memory Access, Engines (FDEs).” Ex-1010, at 3:24-25. As the '098 patent further explains, “[a] Flash DMA Engine (FDE) is basically an intelligent DMA controller that facilitates high speed data transfers to/from a group of flash memory devices.” *Id.* at 3:25-28.

74. Accordingly, based on the description in the intrinsic record, together

with the knowledge of a person of ordinary skill in the art, I agree that the term “FDE” should be construed as “a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices.” Likewise, I agree that “FDE identifier” should be construed as “an identifier for a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices.”

D. “group identifier”

75. Dependent claim 9 recites the “mapping table of claim 1, wherein said set of access parameters includes . . . a group identifier,” while dependent claims 10, 11, and 14 recite different combinations of group identifiers. Dependent claim 15 further recites that the “access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively controlled by different FDEs from said set of FDEs.”

76. I understand that Petitioners have proposed that “group identifier” should be construed as “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE.” Based on my review of the intrinsic record, together with the knowledge of a person of ordinary skill in the art, I agree with this construction.

77. The specification expressly states that the group identifier “is an

identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE.” Ex-1001 at 6:2-6; *see also id.* at 6:11-15 (“Group identifiers are selected and assigned to flash memory devices so that no more than one flash memory device, which is within the same flash array bank interleave or which is controlled by the same FDE, is associated with the same group identifier.”). This is confirmed by the ’098 patent, which explains that “group interleaving performs parallel operations by having a specific FDE send multiple commands to different flash devices it controls.” Ex-1010, at 4:17-20.

78. Accordingly, based on the description in the specification, together with the knowledge of a person of ordinary skill in the art, I agree that the term “group identifier” should be construed as “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE.”

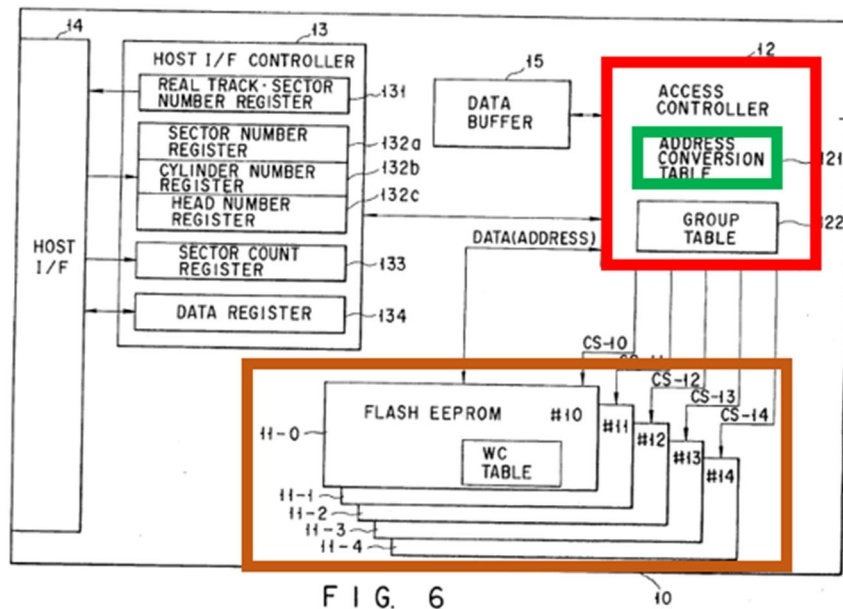
IX. OVERVIEW OF THE PRIOR ART

A. Sukegawa Discloses an LBA to PBA Mapping Table That Facilitates Optimized Operations

79. Sukegawa is directed to semiconductor memory system that uses flash memory to store data. As Sukegawa explains, prior art memory systems typically used magnetic disk drives to store data and information. Ex-1005 at 1:15-20.

However, these disk drives had a number of drawbacks, including size and susceptibility to physical impact. *Id.* at 1:59-61. As a result, Sukegawa recognizes a need for memory systems that can utilize semiconductor memory while still employing existing host drive accessing schemes. *Id.* at 4:36-47.

80. To address this need, Sukegawa discloses a semiconductor memory system that converts read and write requests from a host, using logical addresses, to “real memory addresses”—or unique addressable physical memory locations—in the flash EEPROM memory store. *See, e.g., id.* at 4:51-5:13; 9:14-19; 10:14-19. In particular, Sukegawa describes an “address conversion table,” or mapping table, that can be used by an access controller to associate logical addresses from a host with physical addresses in the memory store during a read or write operation. *Id.* at 7:26-40; 7:46-51; 8:17-19. Annotated Figure 6 of Sukegawa, shown below, illustrates the **access controller**, comprising the **address conversion table**, which converts logical address from the host to real memory addresses for flash EEPROM chips in the **memory store**:



81. As Sukegawa explains, host addresses comprise “sectors” that are part of a “logical address designated by a host system.” *Id.* at 4:67-5:4. As explained below, these logical sectors are LBAs. *See infra* § X.A.1.b. These LBAs are also grouped into sets of consecutive LBAs. For example, Sukegawa discloses sets of LBAs in which a first set comprises eight consecutive LBAs, a second set comprises eight consecutive LBAs, and so on. *See, e.g., id.* at 8:66-9:8; 8:36-39; Figure 11. The address conversion table maps these LBAs to “real memory addresses”—or PBAs—that correspond to physical memory locations in the flash EEPROM. These locations are represented in the address conversion table by various access parameters, such as identifiers for chip numbers and block numbers. *Id.* at 9:14-21.

82. Annotated Figure 11, shown below, illustrates a **first set of LBAs**

associated with a **first PBA**, and a **second set of LBAs** associated with a **second PBA**. As Figure 11 also illustrates, each LBA set is comprised of **consecutive LBAs**:

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0 ↓ 7	#10 ↓	0	(0, 1) ↓ (14, 15)
	8 ↓ 15	#11 ↓		(0, 1) ↓ (14, 15)
1	32 ↓ 39	#14 ↓	1	(0, 1) ↓ (14, 15)
	0 ↓ 7	#10 ↓		(0, 1) ↓ (14, 15)
	8 ↓ 15	#11 ↓		(0, 1) ↓ (14, 15)
2	32 ↓ 39	#14 ↓	3	(0, 1) ↓ (14, 15)
	0 ↓ 7	#10 ↓		(0, 1) ↓ (14, 15)
	8 ↓ 15	#11 ↓		(0, 1) ↓ (14, 15)

FIG. 11

83. Sukegawa teaches that the disclosed address conversion table improves performance by “enabling parallel write operations” on different physical memory

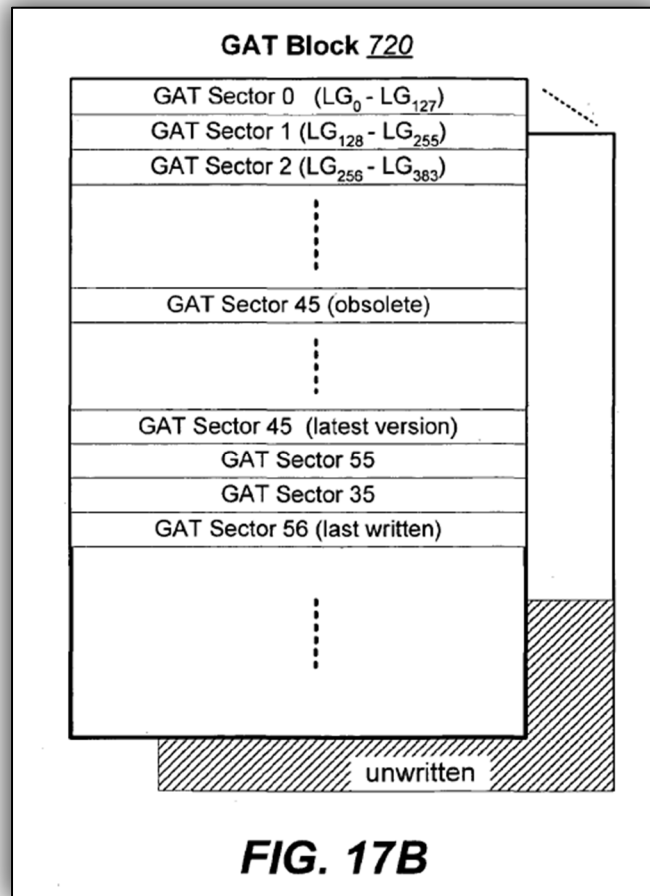
sectors in response to a request from a host. *Id.* at 8:39-43; *see also id.* at 10:20-21 (explaining that “when the host CPU specifies consecutive numbers in the same track, flash EEPROMs are accessed parallelly”); 12:36-41 (stating that “EEPROMs can be accessed parallelly when consecutive sector numbers in the same track are specified,” allowing existing disk accessing techniques to be used effectively); 4:51-57 (explaining that the invention “permit[s] parallel access to a plurality of flash EEPROMs”).

B. Bennett Discloses a Logical-to-Physical Address Mapping Table That Enables Optimized Operations in Memory Systems

84. Bennett is directed to a high-capacity and high-performance non-volatile memory system that can conduct memory operations in large blocks. Ex-1007 at 4:4-8. As Bennett explains, memory devices typically comprise one or more memory chips that include an array of memory cells for data storage. *Id.* at 2:22-26. In order to improve performance of these memory arrays, multiple charge storage elements or memory transistors are read and written together. *Id.* at 2:57-59. However, when data needs to be updated in a physical memory location, an entire physical block containing the physical location first needs to be erased and then rewritten as it is not possible to erase and rewrite only portions of the block. *Id.* at 3:4-14. This is inefficient. *Id.* Moreover, erasing and rewriting entire blocks in this manner can render existing blocks obsolete and necessitate garbage collection,

which is inefficient and causes premature aging of the memory. *Id.* at 3:65-4:1.

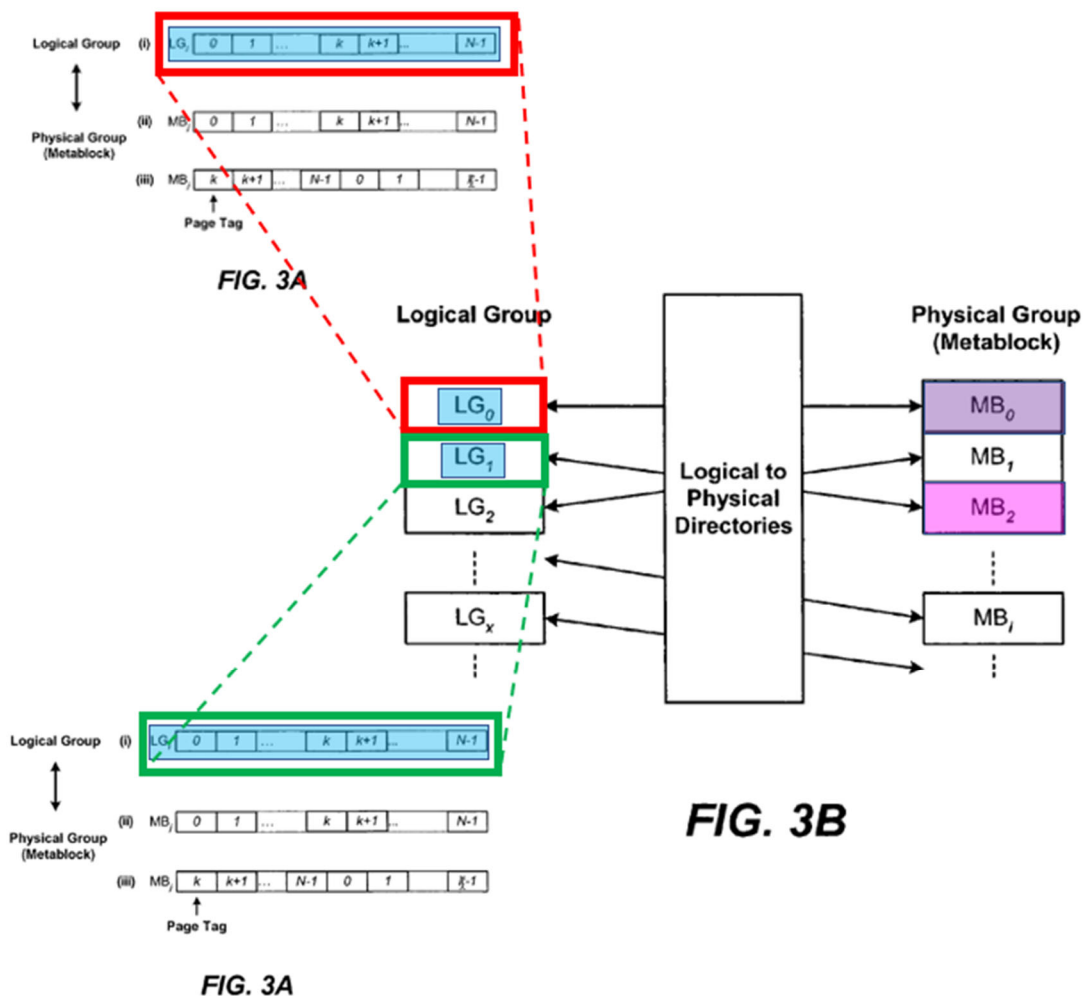
85. To address these issues, Bennett discloses a table for mapping between logical groups and physical groups. *Id.* at 23:61-64; 7:9-11; 9:55-58. For example, Bennett discloses a Group Address Table—or “GAT”—that “is a list of [physical] metablock addresses for all logical groups of host data in the memory system.” *Id.* at 24:19-21. In particular, the GAT “contains one entry for each logical group, ordered sequentially according to logical address.” *Id.* at 24:21-22; *see also id.* at 25:1-3 (“GAT sectors within a GAT block each contain logical to physical mapping information for 128 contiguous logical groups.”); 9:55-58 (explaining that the GAT tracks “mapping between logical groups of sectors and their corresponding metablocks”). Figure 17A, reproduced below, illustrates the GAT:



86. Bennett also teaches that each logical group in the GAT comprises a set of logical sectors that “are in contiguous logical order 0, 1, . . . , N-1.” *Id.* at 7:13-15; 6:61-65 (noting that the host issues a command “to read or write a segment containing a string of logical sectors of data with contiguous addresses”). As explained below, these logical sectors are LBAs. *See infra* § X.B.1.b. Moreover, groups or sets of these contiguous logical sectors are mapped to unique physical “metablocks”—i.e., PBAs, as explained below—and the “mapping information is

maintained in a set of logical to physical directories.” *Id.* at 7:38-44; *see also id.* at 23:61-64 (explaining that the GAT maps “between logical groups and physical groups (metablocks)”).

87. Annotated Figures 3A and 3B of Bennett, shown below, illustrate a **first logical field** (LG₀) comprising **consecutive LBAs** associated with a **first PBA** (MB₀), and a **second logical field** (LG₁) comprising **consecutive LBAs** associated with a **second PBA** (MB₂):



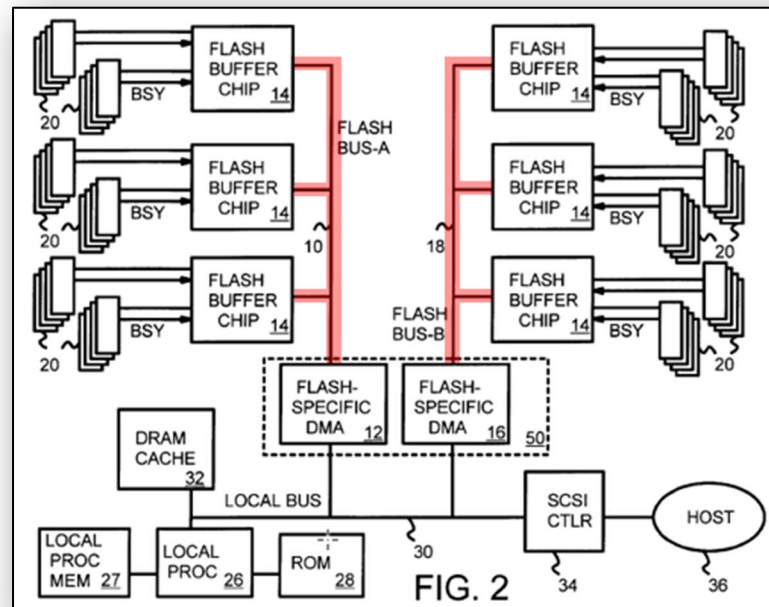
88. Bennett also teaches that the disclosed mapping table results in optimized memory operation. For example, Bennett discloses that “[i]n order to maximize programming speed and erase speed, parallelism is exploited as much as possible by arranging for multiple pages of information . . . to be programmed in parallel, and for multiple MEUs [minimum erasable units] to be erased in parallel.” Ex-1007 at 8:26-30; *see also id.* at 2:57-59 (“In order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in parallel.”). Thus, the mapping table disclosed in Bennett provides a more “efficient use of system resources” by allowing “multiple logical groups to be updated concurrently,” which “increases efficiency and reduces overhead.” *Id.* at 4:50-52.

C. Bruce Discloses Specific Access Parameters for Solid State Memory Systems, Each Having Unique Identifiers

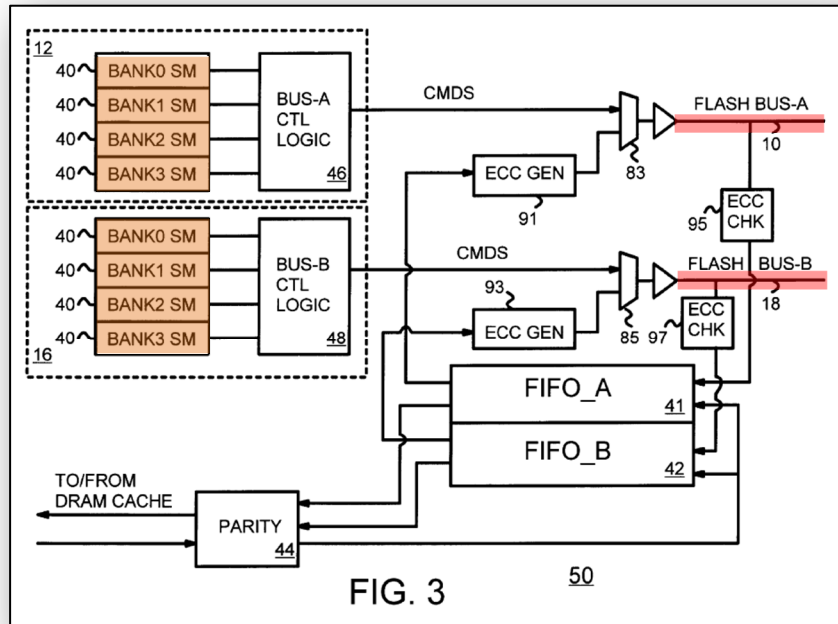
89. Bruce is directed to non-volatile memory storage systems and expansion of flash memory systems. Ex-1006 at 1:18-20. According to Bruce, in previous flash memory systems, the potential for expansion was hampered by issues such as rigid bus architectures. *Id.* at 1:39-43. While direct-memory access (DMA) was known and used to facilitate data transfers, it was not well-suited to transfer addresses and commands required by flash memory chips. *Id.* at 2:22-30. Accordingly, Bruce refers to the need for a modified DMA controller allowing

commands and addresses to be input over shared data/address/command pins. *Id.* at 2:40-42. According to Bruce, there was also need for a flash memory system allowing greater expansion and parallel flash operations across all flash chips. *Id.* at 2:42-46.

90. Bruce thus discloses flash-specific DMA controllers coupled to flash buses that have shared lines for transmitting the sequence of command bytes and address bytes from the DMA controller. *Id.* at 2:56-59. Each flash bus in Bruce is connected to a flash specific DMA controller, as well as an array of flash memory chips (through a flash buffer chip). *Id.* at 2:66-3:1. These flash buses are uniquely identified as “**flash bus 10/Flash Bus-A**” and “**flash bus 18/Flash Bus-B**,” as illustrated in annotated Figure 2, below:



91. Bruce further discloses that each flash DMA controller has a plurality of state machines, each of which control one of four banks of flash-memory chips. *Id.* at Abstract; *see also id.* at 11:23-25. These state machines allow data to 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; 13:32-35. Each of these state machines are separately identified as “**Bank0 SM**,” “**Bank1 SM**,” “**Bank2 SM**,” and “**Bank3 SM**,” as illustrated in annotated Figure 3 below:



92. In addition, Bruce discloses that each individual bank of flash-memory chips “can be separately accessed, allowing many flash operations to be performed in parallel.” *Id.* at 5:21-23; 6:55-57. This is facilitated by the fact that the flash memory chips connected to each flash buffer are “uniquely addressed.” *Id.* at 7:62-67. Accordingly, the DMA controller sends specific device addresses for each chip to the flash buffer, which then decodes the address into chip selects. *Id.* at 8:46-49; 10:17-19. As a result, these unique addresses allow one flash memory device to be identified from another flash memory device from a set of flash devices that are controlled by the same DMA state machine.

93. According to the specification, Bruce provides “[a] high-performance

flash-memory system performing flash operations in parallel . . . even when expansion flash chips are added.” *Id.* at 2:44-46; *see also id.* at 13:32-35 (stating that the invention provides “inherent parallelism [that] improves performance by allowing multiple flash operations to be performed at the same time”). This improves performance of the memory system. *Id.* at 13:40-44; *see also id.* at 5:44-50.

X. THE PRIOR ART DISCLOSES ALL OF THE CLAIMED ELEMENTS OF THE ’740 PATENT

A. Each of the Challenged Claims Is Anticipated or Rendered Obvious by Sukegawa

94. It is my opinion that Sukegawa discloses, and therefore anticipates, every element of claim 1. It is also my opinion that Sukegawa in combination with Bruce renders obvious every element of claims 9-15, 32, and 34.

1. Claim 1 Is Anticipated by Sukegawa

95. As explained below, it is my opinion that Sukegawa discloses, and therefore anticipates, every element of claim 1.

a. [Claim 1 Pre]: “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:”

96. To the extent the preamble is limiting, it is my opinion that Sukegawa discloses this claim element. In particular, Sukegawa discloses an “address

conversion table” that converts logical addresses specified by a host into “real memory addresses” in flash EEPROM chips. Ex-1005 at 10:14-19; *see also id.* at 7:48-52. Sukegawa’s address conversion table is therefore a “mapping table” as recited in the preamble.

97. An illustration of Sukegawa’s mapping table, which correlates logical addresses to real memory addresses, is shown in Figure 11 excerpted below:

ADDRESS
CONVERSION
TABLE

121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0 ↓	0	#10	0	(0, 1)
	↓ 7	↓		(14, 15)
	8	#11		(0, 1)
	↓ 15	↓		(14, 15)
	~			
	32	#14		(0, 1)
	↓ 39	↓		(14, 15)
	1 ↓	0		#10
↓ 7		↓	(14, 15)	
8		#11	(0, 1)	
↓ 15		↓	(14, 15)	
~				
32		#14	(0, 1)	
↓ 39		↓	(14, 15)	
2 ↓		0	#10	3
	↓ 7	↓	(14, 15)	
	8	#11	(0, 1)	
	↓	↓	(14, 15)	

FIG. 11

98. Sukegawa also teaches that the disclosed mapping table enables optimized memory operation by allowing flash EEPROMS to be “accessed parallelly.” *Id.* at 12:35-40;³ *see also id.* at 4:51-57 (explaining that the “primary object of the present invention [is] to provide a semiconductor memory system which is designed to permit parallel access to a plurality of flash EEPROMs”); *id.* at 10:20-21 (stating that the disclosed systems enable “flash EEPROMS [to be] accessed parallelly”); 5:8-13 (discussing means for assigning sequential number sets “so as to permit the memory access means to parallelly access the plurality of flash EEPROM chips”); 10:10-13 (teaching that “transfer to flash EEPROM chips 11-0 through 11-2 and the writing of data into flash EEPROM chips 11-0 through 11-2 are practically performed simultaneously.”); *id.* at 10:20-21 (stating that the

³ The '740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.,* Ex-1001 at 2:14-21 (“***These optimizing memory operations include*** increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally distributed across different storage device resources, ***such as by interleaving or parallel memory operations***”) (emphasis added).

disclosed systems enable “flash EEPROMS [to be] accessed parallelly”).

99. The district court’s claim construction order also states that examples of optimization “include[e] contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one access parameter.” Ex-1008 at 30. As explained below, Sukegawa teaches each of these examples. *Infra* §§ X.A.1.b, X.A.1.c. Thus, pursuant to the district court’s claim construction, these aspects of Sukegawa further demonstrate that Sukegawa’s disclosed mapping table enables optimized memory operation.

100. Sukegawa explains that these optimized memory operations are also performed in response to receiving an I/O transaction request initiated by a host, explaining that the controller accesses memory for read and write transactions “in response to an [sic] disk access request supplied from a host CPU.” *Id.* at 7:28-32; *see also id.* at 11:3-6 (explaining that “[w]hen a write access request for a specified address (logical address) occurs, the block corresponding to the specified address is accessed on the basis of the address conversion table 121”).

101. For at least these reasons, it is my opinion that Sukegawa discloses “[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.”

b. [Claim 1a]: “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;”

102. It is my opinion that Sukegawa discloses this claim element. In particular, Sukegawa discloses logical “sectors” that are part of “a logical address designated by a host system.” Ex-1005 at 4:67-5:4; *see also id.* at 9:14-19 (explaining that “logical addresses . . . specified by the host CPU” comprise track numbers and sector numbers); 4:30-35 (same); 8:48-49 (noting that “the data size of a sector is usually 512 bytes”).⁴ Accordingly, each of Sukegawa’s logical sectors is an LBA—or “an address that is part of a logical addressing system used by a host”—pursuant to Petitioners’ proposed construction of LBA.

103. Sukegawa also discloses logical fields that are disposed to represent a plurality of LBA sets, in which each LBA set represents a set of consecutive LBAs. For example, Sukegawa discloses sets of LBAs (logical sectors) in which a first LBA set comprises eight consecutive LBAs from 0-7, a second set comprises eight

⁴ The ’740 patent confirms that an LBA typically has a “size equal to 512 bytes.” Ex-1001 at 7:22-25. This further confirms that Sukegawa’s logical sector corresponds to an LBA.

consecutive LBAs from 8-15, and so on, all the way up a set that comprises eight consecutive LBAs from 32-39. *See, e.g., id.* at 8:66-9:8; 8:36-39 (noting that a write unit from the host CPU “is allocated consecutive sector numbers in the same track”); Figure 11 (illustrating sets of consecutive LBAs or sectors); claim 9 (reciting a semiconductor memory system for storing “address information representing how consecutive logic addresses corresponding to one track are assigned to physical addresses”). In this manner, Sukegawa discloses a “set of logical fields, including a first logical field and a second logical field” that are “disposed for representing a plurality of LBA sets,” in which each LBA set “represent[s] a set of consecutive LBAs.”

104. Annotated Figure 11, shown below, illustrates a **first logical field** and a **second logical field**, each of which are disposed to represent **a set of consecutive LBAs**:

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0 ↓ 7	#10 ↓	0	(0, 1) ↓ (14, 15)
	8 ↓ 15	#11 ↓		(0, 1) ↓ (14, 15)
	32 ↓ 39	#14 ↓		(0, 1) ↓ (14, 15)
1	0 ↓ 7	#10 ↓	1	(0, 1) ↓ (14, 15)
	8 ↓ 15	#11 ↓		(0, 1) ↓ (14, 15)
	32 ↓ 39	#14 ↓		(0, 1) ↓ (14, 15)
2	0 ↓ 7	#10 ↓	3	(0, 1) ↓ (14, 15)
	8 ↓	#11 ↓		(0, 1)

FIG. 11

105. For at least these reasons, it is my opinion that Sukegawa discloses “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.”

- c. **[Claim 1b]: “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”**

106. It is my opinion that Sukegawa discloses this claim element. Sukegawa discloses “real memory addresses” that correspond to physical memory locations represented by identifiers for chip numbers, block numbers, and page numbers. *See, e.g.*, Ex-1005 at 9:14-21; 4:30-35 (explaining that the “real memory address” is the address “by which the flash EEPROMS in the semiconductor disk drive are selectively accessed”); Figure 11 (illustrating the chip numbers, block numbers, and page numbers for the “real memory addresses”); claim 9 (reciting that logical addresses “are assigned to physical address of the [] flash EEPROM chips”). Accordingly, each “real memory address” is a PBA—or a “unique addressable physical memory location”—within a memory store pursuant to Petitioners’ proposed construction of PBA.

107. Sukegawa also discloses sets of PBA fields that are disposed to

represent a PBA. In particular, as explained above, Sukegawa teaches that the PBAs (real memory locations) comprise fields for identifying aspects of a memory location, such as a chip number, block number, and page number. *See id.* at 9:14-21; Figure 11. As a result, each PBA is “disposed to represent a set of access parameters” representing the physical memory location, such as identifiers for a specific chip number and block number. *See id.* at 9:14-21; Figure 11.

108. Sukegawa’s address conversion table also associates the first and second LBA sets with first and second PBAs. For example, Sukegawa explains that the address conversion, or mapping, table provides a “correspondence between the logical addresses . . . specified by the host CPU and the real memory addresses.” *Id.* at 9:14-19; *see also id.* at 10:14-19 (explaining that the address conversion table enables “converting the logical addresses from the host CPU into the real memory address”); 5:46-51 (stating that the stored address conversion information is “used for converting the logical address from the host system into the real memory address”); claim 1 (reciting means “for converting the logical address designated by a host system, into the real memory address for accessing at least one of said plurality of flash EEPROM chips”).

109. Annotated Figure 11, shown below, illustrates a set of **PBA fields** that includes a first and second PBA field (e.g., chip number and block number), which

represent a **first PBA** and a **second PBA**. Annotated Figure 1 further illustrates that the **first logical field** is associated with the **first PBA**, and the **second logical field** is associated with the **second PBA**:

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0	#10	0	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)
1	32	#14	1	(0, 1)
	39	↓		(14, 15)
	0	#10		(0, 1)
	7	↓		(14, 15)
2	8	#11	3	(0, 1)
	15	↓		(14, 15)
	32	#14		(0, 1)
	39	↓		(14, 15)
3	0	#10	3	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)

FIG. 11

110. For at least these reasons, it is my opinion that Sukegawa discloses “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.”

- d. **[Claim 1c]: “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.”**

111. It is my opinion that Sukegawa discloses this claim element. Sukegawa teaches that the disclosed mapping table causes an electronic storage device to perform optimized memory operations by “enabling parallel write operations” on different sectors in response to request from a host.⁵ Ex-1005 at 8:39-43. For

⁵ The ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.*, Ex-1001 at 2:14-21 (“***These optimizing memory operations include*** increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally

example, Sukegawa explains that “consecutive sector numbers are allocated to flash EEPROM chips 11-0 through 11-4 . . . and *the contents of the allocation are stored in the address conversion table 121 as the address conversion information* for converting the logical address from the host CPU into the real memory address. *Thus*, when the host CPU specifies consecutive numbers in the same track, *flash EEPROMs are accessed parallelly*.” *Id.* at 10:14-21 (emphasis added); *see also id.* 12:36-41 (indicating that “EEPROMs can be accessed parallelly when consecutive sector numbers in the same track are specified,” allowing existing disk accessing techniques to “be used effectively”); 4:51-57 (explaining that the invention “permit[s] parallel access to a plurality of flash EEPROMs”); 5:8-13 (discussing means for assigning sequential number sets “so as to permit the memory access means to parallelly access the plurality of flash EEPROM chips”).

112. Sukegawa further teaches that “transfer to flash EEPROM chips 11-0 through 11-2 and the writing of data into flash EEPROM chips 11-0 through 11-2 are practically performed *simultaneously*.” *Id.* at 10:10-13 (emphasis added); *see also id.* at 9:35-38 (explaining an example in which a write transaction can

distributed across different storage device resources, *such as by interleaving or parallel memory operations*”) (emphasis added).

encompass 23 LBAs or logical sectors of data, which means that each set of LBAs (sets 0-7, 8-15, and 16-23) are written to PBAs); Figure 11. Accordingly, through its teaching of parallel and simultaneous memory operations, which are enabled by the mapping table, Sukegawa discloses that the mapping table performs optimized memory operations on memory locations in the flash EEPROM chips.

113. Further, as noted above, the district court's claim construction order states that examples of optimization "include[] contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA's differing by at least one access parameter." Ex-1008 at 30. Sukegawa teaches each of these examples. *Supra* §§ X.A.1.b, X.A.1.c. Accordingly, these aspects of Sukegawa further demonstrate that Sukegawa's disclosed mapping table enables optimized memory operation pursuant to the district court's claim construction.

114. Sukegawa also indicates that such optimization occurs in response to receiving an I/O transaction request, noting that when an address is specified "by the host CPU," the access controller sets chip select signals base on the conversion table, thereby enabling the memory to "undergo read or write access." Ex-1005 at 9:22-32; 7:28-32, 11:3-6

115. For at least these reasons, it is my opinion that Sukegawa discloses that "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.”

2. Claim 9 Is Rendered Obvious by Sukegawa in View of Bruce

116. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 9 of the '740 patent. In particular, a person of ordinary skill in the art would have been motivated to combine Sukegawa's mapping table with Bruce's approach to an “expandable architecture,” which provides “inherent parallelism [that] improves performance,” to include a “set of access parameters include[ing] a bus identifier, an FDE identifier and a group identifier.” *See* Ex-1006 at 5:45-47.

117. Sukegawa and Bruce expressly recognize issues related to spinning disk drives, as well as the benefits of solid state non-volatile storage. *See* Ex-1005 at 1:21-28, 52-58 (noting that magnetic disk drives have “mechanical moving portions, such as the rotatable disk and the magnetic head, [that] may malfunction or be damaged” in the event of physical shock, and that make it “difficult to make the entire disk drive compact”); Ex-1006 at 1:35-42 (noting that hard drives “suffer from their inherent fixed size” and cannot be expanded). At the same time, both Sukegawa and Bruce also recognize that solid state non-volatile storage had certain

shortcomings related to data management, host addressing schemes, and the erasure and programming of information. Ex-1005 at 4:26-47; Ex-1006 at 1:45-52. Accordingly, Sukegawa and Bruce are both directed to improving the function of solid state non-volatile memory by resolving these shortcomings in similar ways, thereby optimizing memory performance. Ex-1005 at 4:51-6:5; Ex-1006 at 2:40-3:18.

118. For example, both Sukegawa and Bruce are directed to making, using, and/or enabling systems and methods for optimizing memory operations through the use of interleaving and parallel memory operations. As noted above, the “primary object” of Sukegawa is “to provide a semiconductor memory system which is designed to permit *parallel access* to a plurality of flash EEPROMs.” Ex-1005 at 4:51-57 (emphasis added); *see also id.* at 10:20-21 (stating that the disclosed systems enable “flash EEPROMS [to be] accessed parallelly”); 5:8-13 (discussing means for assigning sequential number sets “so as to permit the memory access means to parallelly access the plurality of flash EEPROM chips”). Similarly, Bruce’s stated objective is to provide “[a] high-performance flash-memory system performing flash operations in *parallel* . . . even when expansion flash chips are added.” Ex-1006 at 2:44-46. This “inherent *parallelism* improves performance by allowing multiple flash operations to be performed at the same time.” *Id.* at 13:32-35 (emphasis

added); *see also id.* at 13:45-46 (noting that the disclosed system “is efficient yet allows multiple flash operations to occur at the same time.”).

119. Because Sukegawa and Bruce are in the same field, teach analogous solutions to the same issues, and have the same purpose of enabling and improving parallel memory operations, a POSITA would have been motivated to combine these references. In particular, a POSITA would have been motivated to incorporate Bruce’s physical access parameters—which enable an expandable system architecture—into Sukegawa’s mapping table to provide the benefit of a mapping table covering an expandable solid state storage system capable of improving parallel memory access. This motivation stems from the references themselves, which are both directed to improving and increasing parallel memory operations. For example, Bruce explains that the addition of flash memory chips in a memory device increases busy signals. Ex-1006 at 2:9-11. Moreover, connecting busy signals “can limit performance as parallel flash operations driving the same busy line can interfere with each other” and can also “limit[] parallelism by not providing separate indications of when each flash-memory chip has completed an operation.” *Id.* at 2:12-21. Accordingly, the benefits provided by Bruce align seamlessly with the purpose of Sukegawa. Thus, a POSITA would have been motivated to combine Bruce’s additional access parameters with Sukegawa’s mapping table, which already

contains access parameters as explained above, as this would have allowed Sukegawa's mapping table to further leverage the benefits of solid state non-volatile memory connected to a host system while maintaining and enhancing parallel memory operations.

120. Moreover, including the additional access parameters taught by Bruce in Sukegawa's mapping table would simply combine well-known prior art elements and techniques, according to known methods, to yield the predictable result of a mapping table that accounts for an expandable solid state storage device. This is confirmed by the fact that Sukegawa already teaches a table comprising fields for access parameters, and Bruce simply expands on types of access parameters that can be added to such a table. In addition, Bruce's teachings provide known techniques that can be used in Sukegawa's mapping table in the same way by, for example, providing a flexible mapping table that can translate conventional host commands while also providing the ability to account for a device with expandable capacity.

121. Including the access parameters taught by Bruce in Sukegawa's mapping table would have also been straightforward and predictable, and would have required minimal modification. Like Bruce, Sukegawa recognizes that address translation between the host address system and the solid state memory must be handled independently from the host and within the solid state memory device. *See,*

e.g., Ex-1005 at 7:6-32, 7:46-57, Figure 6, and Figure 11; Ex-1006 at 2:40-55. Accordingly, incorporating Bruce's additional access parameters would simply require altering Sukegawa's mapping table to add additional physical access parameters to account for limited additional hardware in an expandable system, such as distinct buses, DMA engines, and flash chip groups.

122. Moreover, a person of ordinary skill in the art would have had a very reasonable expectation of success in doing so because modifying Sukegawa's mapping table with Bruce's access parameters would require only minor changes to the table while still maintaining the same ability to receive commands over the same type of host interface. A person of ordinary skill in the art would have understood that no external change or new host command and/or interface design would have been required.

123. In my opinion, a motivation to combine Sukegawa with Bruce is further provided by the fact that Sukegawa is explicitly *identified on the face of Bruce* as shown below:

		 US005822251A	
United States Patent [19]		[11] Patent Number:	5,822,251
Bruce et al.		[45] Date of Patent:	Oct. 13, 1998
I			
[54]	EXPANDABLE FLASH-MEMORY MASS-STORAGE USING SHARED BUDDY LINES AND INTERMEDIATE FLASH-BUS BETWEEN DEVICE-SPECIFIC BUFFERS AND FLASH-INTELLIGENT DMA CONTROLLERS	5,535,328 7/1996 Harari et al.	395/182.05
		5,559,956 9/1996 Sukegawa	395/182.06
		5,568,439 10/1996 Harari	365/218
		5,572,466 11/1996 Sukegawa	365/185.33
		5,594,883 1/1997 Pricer	395/440
		5,602,987 2/1997 Harari et al.	395/182.06
		5,603,001 2/1997 Sukegawa et al.	395/430
		5,606,529 2/1997 Honma et al.	365/230.01
[75]	Inventors: Ricardo H. Bruce, Union City; Rolando H. Bruce, South San Francisco; Earl T. Cohen, Fremont, all of Calif.	5,619,470 4/1997 Fukumoto	365/228
		5,627,783 5/1997 Miyauchi	365/185.33
		5,640,349 6/1997 Kakinuma et al.	365/185.33
[73]	Assignee: Bit Microsystems, Inc., Fremont, Calif.	Primary Examiner—Terrell W. Fears Attorney, Agent, or Firm—Stuart T. Auvinen	

In my view, the fact that Bit Microsystems, Inc. (“Bit Microsystems”)—Bruce’s applicant—specifically cites Sukegawa evidences that Bit Microsystems viewed Sukegawa as part of the same field, and relevant to the same issues and solutions, addressed by Bruce. It further confirms that a POSITA would have considered both references together when addressing similar issues, as Bit Microsystems itself appears to have done.

124. This is also underscored by the fact that Bruce is a continuation-in-part of the ’006 patent, which is titled “Unified Re-Map and Cache-Index Table With Dual Write-Counters For Wear-Leveling of Non-Volatile Flash Ram Mass Storage.” Ex-1009. Like Sukegawa, the ’006 patent is directed to a re-mapping “*table* for a flash-memory system [that] has a *plurality of entries*.” *Id.* at 2:62-64 (emphasis

added); *id.*, Abstract. These entries have “*a physical-block-address field* that contains a *physical block address* of a block in an array of flash memory devices.” *Id.* at 2:65-67 (emphasis added).

125. Moreover, the purpose of the '006 patent is to provide a more “efficient” table “to minimize excess writes to flash memory while re-mapping or swapping addresses” in physical memory. Ex-1009 at 2:53-59; 5:48-52 (noting that “[f]lash blocks can be swapped and frequently overwritten addresses allocated to different flash blocks over time.”). Similarly, one of Sukegawa’s stated objectives is to provide “rewrite frequency for the individual flash EEPROM chips using a swapping process.” Ex-1005 at 4:59-63; *see also id.* at 3:27-34, 4:14-19 (noting that “rewrite count is limited” in flash memory and the “swapping process prevents the concentration of rewriting to a specific block to thereby provide the average rewrite frequency for the individual blocks). Accordingly, both the '006 patent (Bruce’s parent patent) and Sukegawa are directed to resolving the same issues in a similar manner. In my opinion, this further confirms that a POSITA would have been motivated to combine Sukegawa with Bruce.

126. In summary, a POSITA would have simply applied Bruce’s known access parameters to Sukegawa’s table—which also includes access parameters—to alter Sukegawa’s table to include an expandable system. As a result, it is my opinion

that it would have been obvious to a POSITA to combine Sukegawa with Bruce.

- a. **[Claim 9]: “The mapping table of claim 1, wherein said set of access parameters includes a bus identifier, an FDE identifier and a group identifier.”**

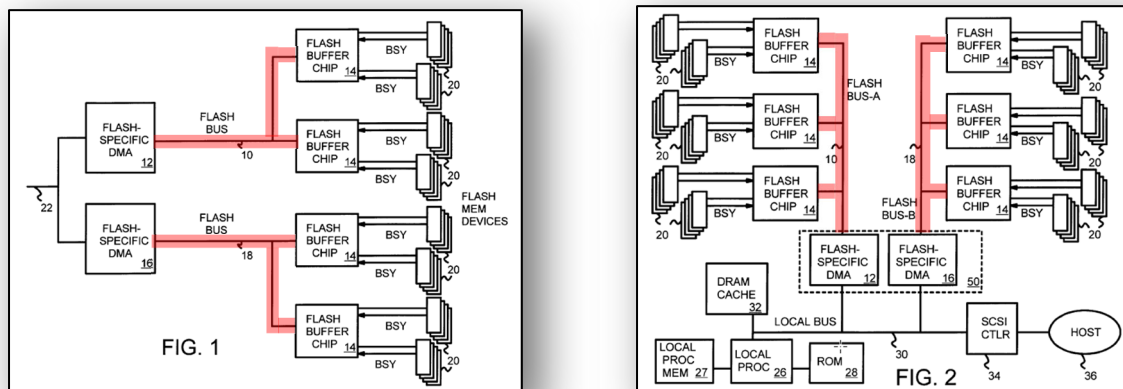
127. As explained above, it is my opinion that Sukegawa discloses all the elements of claim 1. *See supra* § **Error! Reference source not found..** It is also my opinion that Sukegawa in combination with Bruce discloses all the elements of claim 9, thereby rendering claim 9 obvious.

- i. **Bruce Discloses Multiple Buses With Bus Identifiers**

128. It is my opinion that Bruce discloses access parameters in the form of bus identifiers. Bruce discloses two flash buses “coupled to a flash-specific DMA controller,” which have “shared lines that transmit the sequence of command bytes and address bytes from the flash-specific DMA controller.” Ex-1006 at 2:56-59; *see also id.* at 3:62-64 (explaining that a first flash bus “transfers data, addresses, and commands over shared address/data/command lines”); 4:10-21 (explaining that a second flash bus does the same). The flash buses are also coupled to flash buffer chips, which pass commands and addresses from the flash-specific DMA controller to the flash memory. *Id.* at 2:63-65; 5:7-12 (explaining that each flash bus has eight shared data/address/command lines, as well as two lines for an encoded command). Because the flash buses are connected to different controllers, each flash bus “can

operate at the same time, allowing flash operations to be initiated and processed in parallel.” *Id.* at 5:18-20.

129. Bruce identifies these buses using the identifiers “**flash bus 10/Flash Bus-A**” and “**flash bus 18/Flash Bus-B**,” which are illustrated in annotated Figures 1 and 2, below:

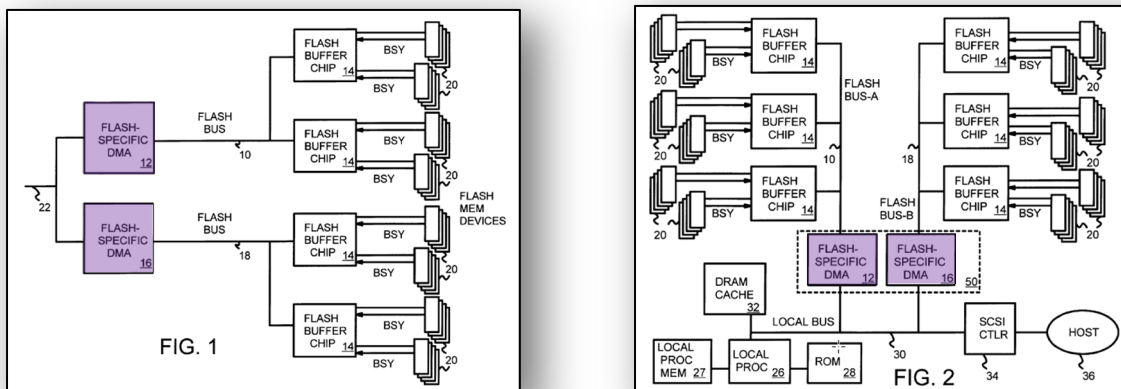


130. As a result, it is my opinion that Bruce’s disclosure of **flash bus 10/Flash Bus-A** and **flash bus 18/Flash Bus-B** teaches “bus identifiers” associated with a PBA, or a “unique addressable physical memory location” pursuant to Petitioners’ construction.

ii. Bruce Discloses Multiple FDEs With FDE Identifiers

131. It is my opinion that Bruce discloses access parameters in the form of FDE identifiers that comprise identifiers for specific DMA state machines. Bruce discloses that the flash buses are “coupled to [a] flash specific DMA controller,” Ex-

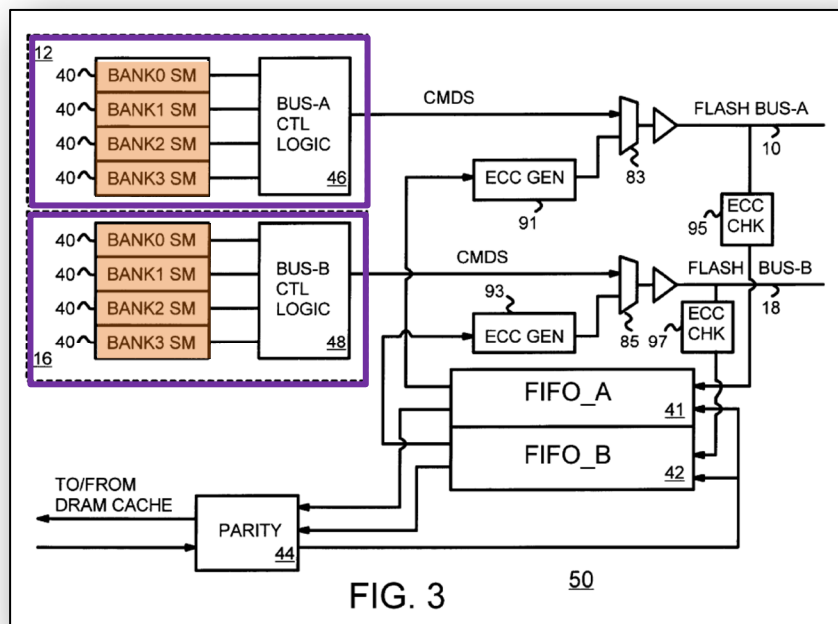
1006 at 2:56-57, which “generates command, address, and data sequences” to the flash memory chips, *id.* at 3:59-62; 4:8-10; 5:2-6 (explaining that the DMA controllers not only transfer data, “but also generate the appropriate command and address bytes that are transmitted over the same lines of the flash bus as the data”). Annotated Figures 1 and 2, below, illustrate dual **flash-specific DMA controllers 12 and 16**:



132. Bruce further discloses that each DMA controller contains four “state machines,” each of which control one of four banks of flash-memory chips. *Id.* at Abstract; *see also id.* at 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-55. Having four DMA state machines “allows four flash-memory chips to be accessed at once for each flash bus, for a total of eight flash accesses.” *Id.* at 6:55-57. Accordingly, the DMA state machines allow data to 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; 13:32-35.

133. Annotated Figure 3, shown below, illustrates the four **state machines** (identified as “**Bank0 SM**,” “**Bank1 SM**,” “**Bank2 SM**,” and “**Bank3 SM**”) within each flash specific **DMA controller**, thereby providing a total of eight state machines.

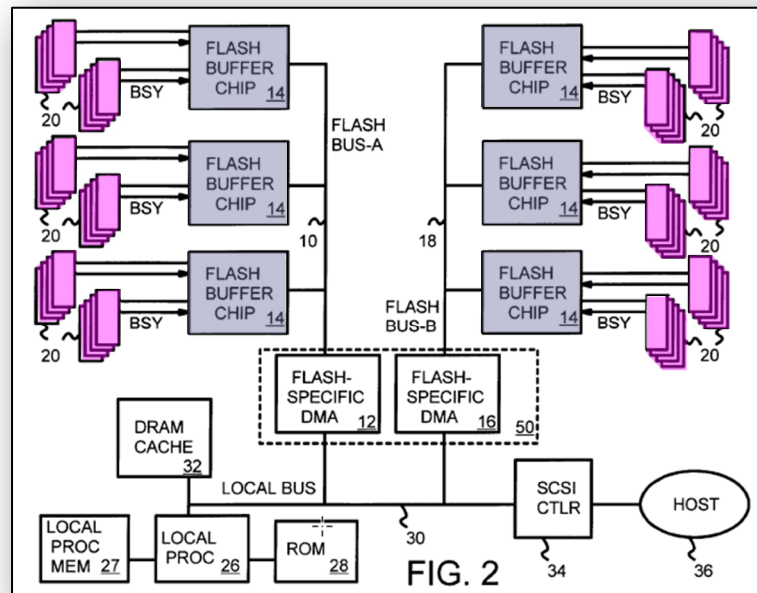


134. Accordingly, it is my opinion that, by disclosing multiple state machines that control different banks of memory chips—each of which has its own designation or “identifier” (e.g., “Bank0 SM,” etc.)—Bruce discloses “an identifier for a component that controls multiple flash devices and facilitates high speed data

transfers to and from a group of flash memory devices” pursuant to Petitioners’ proposed construction of FDE identifier. As a result, these state machines are FDEs pursuant to Petitioners’ proposed construction. Bruce also teaches that each of the FDEs are associated with a PBA, or “unique addressable physical memory location” pursuant to Petitioners’ construction of PBA. *See, e.g., id.* at 6:53-57; 11:23-25. For at least this reason, it is my opinion that Bruce discloses an “FDE identifier” associated with a PBA.

iii. Bruce Discloses Multiple Groups with Group Identifiers

135. It is my opinion that Bruce also discloses access parameters in the form of group identifiers. As noted above, Bruce discloses multiple memory banks and teaches that up to “four banks of flash-memory chips” can be connected to one of a plurality of flash buffer chips. Ex-1006 at 12:59-62. Each bank comprises a plurality of flash memory chips. *Id.* at 2:66-3:1. For example, each memory bank may contain eight flash memory chips, leading to a total of 32 flash-memory chips (8 chips x 4 banks) per flash buffer. *Id.* at 5:31; 5:58-61. The flash memory **chip banks**, which are connected through a plurality of **flash buffer chips**, are illustrated in annotated Figure 2, below.



136. Bruce further teaches that each individual bank of flash-memory chips “can be separately accessed, allowing many flash operations to be performed in parallel.” *Id.* at 5:21-23; 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”). Separate access is enabled by the fact that the flash memory chips connected to each flash buffer are “uniquely addressed.” *Id.* at 7:62-67. Accordingly, to access the memory, the DMA controller sends specific device addresses for each chip to the flash buffer, which then decodes the address into chip selects. *Id.* at 8:46-49; 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); 2:40-42 (noting that, in the invention, a DMA controller inputs “commands and addresses . . . to a flash-memory

chip”).

137. Further, as explained above, each FDE (e.g., state machine) controls an entire memory bank, which comprises a plurality of flash chips. Accordingly, by teaching that each chip in a memory bank has a unique address sent by the DMA controller, Bruce teaches the use of group identifiers—namely, “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE” pursuant to Petitioners’ proposed construction. Bruce also discloses that the group identifiers are associated with a PBA, or a “unique addressable physical memory location” pursuant to Petitioners’ construction. *See, e.g., id.* at 8:46-49; 10:17-19; 2:40-42.

138. For at least these reasons, it is my opinion that Bruce discloses group identifiers associated with a physical memory location.

**iv. It Would Have Been Obvious to Modify
Sukegawa’s Mapping Table to Include the
Identifiers Disclosed in Bruce**

139. As discussed above, it is my opinion that Bruce discloses access parameters that include bus identifiers, FDE identifiers, and group identifiers. Moreover, as previously explained, it is my opinion that it would have been obvious for a person of ordinary skill in the art to combine Sukegawa with Bruce. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in

combination with Bruce renders obvious the “mapping table of claim 1, wherein said set of access parameters includes a bus identifier, an FDE identifier and a group identifier.”

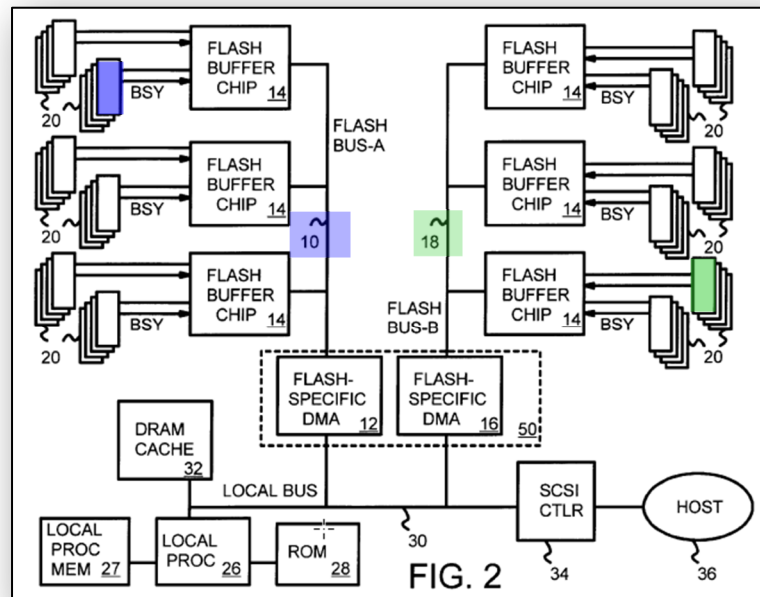
3. Claim 10 Is Rendered Obvious by Sukegawa in View of Bruce

140. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 10.

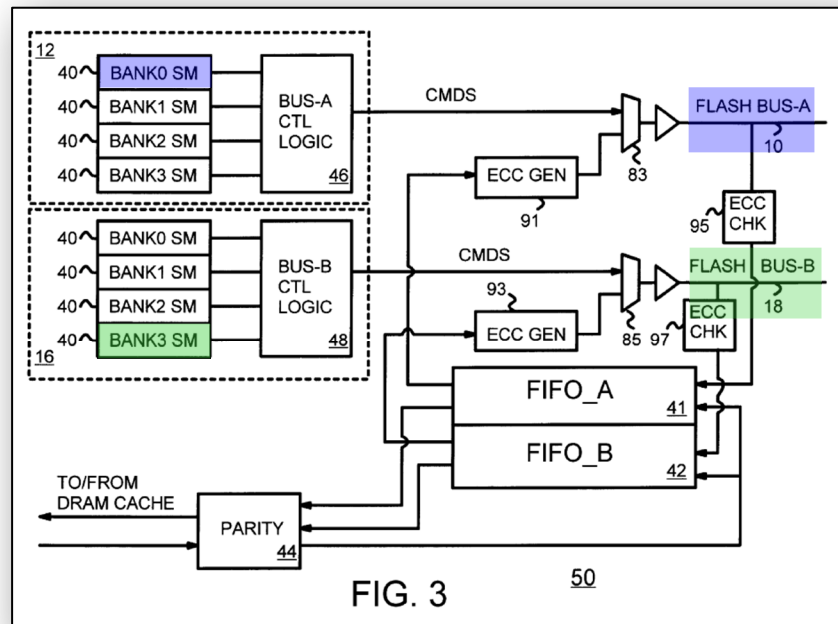
[Claim 10]: “The mapping table of claim 9, wherein said first and second PBA include different bus identifiers, different FDE identifiers and different group identifiers.”

141. It is my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 9. *See supra* § X.A.2. It is also my opinion that Sukegawa and Bruce also teach all of the elements of claim 10, thereby rendering claim 10 obvious.

142. As explained above, Bruce teaches the use of bus identifiers, FDE identifiers, and group identifiers. *See supra* § X.A.2. Bruce also teaches that these identifiers may be different for a first and second memory chip. For example, Bruce demonstrates that a PBA (i.e., a unique addressable physical memory location) for a **first memory chip** on bus 10 would have one bus identifier (“**bus 10**”), while a PBA for a **second memory chip** on bus 18 would have a different bus identifier (“**bus 18**”) as illustrated in annotated Figure 2, below:



143. As explained above, Bruce also teaches that each bank of memory devices is controlled by its own FDE engine (i.e., DMA state machine). *See* Ex-1006 at Abstract; 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-57; *see also supra* § X.A.2.a.ii. Thus, for example, a PBA on a **first memory chip** located on **bus 10** could have one FDE identifier (e.g., “**Bank0 SM**”), while a PBA for a **second memory chip** located on **bus 18** could have a different FDE identifier (e.g., “**Bank3 SM**”), as illustrated in annotated Figure 3, below:



144. Likewise, Bruce teaches that each chip within a memory bank has a unique address that allows it to be separately accessed. *Id.* at 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”); 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); *see also supra* § X.A.2.a.iii. Accordingly, a PBA on a **first memory chip having a specific address**—or group identifier—can be different from a PBA on a **second memory chip having a specific address**.

145. In this manner, Bruce teaches that a first and second PBA can be identified by the following access parameters, which are different:

	Access Parameters		
	Bus Identifier	FDE Identifier	Group Identifier
First unique physical location (PBA)	Bus 10/Bus-A	Bank0 SM	Specific address of first chip
Second unique physical location (PBA)	Bus 18/Bus-B	Bank3 SM	Specific address of second chip

146. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the “mapping table of claim 9, wherein said first and second PBA include different bus identifiers, different FDE identifiers and different group identifiers.”

4. Claim 11 Is Rendered Obvious by Sukegawa in View of Bruce

147. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 11.

- a. **[Claim 11]: “The mapping table of claim 9, wherein said first and second PBA respectively include different combinations of said bus identifiers, said FDE identifiers, and said group identifiers.”**

148. It is my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 9. *See supra* § X.A.2. It is also my opinion that Sukegawa and Bruce teach all of the elements of claim 11.

149. As explained above, Bruce teaches the use of bus identifiers, FDE identifiers, and group identifiers, which may be different for each memory chip. *See supra* §§ X.A.2.a, X.A.3. In this manner, Bruce teaches that a first and second PBA can be identified by the following parameters, which are different for each PBA:

	Access Parameters		
	Bus Identifier	FDE Identifier	Group Identifier
First unique physical location (PBA)	Bus 10/Bus-A	Bank0 SM	Specific address of first chip
Second unique physical location (PBA)	Bus 18/Bus-B	Bank3 SM	Specific address of second chip

See supra § X.A.3. As this table also illustrates, each PBA may be represented by different combinations of bus identifiers, FDE identifiers, and group identifiers.

150. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce's bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is

my opinion that Sukegawa in combination with Bruce renders obvious the “mapping table of claim 9, wherein said first and second PBA respectively include different combinations of said bus identifiers, said FDE identifiers, and said group identifiers.”

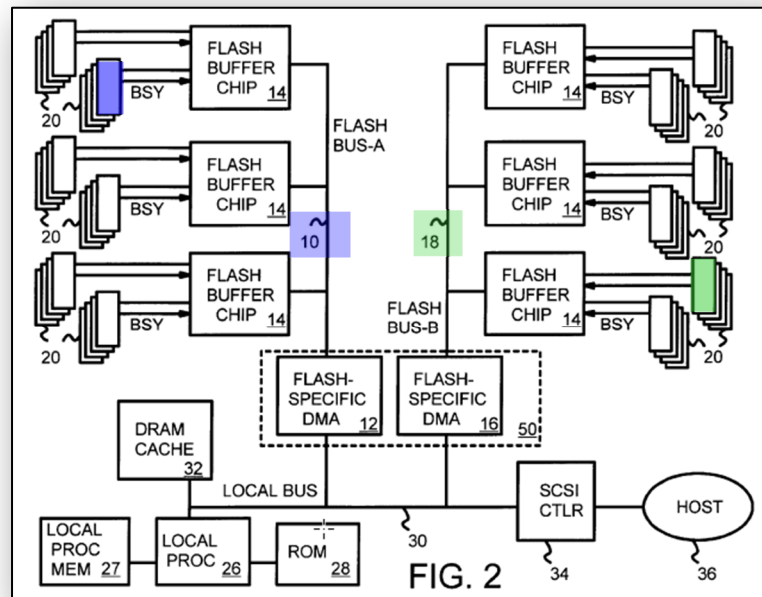
5. Claim 12 Is Rendered Obvious by Sukegawa in View of Bruce

151. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 12.

a. [Claim 12]: “The mapping table of claim 1, wherein said first and second PBA include different bus identifiers.”

152. It is my opinion that Sukegawa teaches all the elements of claim 1. *See supra* § **Error! Reference source not found.** It is also my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 12.

153. As explained above, Bruce teaches the use of bus identifiers. *See supra* §§ X.A.2.a.i, X.A.3. Bruce also teaches that these identifiers may be different for a first and second memory chip. For example, Bruce demonstrates that a PBA (i.e., a unique addressable physical memory location) for a **first memory chip** on bus 10 would have one bus identifier (“**bus 10**”), while a PBA for a **second memory chip** on bus 18 would have a different bus identifier (“**bus 18**”), as illustrated in annotated Figure 2, below:



154. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce's bus identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the “mapping table of claim 1, wherein said first and second PBA include different bus identifiers.”

6. Claim 13 Is Rendered Obvious by Sukegawa in View of Bruce

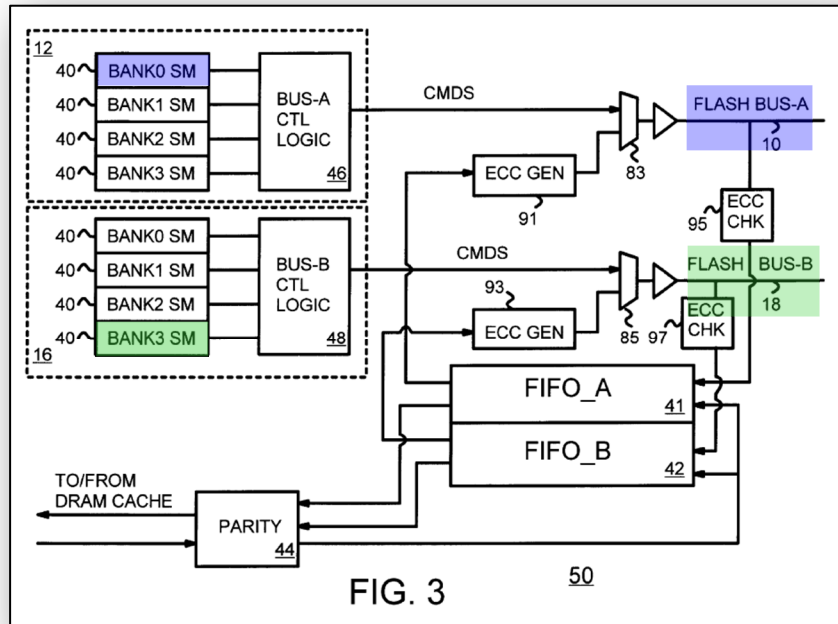
155. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 13.

- a. **[Claim 13]: “The mapping table of claim 1, wherein said first and second PBA include different FDE**

identifiers.”

156. It is my opinion that Sukegawa teaches all the elements of claim 1. *See supra* § **Error! Reference source not found.** It is also my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 13.

157. As explained above, Bruce teaches the use of FDE identifiers. *See supra* §§ X.A.2.a.ii, X.A.3. Bruce also teaches that these FDE identifiers may be different for a first and second memory chip. For example, Bruce teaches that each bank of memory devices is controlled by its own FDE engine (i.e., DMA state machine). *See* Ex-1006 at Abstract; 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-57; *see also supra* § X.A.2.a.ii. Thus, for example, a PBA on a **first memory chip** located on **bus 10** could have one FDE identifier (e.g., “**Bank0 SM**”), while a PBA for a **second memory chip** located on **bus 18** could have a different FDE identifier (e.g., “**Bank3 SM**”), as illustrated in annotated Figure 3, below:



158. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce’s bus identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the “mapping table of claim 1, wherein said first and second PBA include different FDE identifiers.”

7. Claim 14 Is Rendered Obvious by Sukegawa in View of Bruce

159. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 14.

a. [Claim 14]: “The mapping table of claim 1, wherein

said first and second PBA respectively include a first group identifier and a second group identifier.”

160. It is my opinion that Sukegawa teaches all the elements of claim 1. *See supra* § **Error! Reference source not found.** It is also my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 14.

161. As discussed above, Bruce teaches the use of group identifiers. *See supra* §§ X.A.2.a.iii, X.A.3. Bruce also teaches that each chip within a memory bank has a unique address that allows it to be separately accessed. Ex-1006 at 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”); 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); *see also supra* § X.A.2.a.iii. Accordingly, a PBA on a **first memory chip having a specific address**—or group identifier—can be different from a PBA on a **second memory chip having a specific address**.

162. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the “mapping table of claim 1, wherein said first and second PBA respectively include a first group

identifier and a second group identifier.”

8. Claim 15 Is Rendered Obvious by Sukegawa in View of Bruce

163. It is my opinion that Sukegawa in combination with Bruce renders obvious every element of claim 15.

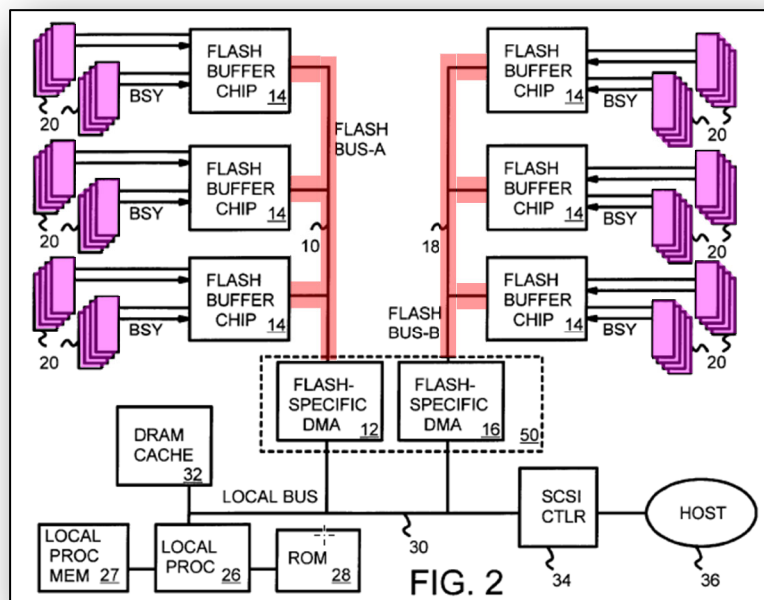
- a. [Claim 15a]: “The mapping table of claim 1: wherein said memory store includes a set of solid state memory devices that are coupled to a set of buses and that are controlled by a set of FDEs; and”**

164. It is my opinion that Sukegawa teaches all the elements of claim 1. *See supra* § **Error! Reference source not found.** It is also my opinion that the combination of Sukegawa and Bruce teach all of the elements of claim 15.

165. Bruce discloses a memory store that includes a set of solid state memory devices. For example, Bruce discloses multiple memory banks, each of which comprise a plurality of flash memory chips. Ex-1006 at 2:66-3:1; 5:31; 5:58-61. Each of these banks can be “separately accessed, allowing many flash operations to be performed in parallel.” *Id.* at 5:21-23. Accordingly, each of these memory banks comprise a set of solid state flash memory devices. *See supra* §§ X.A.2.a.ii, X.A.2.a.iii.

166. Bruce also discloses that the memory banks are coupled to a set of flash buses 10 and 18, which are “coupled to a flash-specific DMA controller” and have

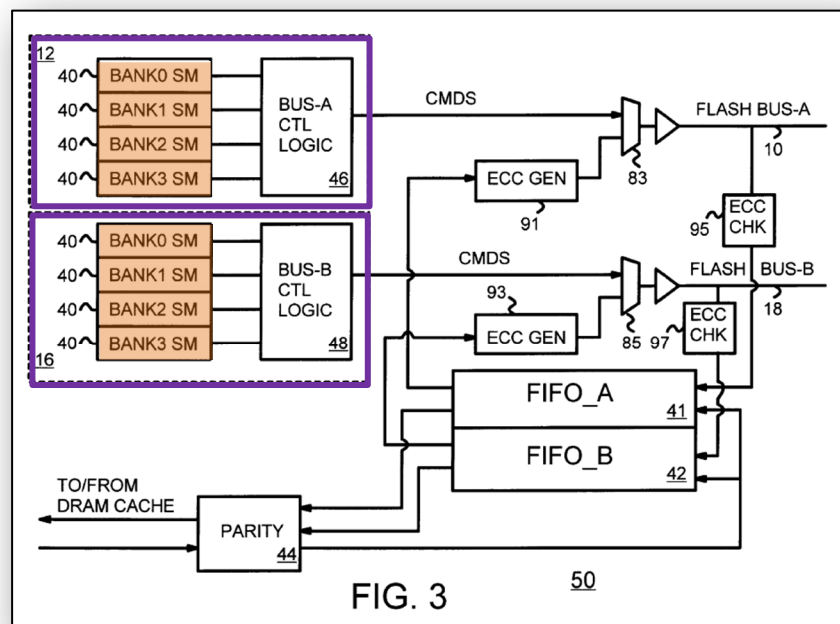
“shared lines that transmit the sequence of command bytes and address bytes from the flash-specific DMA controller.” *Id.* at 2:56-59; 3:62-64; 4:10-21; *see also supra* § X.A.2.a.ii. Annotated Figure 2, below, illustrates the set of **flash buses (bus 10/ Flash Bus-A and bus 18/ Flash Bus-B)** coupled to a plurality of **sets of solid state memory devices (flash memory banks)** through flash buffer chips:



167. Bruce further teaches that the sets of solid state memory devices are controlled by a set of FDEs. As explained above, Bruce teaches multiple DMA controllers containing four “state machines,” each of which control one of four banks of flash-memory chips. *Id.* at Abstract; *see also id.* at 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four

separate devices”); 6:53-55. Having four DMA state machines “allows four flash-memory chips to be accessed at once for each flash bus, for a total of eight flash accesses.” *Id.* at 6:55-57. Accordingly, these state machines are FDEs pursuant to Petitioners’ proposed as explained above. *See supra* § X.A.2.a.ii.

168. Annotated Figure 3, shown below, illustrates the four **state machines** (identified as “**Bank0 SM**,” “**Bank1 SM**,” “**Bank2 SM**,” and “**Bank3 SM**”) within each flash specific **DMA controller**, thereby providing a total of eight state machines:



169. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to

include Bruce's bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the mapping table of claim 1 "wherein said memory store includes a set of solid state memory devices that are coupled to a set of buses and that are controlled by a set of FDE."

- b. [Claim 15b]: "wherein said access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively controlled by different FDEs from said set of FDEs."**

170. It is my opinion that Bruce discloses this claim element. Bruce teaches that each individual bank of flash-memory chips "can be separately accessed, allowing many flash operations to be performed in parallel." Ex-1006 at 5:21-23; 6:55-57 (noting that "four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses"). Separate access is enabled by the fact that the flash memory chips connected to each flash buffer are "uniquely addressed." *Id.* at 7:62-67. Accordingly, to access the memory, the DMA controller sends specific device addresses for each chip to the flash buffer, which then decodes the address into chip selects. *Id.* at 8:46-49; 10:17-19 ("The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips."); *see also id.* at 2:40-42 (noting that, in the invention, a DMA controller inputs "commands and addresses . . . to a flash-memory chip"). As

also explained above, each FDE (e.g., state machine) controls an entire memory bank comprising a plurality of flash chips. *Supra* § X.A.2.a.ii.

171. Accordingly, by teaching that each chip in a memory bank has a unique address sent by the DMA controller, Bruce teaches the use of group identifiers—namely, “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE” pursuant to Petitioners’ proposed construction of group identifier. Moreover, Bruce discloses that each flash buffer chip may be connected to multiple banks of flash memory chips, each of which has its own state machine or FDE. Accordingly, the address (or group identifier) for the chips within each separate bank attached to flash buffer chip would identify “at least a portion of the set of memory devices” that are controlled by a different FDE from the set of FDEs.

As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Sukegawa with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Sukegawa. *See supra* ¶¶ 116-126. For at least these reasons, it is my opinion that Sukegawa in combination with Bruce renders obvious the mapping table of claim 1 “wherein said access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively

controlled by different FDEs from said set of FDEs.”

9. Claim 32 Is Anticipated By Sukegawa

172. As explained below, it is my opinion that Sukegawa discloses, and therefore anticipates, every element of claim 32.

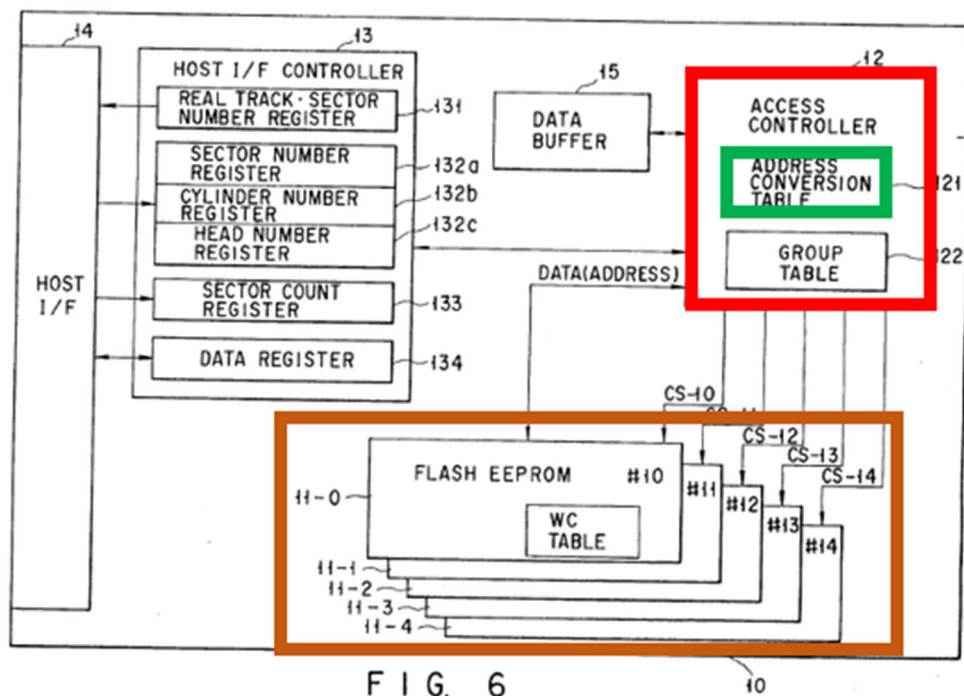
- a. **[Claim 32 pre]: “An electronic storage device that includes a memory system that uses a memory table for increasing the likelihood that an operational load imposed on the storage apparatus during the processing of an I/O transaction request will be optimally distributed across storage device resources, said I/O transaction request received by said storage device from a requesting host, said electronic storage device comprising:”**

173. To the extent the preamble is limiting, it is my opinion that Sukegawa discloses this claim element. For example, Sukegawa discloses an electronic storage device in the form of a semiconductor disk drive, which can be used “in place of a hard disk drive or a floppy disk drive” and utilizes a plurality of flash EEPROM chips “as data storage elements.” Ex-1005 at 7:6-13; *see also id.* at 7:26-32 (describing the structure and components of the semiconductor disk drive); claim 1 (reciting a “[s]emiconductor memory system having a plurality of flash EEPROM chips”).

174. Sukegawa also discloses that this electronic storage device comprises a memory system, which the ’740 patent describes as comprising a controller and

mapping table. Ex-1001 at 3:20-31. In particular, Sukegawa discloses an access controller, which “provides access control of the flash EEPROM chips.” Ex-1005 at 7:26-31, 7:46-51. Sukegawa further discloses that the access controller includes and is disposed to use a mapping table—or address conversion table—to select a flash EEPROM and “read[] and write[] data from and into the selected EEPROM.” *Id.*, 8:17-19, 7:35-40. The access controller is thus a memory system.

175. The electronic device including an **access controller**, which comprises the **address conversion table** and is coupled to the **memory store**, is shown below:



176. Sukegawa also teaches that the disclosed memory system uses a memory table. For example, Sukegawa discloses an “address conversion table” that converts logical addresses specified by a host into “real memory address[es]” in Flash EEPROM chips. Ex-1006 at 10:14-19; *see also id.* at 7:48-52. Sukegawa’s address conversion table is therefore a “memory table” as recited in the preamble. An illustration of Sukegawa’s memory table, which correlates logical addresses to real memory addresses, is shown in Figure 11 excerpted below:

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0	#10	0	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)
1	32	#14	1	(0, 1)
	39	↓		(14, 15)
	0	#10		(0, 1)
	7	↓		(14, 15)
2	8	#11	3	(0, 1)
	15	↓		(14, 15)
	32	#14		(0, 1)
	39	↓		(14, 15)

FIG. 11

177. Sukegawa also teaches that the disclosed memory table increases the likelihood that an operational load imposed on the storage apparatus will be optimally distributed across storage device resources by enabling flash EEPROMS

to be “accessed parallelly.” *Id.* at 12:35-40;⁶ *see also id.* at 4:51-57 (explaining that the “primary object of the present invention is to provide a semiconductor memory system which is designed to permit parallel access to a plurality of flash EEPROMs”); 5:8-13 (discussing means for assigning sequential number sets “so as to permit the memory access means to parallelly access the plurality of flash EEPROM chips”); 10:10-13 (teaching that “transfer to flash EEPROM chips 11-0 through 11-2 and the writing of data into flash EEPROM chips 11-0 through 11-2 are practically performed simultaneously”); *id.* at 10:20-21 (stating that the disclosed systems enable “flash EEPROMS [to be] accessed parallelly”). Sukegawa also discloses “contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one parameter,” *infra* §§ X.A.9.b, X.A.9.c, X.A.9.e, which the district

⁶ The ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.,* Ex-1001 at 2:14-21 (“***These optimizing memory operations include*** increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally distributed across different storage device resources, ***such as by interleaving or parallel memory operations***”) (emphasis added).

court has identified as examples of optimized memory operations. Ex-1008 at 30.

178. In addition, Sukegawa explains that these optimized memory operations are also performed in response to receiving an I/O transaction request initiated by a host, explaining that the controller accesses memory for read and write transactions “in response to an [sic] disk access request supplied from a host CPU.” *Id.* at 7:28-32; *see also id.* at 11:3-6 (explaining that “[w]hen a write access request for a specified address (logical address) occurs, the block corresponding to the specified address is accessed on the basis of the address conversion table 121”).

179. For at least these reasons, it is my opinion that Sukegawa discloses “[a]n electronic storage device that includes a memory system that uses a memory table for increasing the likelihood that an operational load imposed on the storage apparatus during the processing of an I/O transaction request will be optimally distributed across storage device resources, said I/O transaction request received by said storage device from a requesting host.”

b. [Claim 32a]: “a mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said first and second LBA sets including respective sets of consecutive LBAs;”

180. It is my opinion that Sukegawa discloses this claim element. In particular, Sukegawa discloses logical “sectors” that are part of “a logical address designated by a host system.” Ex-1005 at 4:67-5:4; *see also id.* at 9:14-19

(explaining that “logical addresses . . . specified by the host CPU” comprise track numbers and sector numbers); 4:30-35 (same). Accordingly, each of Sukegawa’s logical sectors is an LBA—or “an address that is part of a logical addressing system used by a host”—pursuant to Petitioners’ proposed construction of LBA.

181. Sukegawa also illustrates the use of LBA sets, each of which represent a set of consecutive LBAs. For example, Sukegawa discloses sets of LBAs (logical sectors) in which a first LBA set comprises eight consecutive LBAs from 0-7, a second LBA set comprises eight consecutive LBAs from 8-15, and so on, all the way up a set that comprises eight consecutive LBAs from 32-39. *See, e.g., id.* at 8:66-9:8; 8:36-39 (noting that a write unit from the host CPU “is allocated consecutive sector numbers in the same track”); Figure 11 (illustrating sets of consecutive LBAs or sectors); claim 9 (reciting a semiconductor memory system for storing “address information representing how consecutive logic addresses corresponding to one track are assigned to physical addresses”). These sets thus comprise “a first LBA set” and a “second LBA set” that each “include respective sets of consecutive LBAs.”

182. Sukegawa further discloses “real memory addresses,” which correspond to physical memory locations represented by chip numbers, block numbers, and page numbers and are associated with the LBAs. Ex-1005 at 9:14-21;

4:30-35 (explaining that the “real memory address” is the address “by which the flash EEPROMS in the semiconductor disk drive are selectively accessed”); Figure 11 (illustrating the chip numbers, block numbers, and page numbers for the “real memory addresses”); claim 9 (reciting that that logical addresses “are assigned to physical address of the [] flash EEPROM chips”). Accordingly, each “real memory address” is a PBA—or a “unique addressable physical memory location”—within a memory store pursuant to Petitioners’ proposed construction of PBA. In this manner, Sukegawa teaches a “first PBA” and a “second PBA” as recited in this claim element.

183. Sukegawa also teaches that the disclosed “address conversion table” maps a first LBA set to a first PBA, and second LBA set to a second PBA. For example, Sukegawa explains that the disclosed tables provide an association or “correspondence between the logical addresses . . . specified by the host CPU and the real memory addresses.” *Id.* at 9:14-19; *see also id.* at 10:14-19 (explaining that the address conversion table enables “converting the logical addresses from the host CPU into the real memory address”); 5:46-51 (explaining that the stored address conversion information is “used for converting the logical address from the host system into the real memory address”); claim 1 (reciting means “for converting the logical address designated by a host system, into the real memory address for accessing at least one of said plurality of flash EEPROM chips”). As a result,

Sukegawa's address conversion table "maps a first LBA set to a first PBA, and a second LBA set to a second PBA" as recited in this claim element.

184. Annotated Figure 11, shown below, illustrates that Sukegawa's mapping table maps a **first LBA set** to a **first PBA**, and a **second LBA set** to a **second PBA**, in which each of the LBA sets include sets of **consecutive LBAs**:

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0	#10	0	(0, 1)
	7			(14, 15)
	8	#11		(0, 1)
	15			(14, 15)
1	32	#14		(0, 1)
	39			(14, 15)
	0	#10	1	(0, 1)
	7			(14, 15)
2	8	#11		(0, 1)
	15			(14, 15)
	32	#14		(0, 1)
	39			(14, 15)
3	0	#10	3	(0, 1)
	7			(14, 15)
	8	#11		(0, 1)
	15			(14, 15)

FIG. 11

185. For at least these reasons, it is my opinion that Sukegawa discloses a

“mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said first and second LBA sets including respective sets of consecutive LBAs.”

- c. **[Claim 32b]: “a memory store that includes a set of flash devices, said set of flash devices includes a first physical memory location and a second physical memory location that are addressable by said first and second PBAs respectively;”**

186. It is my opinion that Sukegawa discloses this claim element. As explained above, Sukegawa teaches that the semiconductor disk drive includes a memory store comprising a plurality of flash EEPROM chips “as data storage elements.” Ex-1005 at 7:6-13; *see also id.* at 7:26-32 (describing the structure and components of the semiconductor disk drive); claim 1 (reciting a “[s]emiconductor memory system having a plurality of flash EEPROM chips”); *supra* § X.A.9.a. More specifically, Sukegawa discloses a memory store comprising five flash EEPROM chips, designated as EEPROM chips 11-0 to 11-4. *Id.* at 7:11-16.⁷

⁷ EEPROM chip 11-0 is also referred to as “chip #10,” EEPROM chip 11-1 is also referred to as “chip #11,” EEPROM chip 11-2 is also referred to as “chip #12,” EEPROM chip 11-3 is also referred to as “chip #13,” and EEPROM chip 11-4 is also referred to as “chip #14.” Ex-1005 at 9:14-21.

Sukegawa's plurality of flash EEPROM chips thus comprise "a memory store that includes a set of flash devices" as recited in this claim element.

187. Sukegawa also teaches that these flash devices comprise physical memory locations addressable by the PBAs. For example, Sukegawa explains that each PBA enables "accessing the flash EEPROMs 11-0 through 11-4." *Id.* at 9:14-21; Figure 11 (illustrating the chip numbers for each real memory address stored in the mapping table). Sukegawa also teaches that each PBA addresses a specific location within the memory store, as represented by a block number and page number. As a result, Sukegawa discloses a memory store with a "first physical memory location" and a "second physical memory location" that are "addressable by said first and second PBAs."

188. Annotated Figures 6 and 11, below, illustrate the **memory store**, in which a first physical location is represented by **a first PBA**, and a second physical location is represented by a **second PBA**:

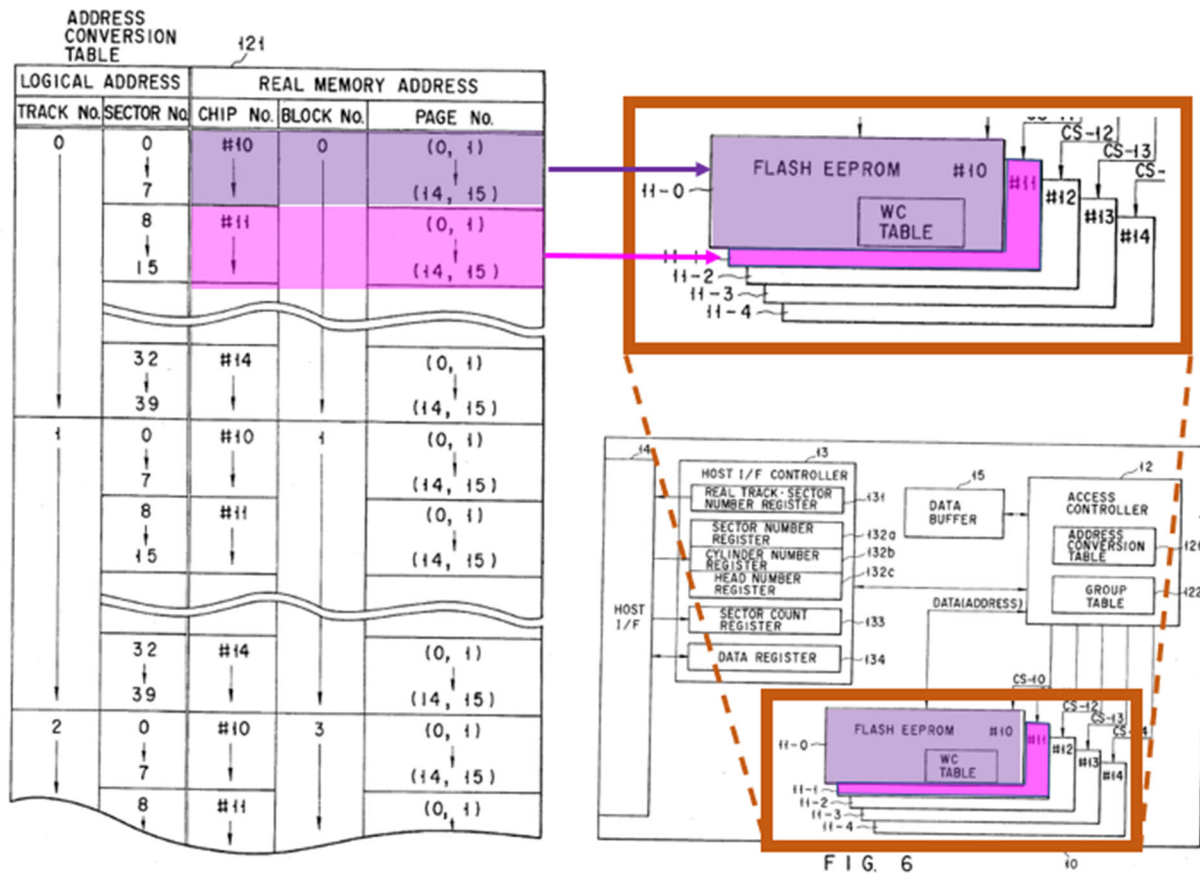


FIG. 11

FIG. 6

189. For at least these reasons, it is my opinion that Sukegawa teaches “a memory store that includes a set of flash devices, said set of flash devices includes a first physical memory location and a second physical memory location that are addressable by said first and second PBAs respectively.”

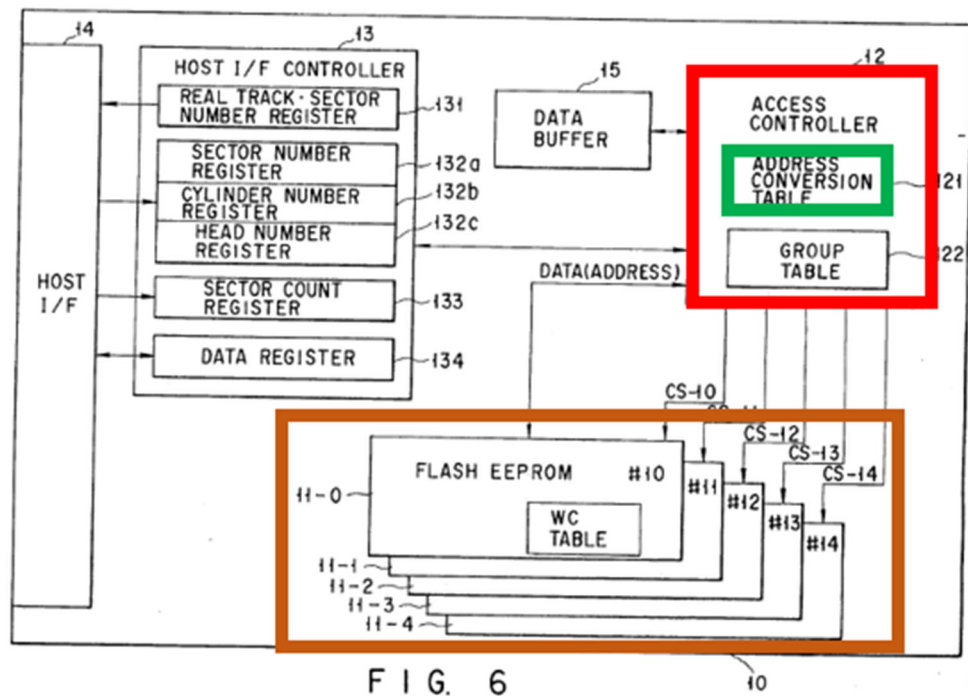
d. [Claim 32c]: “a memory system coupled to said memory store and disposed to use said mapping table during a memory operation;”

190. It is my opinion that Sukegawa discloses this claim element. For example, Sukegawa discloses a memory system comprising an access controller that

“provides access control of the flash EEPROM chips.” Ex-1005 at 7:26-32.⁸ In particular, the access controller is provided with the address conversion table—or mapping table—that supplies “the correspondence between the logical addresses . . . from the host CPU and the real addresses for accessing the flash EEPROM chips.” *Id.* at 7:46-51. As a result, the access controller utilizes the mapping information to select a flash EEPROM and “reads and writes data from and into the selected flash EEPROM.” *Id.* at 8:17-19; 7:35-40 (explaining that the access controller “supplies an address” for a write operation “indicating an access position to the flash EEPROM chips”). Thus, Sukegawa’s access controller is a “memory system” that is disposed to “use said mapping table during a memory operation.”

191. Annotated Figure 6, below, illustrates the **access controller**, comprising the **address conversion table**, which is coupled to the **memory store**:

⁸ The '740 patent describes the “memory system” as comprising a DMA controller and mapping table. *See* Ex-1001 at 3:20-31; Figure 1 (illustrating Memory System 4).



192. For at least these reasons, it is my opinion that Sukegawa teaches “a memory system coupled to said memory store and disposed to use said mapping table during a memory operation.”

- e. **[Claim 32d]: “wherein said first PBA includes a first set of access parameters and said second PBA includes a second set of access parameters, and at least one difference between said first and second PBAs;”**

193. It is my opinion that Sukegawa discloses this claim element. For example, as explained above, Sukegawa teaches that the PBAs (real memory locations) comprise fields for identifying aspects of a memory location, such as a chip number, block number, and page number. *See* Ex-1005 at 9:14-21; Figure 11. As a result, each PBA is “disposed to represent a set of access parameters”

representing the physical memory location, such as identifiers for a specific chip number and block number. *See id.* at 9:14-21; Figure 11.

194. Sukegawa further discloses “at least one difference between the first and second PBAs.” For example, as annotated Figure 11 demonstrates, **first PBA** designates chip #10, while the **second PBA** designates chip #11:

Different chip numbers = difference
between first and second PBAs

ADDRESS
CONVERSION
TABLE 121

LOGICAL ADDRESS		REAL MEMORY ADDRESS				
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.		
0	0	#10	0	(0, 1)		
	↓	↓		↓		
	7	↓		(14, 15)		
	8	#11		(0, 1)		
	↓	↓		↓		
	15	↓		(14, 15)		
	~					
	32	#14		(0, 1)		
	↓	↓		↓		
	39	↓		(14, 15)		
	1	0		#10	1	(0, 1)
	↓	↓		↓		
7	↓	(14, 15)				
8	#11	(0, 1)				
↓	↓	↓				
15	↓	(14, 15)				
~						
32	#14	(0, 1)				
↓	↓	↓				
39	↓	(14, 15)				
2	0	#10	3	(0, 1)		
↓	↓	↓				
7	↓	(14, 15)				
	8	#11		(0, 1)		

FIG. 11

195. For at least these reasons, it is my opinion that Sukegawa discloses a mapping table “wherein said first PBA includes a first set of access parameters and said second PBA includes a second set of access parameters, and at least one difference between said first and second PBAs.”

- f. **[Claim 32e]: “and wherein said mapping table increases the likelihood that that the operational load imposed on the storage apparatus during the processing of the I/O transaction request will be optimally distributed across storage device resources.”**

196. It is my opinion that Sukegawa discloses this claim element. Sukegawa teaches that the disclosed mapping table increases the likelihood that an operational load will be optimally distributed across storage device resources by “enabling parallel write operations” on different sectors in response to a request from a host.⁹ Ex-1005 at 8:39-43; *see also id.* at 10:20-21 (explaining that “when the host CPU

⁹ The '740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.,* Ex-1001 at 2:14-21 (“***These optimizing memory operations include*** increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally distributed across different storage device resources, ***such as by interleaving or parallel memory operations***”) (emphasis added).

specifies consecutive numbers in the same track, flash EEPROMs are accessed parallelly”); 12:36-41 (indicating that “EEPROMs can be accessed parallelly when consecutive sector numbers in the same track are specified,” allowing existing disk accessing techniques to be used effectively); 4:51-57 (explaining that the invention “permit[s] parallel access to a plurality of flash EEPROMs”); 5:8-13 (discussing means for assigning sequential number sets “so as to permit the memory access means to parallelly access the plurality of flash EEPROM chips”). Sukegawa also discloses “contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one parameter,” *supra* §§ X.A.9.b, X.A.9.c, X.A.9.e, which the district court has identified as additional examples of optimized memory operations. Ex-1008 at 30.

197. In addition, Sukegawa explains that these optimized memory operations are also performed in response to receiving an I/O transaction request initiated by a host, explaining that the controller accesses memory for read and write transactions “in response to an [sic] disk access request supplied from a host CPU.” *Id.* at 7:28-32; *see also id.* at 11:3-6 (explaining that “[w]hen a write access request for a specified address (logical address) occurs, the block corresponding to the specified address is accessed on the basis of the address conversion table 121”).

198. For at least these reasons, it is my opinion that Sukegawa discloses a

mapping table that “increases the likelihood that that the operational load imposed on the storage apparatus during the processing of the I/O transaction request will be optimally distributed across storage device resources.”

10. Claim 34 Is Anticipated by Sukegawa

199. As explained below, it is my opinion that Sukegawa discloses, and therefore anticipates, every element of claim 34.

- a. **[Claim 34a]: “The electronic storage device of claim 32, wherein the processing of the I/O transaction request includes a first memory operation performed using said first PBA and a second memory operation performed using said second PBA;”**

200. It is my opinion that Sukegawa teaches all of the elements of claim 32. *See supra* § X.A.9. It is also my opinion that Sukegawa teaches all of the elements of claim 34.

201. Sukegawa discloses performing a first memory operation using a PBA and a second memory operation using a second PBA. For example, Sukegawa teaches that a host CPU can specify a memory write transaction encompassing 23 logical sectors of data, beginning with a logical address of track 0 and sector 0. Ex-1005 at 9:35-38. In performing this transaction, the controller begins writing data to page 0, of block 0, in flash EEPROM chip #10 (or EEPROM 11-0). *Id.* at 9:51-54. This location corresponds to the **first PBA**, as illustrated in annotated Figure 11,

below.

202. Once this location is full, Sukegawa teaches that the controller continues by writing to page 0, of block 0, of flash EEPROM chip #11 (or EEPROM 11-1). *Id.* at 9:56-65. This location corresponds to the **second PBA**, as illustrated in annotated Figure 11, below. In this manner, Sukegawa discloses that “processing the I/O transaction includes a first memory operation” using the first PBA, and a “second memory operation” using the second PBA.

ADDRESS
CONVERSION
TABLE

121

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0	#10	0	(0, 1)
	↓	↓		↓
	7			(14, 15)
	8	#11		(0, 1)
	↓	↓		↓
	15			(14, 15)
	32	#14		(0, 1)
	↓	↓		↓
	39			(14, 15)
1	0	#10	1	(0, 1)
	↓	↓		↓
	7			(14, 15)
	8	#11		(0, 1)
	↓	↓		↓
	15			(14, 15)
	32	#14		(0, 1)
	↓	↓		↓
	39			(14, 15)
2	0	#10	3	(0, 1)
	↓	↓		↓
	7			(14, 15)
	8	#11		(0, 1)
	↓	↓		↓

First write operation at
Chip 10, Block 0, Pages 0-15

Second write operation at
Chip 11, Block 0, Pages 0-15

FIG. 11

203. For at least these reasons, it is my opinion that Sukegawa discloses the “electronic storage device of claim 32, wherein the processing of the I/O transaction request includes a first memory operation performed using said first PBA and a second memory operation performed using said second PBA.”

- b. **[Claim 34b]: “said first and second PBAs respectively associated with a first set of access parameters and a second set of access parameters, and said set of first and second set of access parameters differing by at least one access parameter.”**

204. It is my opinion that Sukegawa discloses this claim element. As explained above, Sukegawa teaches that PBAs (or real memory locations) can be represented by various fields, including fields for a particular chip, block, and page. Ex-1005 at 9:14-21; Figure 11; *supra* § X.A.9.e. As a result, each PBA is disposed to represent a specific set of access parameters for representing the physical memory location, such as a specific chip, specific block within the chip, and the specific pages within the block. Ex-1005 at 9:14-21; Figure 11.

205. Sukegawa further discloses at least one difference between these access parameters. For example, as annotated Figure 11 demonstrates, the access parameters for the **first PBA** identify chip #10, while the access parameters for the **second PBA** identify chip #11:

ADDRESS
CONVERSION
TABLE

121

Difference between
access parameters

The diagram shows an address conversion table with the following structure:

LOGICAL ADDRESS		REAL MEMORY ADDRESS		
TRACK No.	SECTOR No.	CHIP No.	BLOCK No.	PAGE No.
0	0	#10	0	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)
1	0	#10	1	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)
2	0	#10	3	(0, 1)
	7	↓		(14, 15)
	8	#11		(0, 1)
	15	↓		(14, 15)

The table is divided into three main sections for TRACK No. 0, 1, and 2. Each section contains four rows for SECTOR No. 0, 7, 8, and 15. The CHIP No. is #10 for sectors 0 and 7, and #11 for sectors 8 and 15. The BLOCK No. is 0 for TRACK 0, 1 for TRACK 1, and 3 for TRACK 2. The PAGE No. is (0, 1) for sectors 0 and 8, and (14, 15) for sectors 7 and 15. A red arrow points from the text "Difference between access parameters" to the difference between the CHIP No. and the BLOCK No. for the first row of each section.

FIG. 11

206. For at least these reasons, it is my opinion that Sukegawa discloses “said first and second PBAs respectively associated with a first set of access parameters and a second set of access parameters, and said set of first and second set of access parameters differing by at least one access parameter.”

B. Each of the Challenged Claims Is Anticipated or Rendered Obvious by Bennett

207. It is my opinion that Bennett discloses, and therefore anticipates, every element of claim 1. It is also my opinion that Bennett in combination with Bruce renders obvious every element of claims 9-15, 32, and 34.

1. Claim 1 Is Anticipated by Bennett

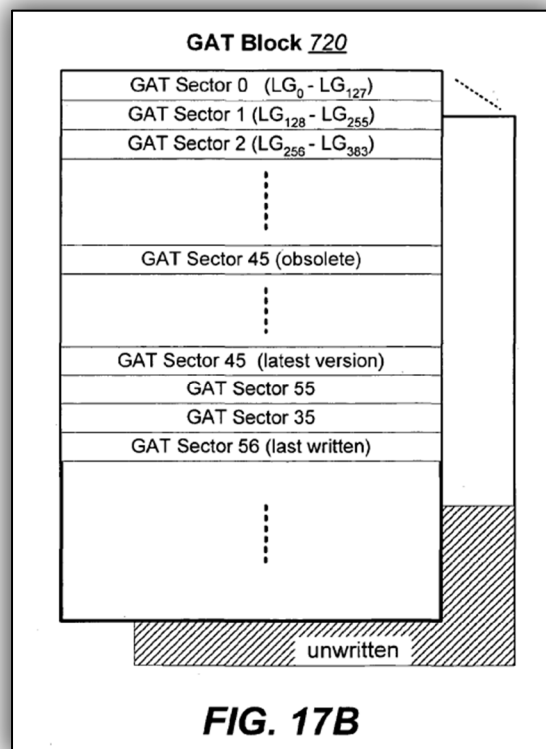
208. As explained below, it is my opinion that Bennett discloses, and therefore anticipates, every element of claim 1.

a. [Claim 1 Pre]: “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:”

209. To the extent the preamble is limiting, it is my opinion that Bennett discloses this claim element. Bennett discloses tables for “[m]apping between logical groups and physical groups (metablocks).” Ex-1007 at 23:61-64; 7:9-11; 9:55-58. For example, Bennett discloses a Group Address Table—or “GAT”—that “is a list of [physical] metablock addresses for all logical groups of host data in the

memory system.” *Id.* at 24:19-21. The GAT “contains one entry for each logical group, ordered sequentially according to logical address.” *Id.* at 24:21-22; *see also id.* at 25:1-3 (“GAT sectors within a GAT block each contain logical to physical mapping information for 128 contiguous logical groups.”); 9:55-58 (explaining that the GAT tracks “mapping between logical groups of sectors and their corresponding metablocks”).

210. Figure 17A, shown below, illustrates the GAT with entries for each logical group:



211. Bennett further discloses that the mapping table optimizes memory

operations by enabling parallel memory operations. For example, Bennett discloses that “[i]n order to maximize programming speed and erase speed, **parallelism** is exploited as much as possible by arranging for multiple pages of information . . . to be **programmed in parallel**, and for multiple MEUs [minimum erasable units] to be **erased in parallel**.”¹⁰ *Id.* at 8:26-30; *see also id.* at 2:57-59 (“In order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in **parallel**.”) (all emphasis added). Thus, the mapping table disclosed in Bennett provides a more “efficient use of system resources” by allowing “multiple logical groups to be updated concurrently,” which “increases efficiency and reduces overhead.” *Id.* at 4:50-52.

212. The district court’s claim construction order also states that examples of optimization “include[e] contiguous LBA sets, adjacent LBA sets, [and] adjacent

¹⁰ The ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.,* Ex-1001 at 2:14-21 (“**These optimizing memory operations include** increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally distributed across different storage device resources, **such as by interleaving or parallel memory operations**”) (emphasis added).

PBA's differing by at least one access parameter." Ex-1008 at 30. As explained below, Bennett teaches each of these examples. *Infra* §§ X.B.1.b, X.B.1.c. Thus, pursuant to the district court's claim construction, these aspects of Bennett further demonstrate that Bennett's mapping table enables optimized memory operation.

213. Bennett further discloses that the mapping table optimizes memory operations in response to processing of an I/O transaction with a host. For example, Bennett discloses that the host "accesses the memory" and reads or writes "to the memory system in unit[s] of logical clusters, each consisting of one or more logical sectors." Ex-1007 at 6:53-59; *see also id.* at 6:61-65 (explaining that the host "issues a command to the memory system 20 to read or write a segment containing a string of logical sectors of data with contiguous addresses"). As a result, the system disclosed by Bennett "maps the logical address from the host to a physical memory location." *Id.* at 9:20-22.

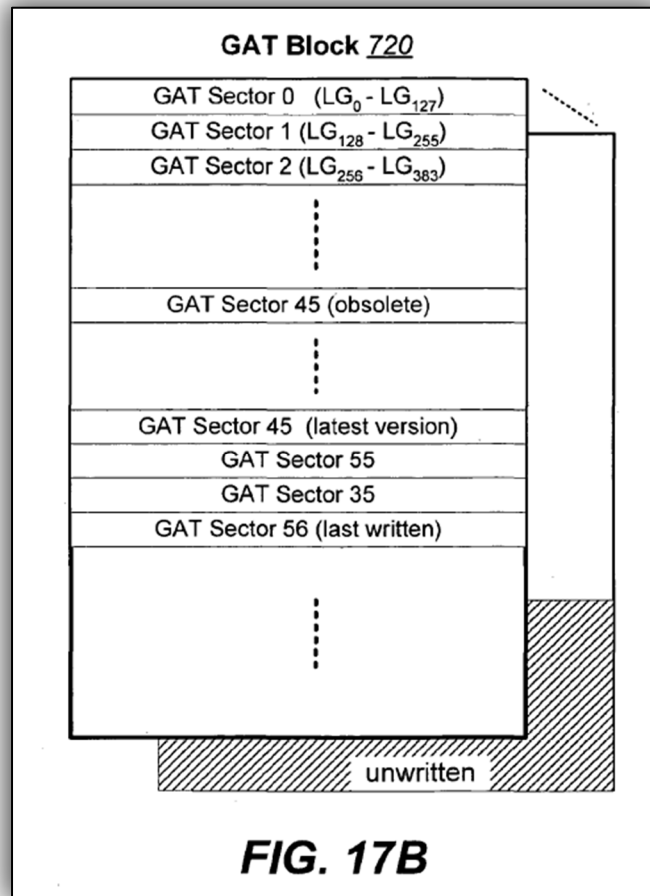
214. For at least these reasons, it is my opinion that Bennett discloses "[a] mapping table for optimizing memory operations performed by an electronic storage device in response to receiving an I/O transaction request initiated by a host."

- b. [Claim 1a]: "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;”**

215. It is my opinion that Bennett discloses this claim element. As Bennett explains, the GAT comprises a “set of sectors (referred to as GAT sectors) with entries defining metablock addresses for every logical group in the memory system.” Ex-1007 at 24:24-27. Moreover, each GAT sector “contain[s] logical to physical mapping information for 128 contiguous logical groups.” *Id.* at 25:1-3; *see also id.* at 24:21-22 (explaining that the GAT “contains one entry for each logical group, ordered sequentially according to logical address”); 24:66-25:1 (“[A] GAT block contains entries for logically contiguous set of groups in a region of logical address space.”).

216. Figure 17A, reproduced below, illustrates sets of consecutive logical groups within each GAT sector. For example, GAT Sector 0 contains consecutive logical groups 0-127, GAT Sector 1 contains consecutive logical groups 128-255, etc.:

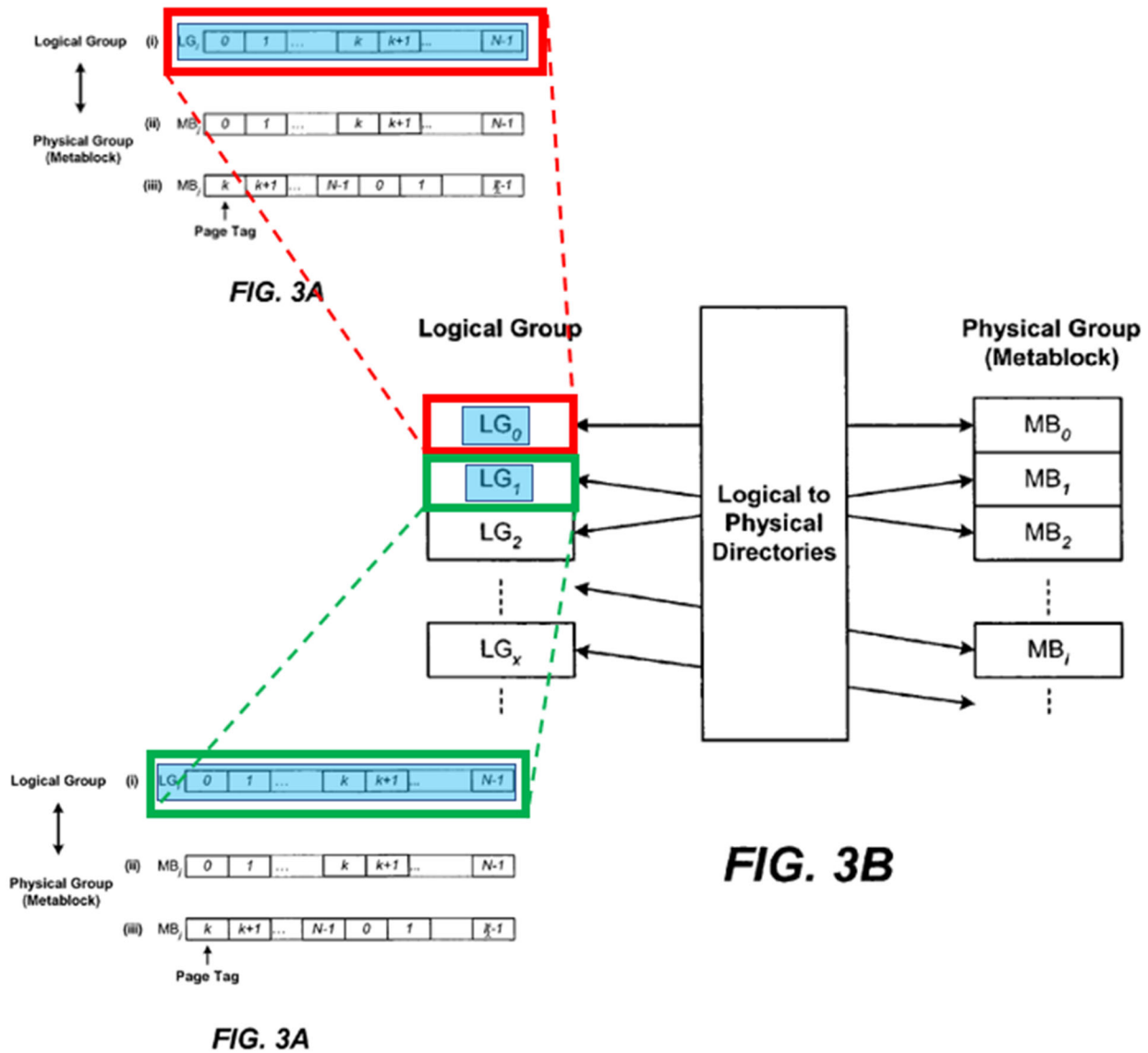


217. Bennett also teaches that each logical group comprises a set of logical sectors, and that the “host system addresses data in units of logical sectors where, for example, each sector may contain 512 bytes of data.”¹¹ *Id.* at 6:54-57; *see also id.* at 6:66-7:2 (explaining that the controller “manage[s] the storage and retrieval of

¹¹ The '740 patent confirms that an LBA typically has a “size equal to 512 bytes.” Ex-1001 at 7:22-25.

the data of host logical sectors”). As Bennett explains, the logical sectors within the logical group “are in **contiguous** logical order 0, 1, . . . , N-1.” *Id.* at 7:13-15; 6:61-65 (noting that the host issues a command “to read or write a segment containing a string of logical sectors of data with **contiguous addresses**”); 11:6-9 (explaining that “each host write is a segment of one or more sector with **contiguous** logical addresses”) (all emphasis added). Accordingly, each of Bennett’s logical sectors is an LBA—or “an address that is part of a logical addressing system used by a host”—pursuant to Petitioners’ proposed construction. Further, by disclosing multiple logical groups that each contain contiguous LBAs (sectors), Bennett teaches a plurality of logical sets (logical groups) comprising a set of consecutive LBAs.

218. Annotated Figures 3A and 3B, shown below, illustrates a **first logical field** (LG₀) comprising a set of **consecutive LBAs** (logical sectors) from 0 to N-1, and a **second logical field** (LG₁) comprising a set of **consecutive LBAs** (logical sectors) from 0 to N-1:



219. For at least these reasons, it is my opinion that Bennett discloses “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.”

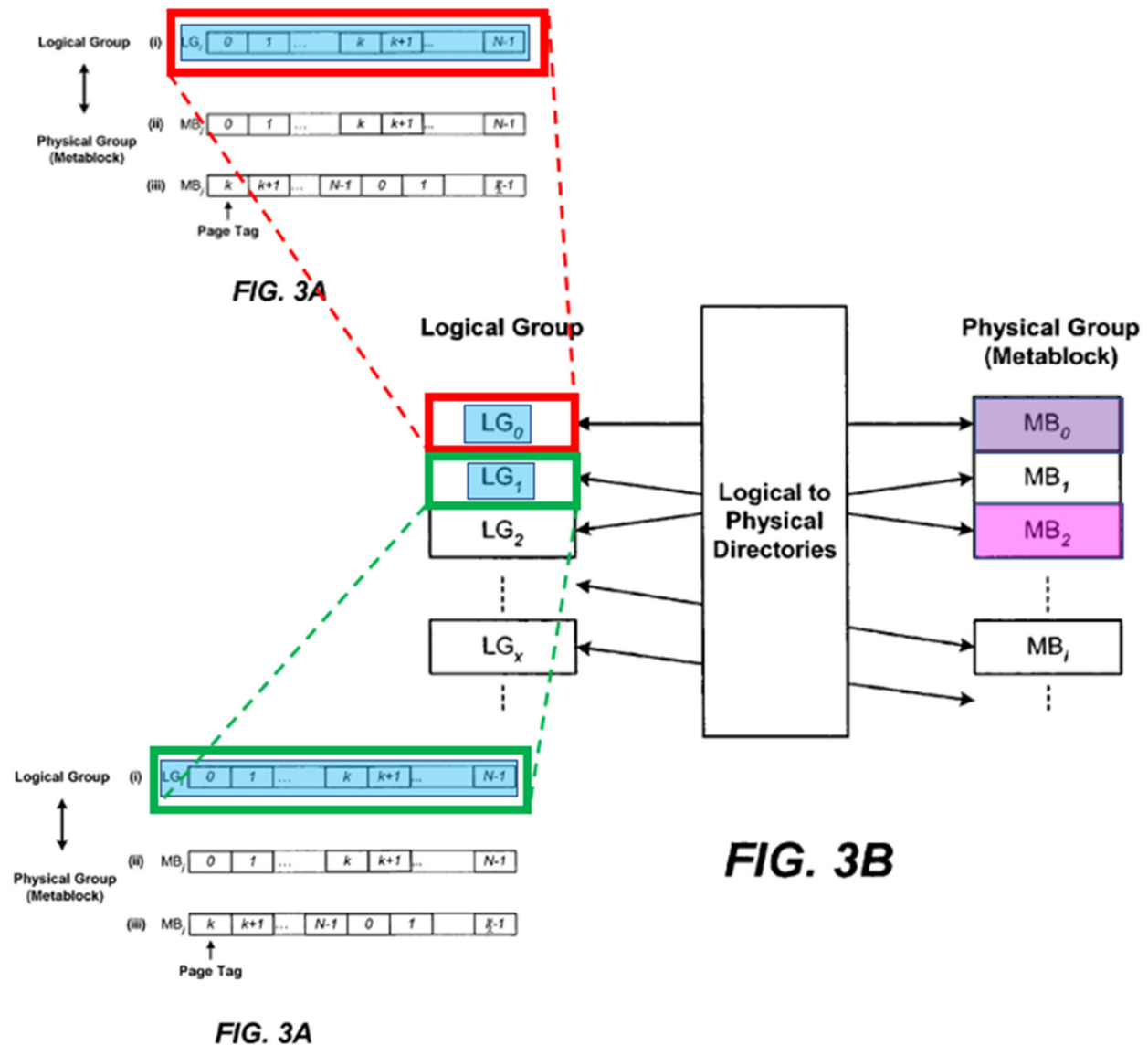
- c. **[Claim 1b]: “a set of PBA fields, including a first PBA field and a second PBA field, said set of PBA fields respectively disposed for representing a set of PBAs, including a first PBA disposed for representing a first set of access parameters and a second PBA disposed for representing a second 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”**

220. It is my opinion that Bennett discloses this claim element. For example, Bennett discloses metablocks that correspond to physical memory locations, explaining that “[t]he physical address space of the flash memory is treated as a set of metablocks.” Ex-1007 at 8:17-19; *see also id.* at 4:13-15 (noting that “[e]ach physical group (metablock) is erasable as a unit and can be used to store a logical group of data”); *id.* at 23:64-66 (stating that “[a]n address table is maintained in flash memory, containing a metablock address for every logical group in the memory system.”). Accordingly, Bennett’s metablocks are “unique addressable physical memory locations” pursuant to Petitioners’ proposed construction of “PBA.”

221. Bennett’s table also associates the first and second LBA sets with first and second PBAs (metablocks). As Bennett explains, “[e]ach logical group is mapped to a unique metablock,” and the “mapping information is maintained in a

set of logical to physical directories.” Ex-1007 at 7:38-44; *see also id.* at 23:61-64 (explaining that the GAT maps “between logical groups and physical groups (metablocks).”); *id.* at 9:56-58 (“The GAT 120 keeps track of the mapping between logical groups of sectors and their corresponding metablocks.”); *id.* at 23:64-66 (“An address table is maintained in flash memory, containing a metablock address for every logical group in the memory system.”).

222. Annotated Figures 3A and 3B, shown below, illustrate the **first logical field** (LG₀,) associated with the **first PBA** (MB₀), and the **second logical field** (LG₁) is associated with the **second PBA** (MB₂):



223. Bennett also discloses that the PBAs are disposed to represent access parameters. For example, Bennett explains that the GAT comprises data fields for each sector, which include the “metablock number, the page tag . . . and a flag indicating whether the metablock has been relinked.” *Id.* at 24:39-42. In this manner, Bennett teaches that each PBA (metablock) is “disposed to represent a set

of access parameters” representing the physical memory location, such as identifiers for a metablock, page tag, and relinking flag. *See id.* at 24:39-42; Figures 3A, 3B, and 17A.

224. Combined Figures 3A and 17A, shown below, illustrate the **PBA fields** for the **first PBA** (which corresponds to the **first logical field**) and the **second PBA** (which corresponds to the **second logical field**):

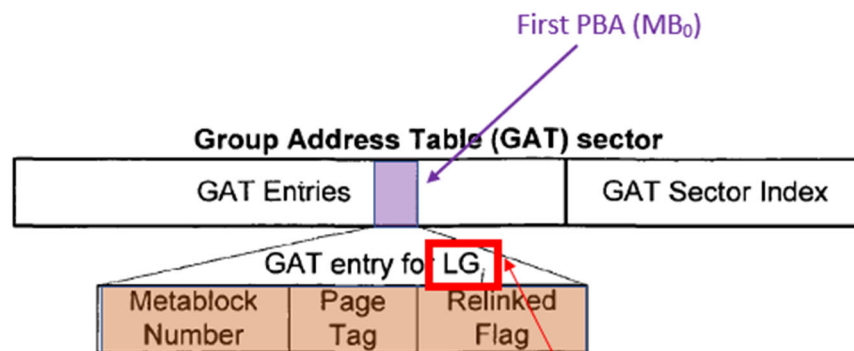


FIG. 17A

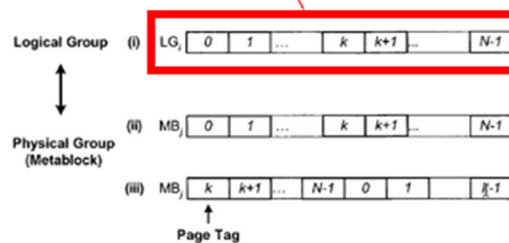


FIG. 3A

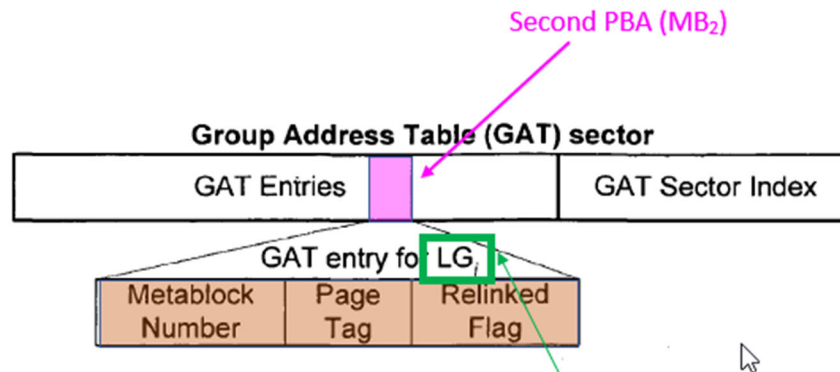


FIG. 17A



FIG. 3A

225. For at least these reasons, it is my opinion that Bennett discloses “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.”

d. [Claim 1c]: “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.”

226. It is my opinion that Bennett discloses this claim element. Bennett discloses that “[i]n order to maximize programming speed and erase speed, *parallelism* is exploited as much as possible by arranging for multiple pages of information . . . to be *programmed in parallel*, and for multiple MEUs [minimum erasable units] to be erased in parallel.” Ex-1007 at 8:26-30;¹² *see also id.* at 2:57-59 (“In order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in *parallel*.”) (all emphasis added). Thus, the mapping table disclosed in Bennett provides a more “efficient use of system resources” by allowing “multiple logical groups to be

¹² The ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *See, e.g.,* Ex-1001 at 2:14-21 (“*These optimizing memory operations include* increasing the likelihood that . . . the operational load imposed on the storage devices by these memory operations will be optimally distributed across different storage device resources, *such as by interleaving or parallel memory operations*”) (emphasis added).

updated concurrently,” which “increases efficiency and reduces overhead[.]” *Id.* at 4:50-52.

227. Further, as noted above, the district court’s claim construction order states that examples of optimization “include[e] contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one access parameter.” Ex-1008 at 30. Bennett teaches each of these examples. *Supra* § X.B.1.b, X.B.1.c. Accordingly, these aspects of Bennett further demonstrate that Bennett’s disclosed mapping table enables optimized memory operation pursuant to the district court’s claim construction.

228. Bennett further discloses that the mapping table optimizes memory operations in response to processing of an I/O transaction with a host. For example, Bennett discloses that the host “accesses the memory” and reads or writes “to the memory system in unit[s] of logical clusters, each consisting of one or more logical sectors.” Ex-1007 at 6:54-59; *see also id.* at 6:61-65 (explaining that the host “issues a command to the memory system 20 to read or write a segment containing a string of logical sectors of data with contiguous addresses”). As a result, the system disclosed by Bennett “maps the logical address from the host to a physical memory location.” *Id.* at 9: 20-22.

229. For at least these reasons, it is my opinion that Bennett discloses that, “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.”

2. Claim 9 Is Rendered Obvious by Bennett in View of Bruce

230. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 9 of the '740 patent. In particular, a person of ordinary skill in the art would have been motivated to combine Bennett's mapping table with Bruce's approach of “expandable architecture,” which provides “inherent parallelism [that] improves performance,” to include a “set of access parameters include[ing] a bus identifier, an FDE identifier and a group identifier.” *See* Ex-1006 at 5:45-47.

231. Bennett and Bruce expressly recognize issues related to spinning disk drives, as well as the benefits of solid state non-volatile storage. *See* Ex-1007 at 1:20-34 (noting that “rotating magnetic medium such as hard drives . . . is unsuitable for the mobile and handheld environment”); Ex-1006 at 1:35-42 (noting that hard drives “suffer from their inherent fixed size” and cannot be expanded). At the same time, both Bennett and Bruce recognize that solid state non-volatile storage had certain

shortcomings, including shortcomings related to data management, host addressing schemes, and the erasure and programming of information. Ex-1007 at 2:57-4:8; Ex-1006 at 1:45-52. Accordingly, Bennett and Bruce are both directed to improving the function of solid state non-volatile memory by resolving these shortcomings in similar ways, thereby optimizing memory performance. Ex-1007 at 4:4-8; Ex-1006 at 2:40-3:18.

232. For example, both Bennett and Bruce are directed to making, using, and/or enabling systems and methods for optimizing memory operations through the use of interleaving and parallel memory operations. As noted above, Bennett explains that “[i]n order to maximize programming speed and erase speed, ***parallelism*** is exploited as much as possible by arranging for multiple pages of information . . . to be ***programmed in parallel***, and for multiple MEUs [minimum erasable units] to be erased in parallel.” Ex-1007 at 8:26-30; *see also id.* at 2:57-59 (“In order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in ***parallel***.”) (all emphasis added). Thus, the mapping table disclosed in Bennett provides a more “efficient use of system resources” by allowing “multiple logical groups to be updated concurrently,” which “increases efficiency and reduces overhead.” *Id.* at 4:50-52. Similarly, Bruce’s stated objective is to provide “[a] high-performance

flash-memory system performing flash operations in *parallel* . . . even when expansion flash chips are added.” Ex-1006 at 2:44-46. This “inherent *parallelism* improves performance by allowing multiple flash operations to be performed at the same time.” *Id.* at 13:32-35 (emphasis added); *see also id.* at 13:45-46 (noting that the disclosed system “is efficient yet allows multiple flash operations to occur at the same time.”).

233. Because Bennett and Bruce are in the same field, teach analogous solutions to the same issues, and have the same purpose of enabling and improving parallel memory operations, a POSITA would have been motivated to combine these references. In particular, a POSITA would have been motivated to incorporate Bruce’s physical access parameters—which enable an expandable system architecture—into Bennett’s mapping table to provide the benefit of a mapping table covering an expandable solid state storage system capable of improving parallel memory access. This motivation stems from the references themselves, which are both directed to improving and increasing parallel memory operations. For example, Bruce explains the addition of flash memory chips in a memory device increases busy signals. Ex-1006 at 2:9-11. Moreover, connecting busy signals “can limit performance as parallel flash operations driving the same busy line can interfere with each other” and can also “limit[] parallelism by not providing separate indications

of when each flash-memory chip has completed an operation.” *Id.* at 2:12-21; *see also* Ex-1007 at 3:31-37 (noting that the initialization and preparation of directory information takes time, “all the more so with ever increasing flash memory capacity.”). Accordingly, the benefits provided by Bruce align directly with the purpose of Bennett. Thus, a POSITA would have been motivated to combine Bruce’s additional access parameters with Bennett’s mapping table, which already contains access parameters as explained above, as this would have allowed Bennett’s mapping table to further leverage the benefits of solid state non-volatile memory connected to a host system while maintaining and enhancing parallel memory operations.

234. Moreover, including the additional access parameters taught by Bruce in Bennett’s mapping table combines well-known prior art elements and techniques, according to known methods, to yield the predictable result of a mapping table that accounts for an expandable solid state storage device. This is confirmed by the fact that Bennett already teaches a table comprising fields for access parameters, and Bruce simply expands on types of access parameters that can be added to a mapping table. In addition, Bruce’s teachings provide known techniques to use in Bennett’s mapping table in the same way by, for example, providing a flexible mapping table that can translate conventional host commands while also providing the ability to

account for a device with expandable capacity.

235. Including the access parameters taught by Bruce in Bennett's mapping table would have also been straightforward and predictable, and would have required minimal modification. Like Bruce, Bennett recognizes that address translation between the host address system and the solid state memory must be handled independently from the host and within the solid state memory device. Ex-1007 at 9:7-10:6 and Figure 6; Ex-1006 at 2:40-55. Accordingly, incorporating Bruce's additional access parameters would simply require altering Bennett's mapping table to add additional physical access parameters to account for limited additional hardware in an expandable system, such as distinct buses, DMA engines, and flash chip groups.

236. Moreover, a person of ordinary skill in the art would have had a very reasonable expectation of success in doing so because modifying Bennett's mapping table with Bruce's access parameters would require only minor changes to the table while still maintaining the same ability to receive commands over the same type of host interface. A person of ordinary skill in the art would have understood that no external change or new host command and/or interface design would have been required.

237. A motivation to combine Bennett with Bruce is further demonstrated

by the fact that Bruce is a continuation-in-part of the '006 patent. Ex-1006. Like Bennett, the '006 patent is directed to a re-mapping “*table* for a flash-memory system [that] has a *plurality of entries*.” Ex-1009 at 2:62-64 (emphasis added); *id.*, Abstract. These entries have “*a physical-block-address field* that contains a *physical block address* of a block in an array of flash memory devices.” *Id.* at 2:65-67 (emphasis added).

238. Moreover, the purpose of the '006 patent is to provide a more “efficient” table “to minimize excess writes to flash memory while re-mapping addresses” to physical memory. *Id.* at 2:53-59. Bennett similarly recognizes issues related to re-mapping or “updating” the physical memory location, noting that existing processes are “inefficient” and “also result[] in a higher frequency of erase recycling of the memory block, which is undesirable in view of the limited endurance of this type of memory device.” Ex-1007 at 3:3-17. Accordingly, like the '006 patent (Bruce’s parent), Bennett is directed to a high capacity nonvolatile memory that operates “without the aforementioned problems.” *Id.* at 4:4-8.

239. In summary, a POSITA would have simply applied Bruce’s known access parameters to Bennett’s table—which also includes access parameters—to alter Bennett’s table to include an expandable system. As a result, it is my opinion that it would have been obvious to a POSITA to combine Bennett with Bruce.

- a. **[Claim 9]: “The mapping table of claim 1, wherein said set of access parameters includes a bus identifier, an FDE identifier and a group identifier.”**

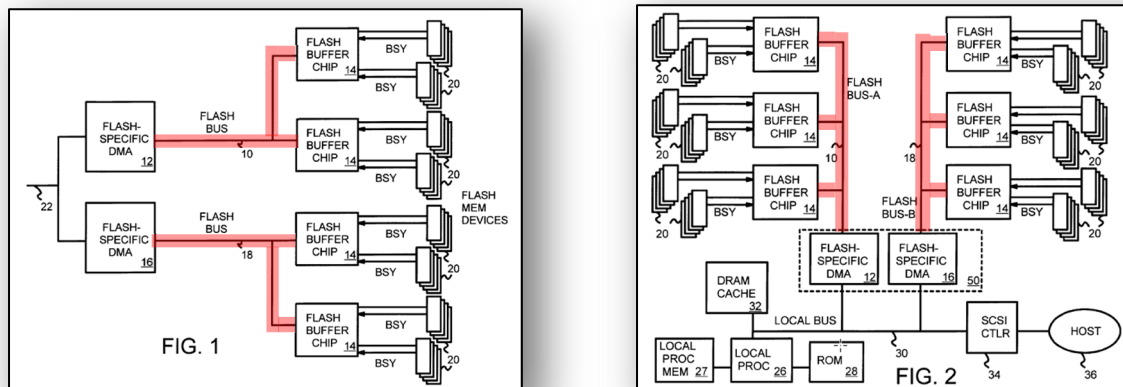
240. As explained above, it is my opinion that Bennett discloses all the elements of claim 1. *See supra* § X.B.1. It is also my opinion that Bennett in combination with Bruce discloses all the elements of claim 9, thereby rendering claim 9 obvious.

i. **Bruce Discloses Multiple Buses With Bus Identifiers**

241. It is my opinion that Bruce discloses access parameters in the form of bus identifiers. Bruce discloses two flash buses “coupled to a flash-specific DMA controller,” which have “shared lines that transmit the sequence of command bytes and address bytes from the flash-specific DMA controller.” Ex-1006 at 2:56-59; 3:62-64 (explaining that a first flash bus “transfers data, addresses, and commands over shared address/data/command lines”) 4:10-21 (explaining that a second flash bus does the same). The flash buses are also coupled to flash buffer chips, which pass commands and addresses from the flash-specific DMA controller to the flash memory. *Id.* at 2:63-65; 5:7-12 (explaining that each flash bus has eight shared data/address/command lines, as well as two lines for an encoded command). Because the flash buses are connected to different controllers, each flash bus “can operate at the same time, allowing flash operations to be initiated and processed in

parallel.” *Id.* at 5:18-20.

242. Bruce identifies these buses using the identifiers “**flash bus 10/ Flash Bus-A**” and “**flash bus 18/ Flash Bus-B**,” which are illustrated in annotated Figures 1 and 2, below:



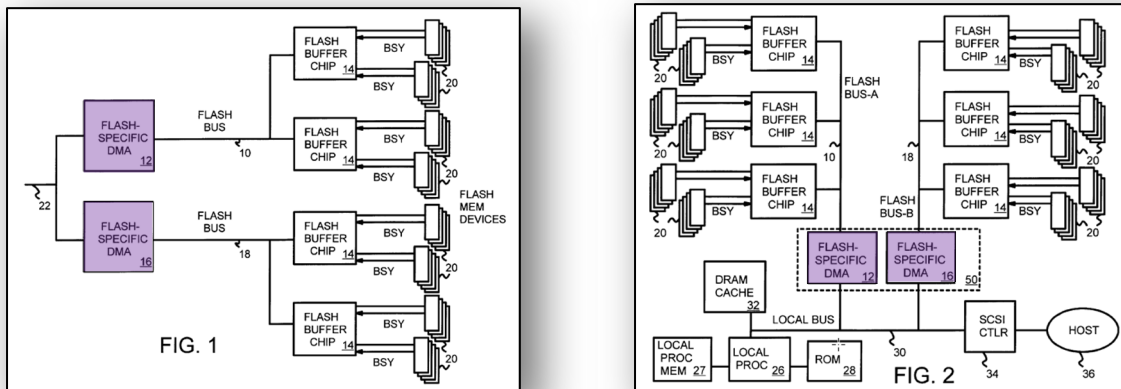
243. As a result, it is my opinion that Bruce’s disclosure of **flash bus 10/ Flash Bus-A** and **flash bus 18/ Flash Bus-B** teaches a bus identifier associated with a PBA, or a “unique addressable physical memory location,” pursuant to Petitioners’ proposed construction.

ii. Bruce Discloses Multiple FDEs With FDE Identifiers

244. It is my opinion that Bruce discloses access parameters in the form of FDE identifiers that comprise identifiers for specific DMA state machines. Bruce discloses that the flash buses are “coupled to [a] flash specific DMA controller,” Ex-1006 at 2:56-57, which “generates command, address, and data sequences” to the

flash memory chips, *id.* at 3:59-62; 4:8-10; 5:2-6 (explaining that the DMA controllers not only transfer data, “but also generate the appropriate command and address bytes that are transmitted over the same lines of the flash bus as the data”).

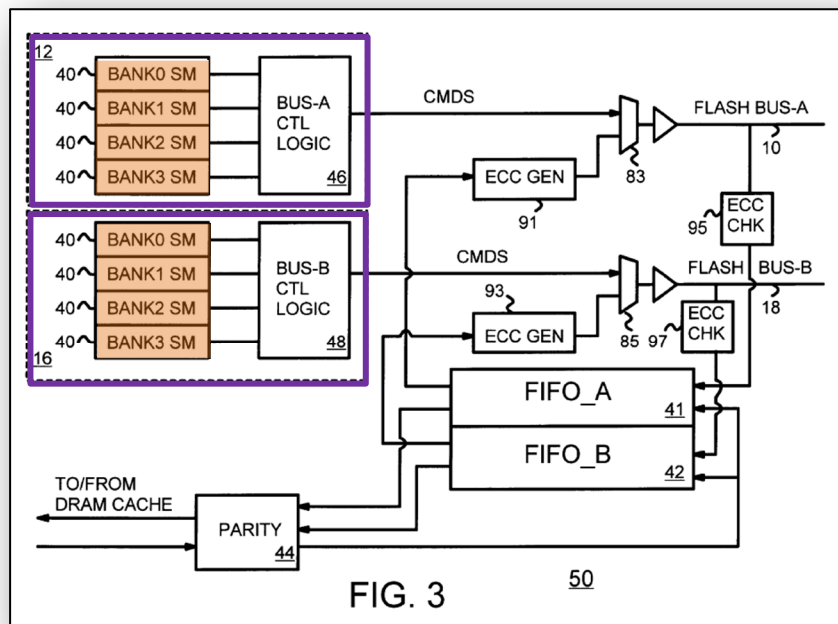
Annotated Figures 1 and 2, below, illustrate dual **flash-specific DMA controllers 12 and 16**:



245. Bruce also discloses that each DMA controller contains four “state machines,” each of which control one of four banks of flash-memory chips. *Id.* at Abstract; *see also id.* at 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-55. Having four DMA state machines “allows four flash-memory chips to be accessed at once for each flash bus, for a total of eight flash accesses.” *Id.* at 6:55-57. Accordingly, the DMA state machines allow data to 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; 13:32-35.

246. Annotated Figure 3, shown below, illustrates the four **state machines** (identified as “**Bank0 SM**,” “**Bank1 SM**,” “**Bank2 SM**,” and “**Bank3 SM**”) within each flash specific **DMA controller**, thereby providing total of eight state machines.

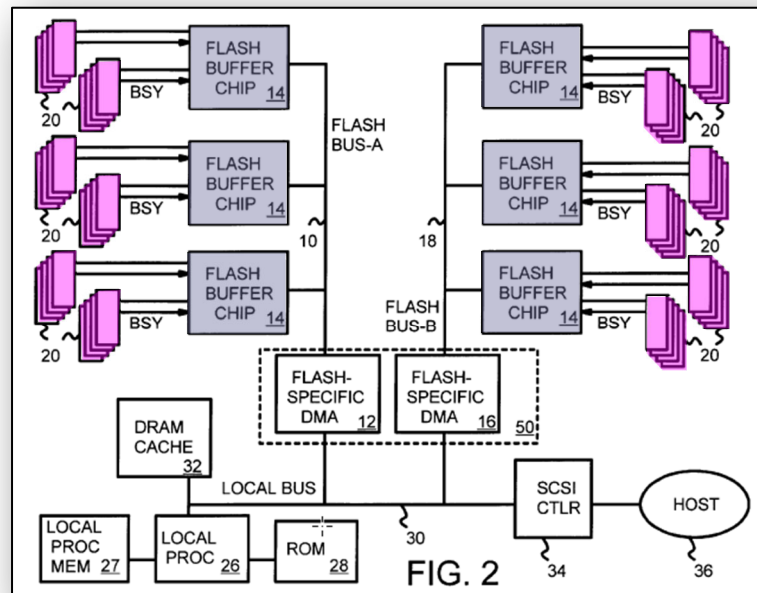


247. Accordingly, it is my opinion that, by disclosing multiple state machines that control different banks of memory chips—each of which has its own designation or “identifier” (e.g., “Bank0 SM” etc.)—Bruce discloses “an identifier for a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices” pursuant to Petitioners’ proposed construction of FDE identifier. As a result, these state machines are FDEs

pursuant to Petitioners’ proposed construction. Bruce also teaches that each of the FDEs are associated with a PBA, or “unique addressable physical memory location” pursuant to Petitioners’ construction of PBA. *See, e.g., id.* at 6:53-57; 11:23-25. For at least this reason, it is my opinion that Bruce discloses an “FDE identifier” associated with a PBA.

iii. Bruce Discloses Multiple Groups with Group Identifiers

248. It is my opinion that Bruce also discloses access parameters in the form of group identifiers. As noted above, Bruce discloses multiple memory banks and teaches that up to “four banks of flash-memory chips” can be connected to one of a plurality of flash buffer chips. Ex-1006 at 12:59-62. Each memory bank comprises a plurality of flash memory chips. *Id.* at 2:66-3:1. For example, each memory bank may contain eight flash memory chips, leading to a total of 32 flash-memory chips (8 chips x 4 banks) per flash buffer. *Id.* at 5:31; 5:58-61. The flash memory **chip banks**, which are connected through a plurality of **flash buffer chips**, are illustrated in annotated Figure 2, below.



249. Bruce further teaches that each individual bank of flash-memory chips “can be separately accessed, allowing many flash operations to be performed in parallel.” *Id.* at 5:21-23; 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”). Separate access is enabled by the fact that the flash memory chips connected to each flash buffer are “uniquely addressed.” *Id.* at 7:62-67. Accordingly, to access the memory, the DMA controller sends specific device addresses for each chip to the flash buffer, which then decodes the address into chip selects. *Id.* at 8:46-49; 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); *see also id.* at 2:40-42 (noting that, in the invention, a DMA controller inputs “commands and addresses . . . to a

flash-memory chip”).

250. Further, as explained above, each FDE (e.g., state machine) controls an entire memory bank, which comprises a plurality of flash chips. Accordingly, by teaching that each chip in a memory bank has a unique address sent by the DMA controller, Bruce teaches the use of group identifiers—namely, “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE” pursuant to Petitioners’ proposed construction. Bruce also discloses that the group identifiers are associated with a PBA, or a “unique addressable physical memory location,” pursuant to Petitioners’ construction of PBA. *See, e.g., id.* at 8:46-49; 10:17-19; 2:40-42.

251. For at least these reasons, it is my opinion that Bruce discloses a “group identifier” associated with a PBA.

**iv. It Would Have Been Obvious to Modify
Bennett’s Mapping Table to Include the
Identifiers Disclosed in Bruce**

252. As discussed above, it is my opinion that Bruce discloses access parameters that include bus identifiers, FDE identifiers, and group identifiers. Moreover, as previously explained, it is my opinion that it would have been obvious for a person of ordinary skill in the art to combine Bennett with Bruce. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination

with Bruce renders obvious the “mapping table of claim 1, wherein said set of access parameters includes a bus identifier, an FDE identifier and a group identifier.”

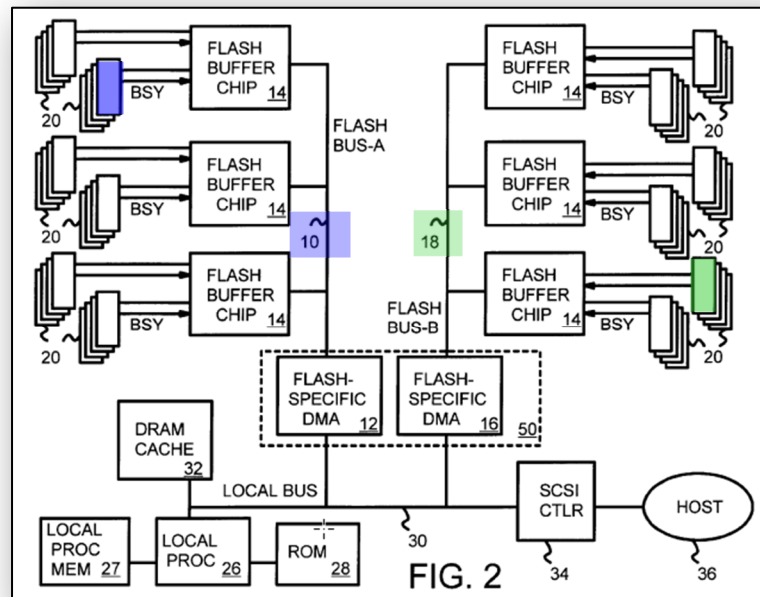
3. Claim 10 Is Rendered Obvious by Bennett in View of Bruce

253. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 10.

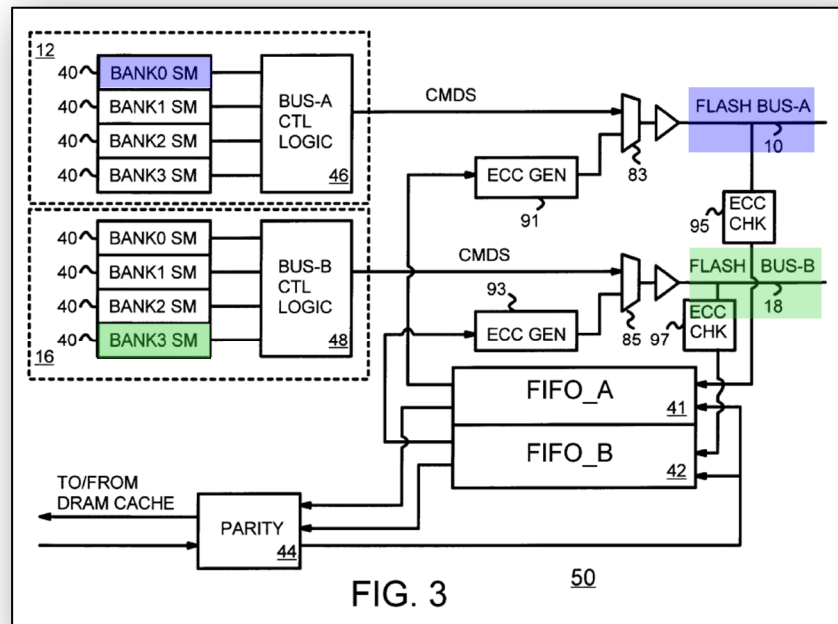
[Claim 10]: “The mapping table of claim 9, wherein said first and second PBA include different bus identifiers, different FDE identifiers and different group identifiers.”

254. It is my opinion that the combination of Bennett and Bruce teach all of the elements of claim 9. *See supra* § X.B.2. It is also my opinion that Bennett and Bruce teach all of the elements of claim 10.

255. As explained above, Bruce teaches the use of bus identifiers, FDE identifiers, and group identifiers. *See supra* § X.B.2. Bruce also teaches that these identifiers may be different for a first and second memory chip. For example, Bruce demonstrates that a PBA (i.e., a unique addressable physical memory location) for a **first memory chip** on bus 10 would have one bus identifier (“**bus 10**”), while a PBA for a **second memory chip** on bus 18 would have a different bus identifier (“**bus 18**”), as illustrated in annotated Figure 2, below:



256. As explained above, Bruce also teaches that each bank of memory devices is controlled by its own FDE engine (i.e., DMA state machine). *See* Ex-1006 at Abstract; 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-57; *see also supra* § X.B.2.a.ii. Thus, for example, a PBA on a **first memory chip** located on **bus 10** could have one FDE identifier (e.g., “**Bank0 SM**”), while a PBA for a **second memory chip** located on **bus 18** could have a different FDE identifier (e.g., “**Bank3 SM**”), as illustrated in annotated Figure 3, below:



257. Likewise, Bruce teaches that each chip within a memory bank has a unique address that allows it to be separately accessed. *Id.* at 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”); 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); *see also supra* § X.B.2.a.iii. Accordingly, a PBA on a **first memory chip having a specific address**—or group identifier—can be different from a PBA on a **second memory chip having a specific address**.

258. In this manner, Bruce teaches that a first and second PBA can be identified by the following access parameters, which are different:

	Access Parameters		
	Bus Identifier	FDE Identifier	Group Identifier
First unique physical location (PBA)	Bus 10/Bus-A	Bank0 SM	Specific address of first chip
Second unique physical location (PBA)	Bus 18/Bus-B	Bank3 SM	Specific address of second chip

259. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art to combine Bennett with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the “mapping table of claim 9, wherein said first and second PBA include different bus identifiers, different FDE identifiers and different group identifiers.”

4. Claim 11 Is Rendered Obvious by Bennett in View of Bruce

260. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 11.

- a. **[Claim 11]: “The mapping table of claim 9, wherein said first and second PBA respectively include different combinations of said bus identifiers, said FDE identifiers, and said group identifiers.”**

261. It is my opinion that the combination of Bennett and Bruce teach all of the elements of claim 9. *See supra* § X.B.2. It is also my opinion that the combination of Bennett and Bruce teach all of the elements of claim 11.

262. As explained above, Bruce teaches the use of bus identifiers, FDE identifiers, and group identifiers, which may be different for each memory chip. *See supra* §§ X.B.2.a, X.B.3. As a result, Bruce teaches that a first and second PBA can be identified by the following parameters:

	Access Parameters		
	Bus Identifier	FDE Identifier	Group Identifier
First unique physical location (PBA)	Bus 10/Bus-A	Bank0 SM	Specific address of first chip
Second unique physical location (PBA)	Bus 18/Bus-B	Bank3 SM	Specific address of second chip

See supra § X.B.3. As this table illustrates, each PBA may be represented by different combinations of bus identifiers, FDE identifiers, and group identifiers.

263. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce's bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is

my opinion that Bennett in combination with Bruce renders obvious the “mapping table of claim 9, wherein said first and second PBA respectively include different combinations of said bus identifiers, said FDE identifiers, and said group identifiers.”

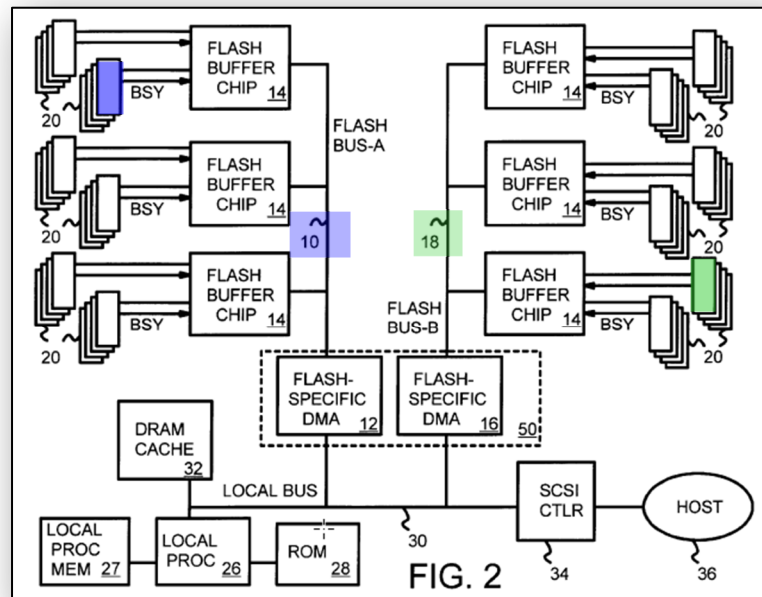
5. Claim 12 Is Rendered Obvious by Bennett in View of Bruce

264. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 12.

a. [Claim 12]: “The mapping table of claim 1, wherein said first and second PBA include different bus identifiers.”

265. It is my opinion that Bennett teaches all the elements of claim 1. *See supra* § X.B.1. It is also my opinion that the combination of Bennett and Bruce teach all of the elements of claim 12.

266. As explained above, Bruce teaches the use of bus identifiers. *See supra* §§ X.B.2.a.i, X.B.3. Bruce also teaches that these identifiers may be different for a first and second memory chip. For example, Bruce demonstrates that a PBA (i.e., a unique addressable physical memory location) for a **first memory chip** on bus 10 would have one bus identifier (“**bus 10**”), while a PBA for a **second memory chip** on bus 18 would have a different bus identifier (“**bus 18**”), as illustrated in annotated Figure 2, below:



267. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce's bus identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the "mapping table of claim 1, wherein said first and second PBA include different bus identifiers."

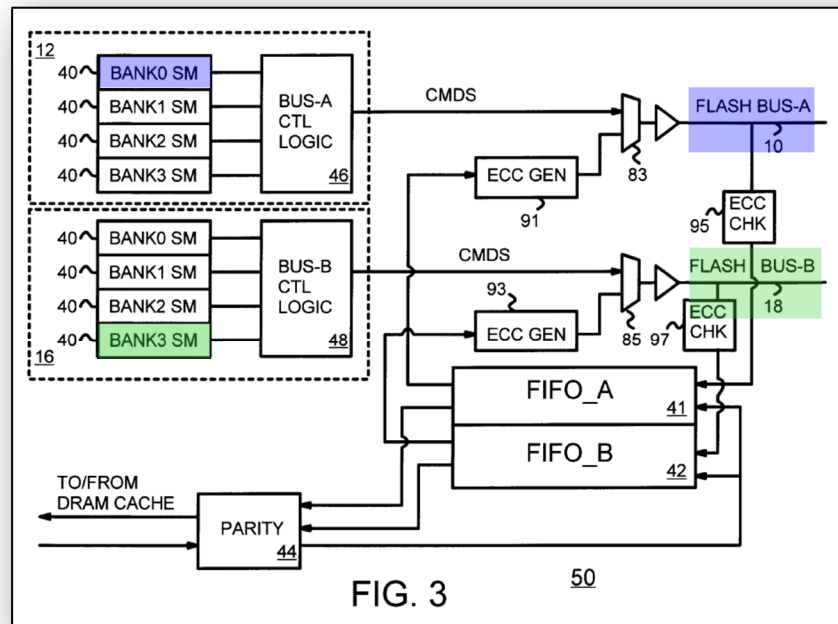
6. Claim 13 Is Rendered Obvious by Bennett in View of Bruce

268. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 13.

- a. **[Claim 13]: "The mapping table of claim 1, wherein said first and second PBA include different FDE identifiers."**

269. It is my opinion that Bennett teaches all the elements of claim 1. *See supra* § X.B.1. It is also my opinion that the combination of Bennett and Bruce teach all of the elements of claim 13.

270. As explained above, Bruce teaches the use of FDE identifiers. *See supra* §§ X.B.2.a.ii, X.B.3. Bruce also teaches that these FDE identifiers may be different for a first and second memory chip. For example, Bruce teaches that each bank of memory devices is controlled by its own FDE engine (i.e., DMA state machine). *See* Ex-1006 at Abstract; 11:23-25 (noting that four state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-57; *see also supra* § X.B.2.a.ii. Thus, for example, a PBA on a **first memory chip** located on **bus 10** could have one FDE identifier (e.g., “**Bank0 SM**”), while a PBA for a **second memory chip** located on **bus 18** could have a different FDE identifier (e.g., “**Bank3 SM**”), as illustrated in annotated Figure 3, below:



271. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce’s bus identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the “mapping table of claim 1, wherein said first and second PBA include different FDE identifiers.”

7. Claim 14 Is Rendered Obvious by Bennett in View of Bruce

272. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 14.

- a. [Claim 14]: “The mapping table of claim 1, wherein said first and second PBA respectively include a first

group identifier and a second group identifier.”

273. It is my opinion that Bennett teaches all the elements of claim 1. *See supra* § X.B.1. It is also my opinion that the combination of Bennett and Bruce teach all of the elements of claim 14.

274. As discussed above, Bruce teaches the use of group identifiers, or addresses that identify one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE. *See supra* §§ X.B.2.a.iii, X.B.3. Bruce also teaches that each chip within a memory bank has a unique address that allows it to be separately accessed. Ex-1006 at 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”); 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”); *see also supra* § X.B.2.a.iii. Accordingly, a PBA on a **first memory chip having a specific address**—or group identifier—can be different from a PBA on a **second memory chip having a specific address**.

275. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is

my opinion that Bennett in combination with Bruce renders obvious the “mapping table of claim 1, wherein said first and second PBA respectively include a first group identifier and a second group identifier.”

8. Claim 15 Is Rendered Obvious by Bennett in View of Bruce

276. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 15.

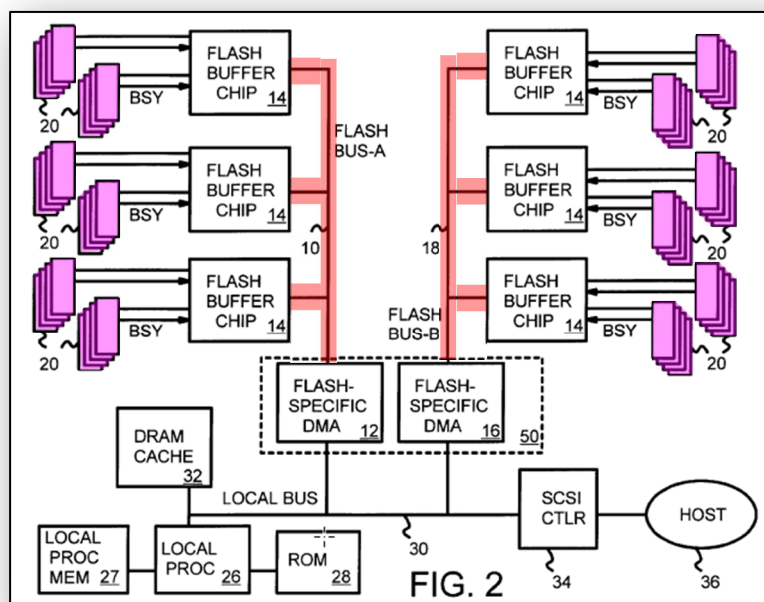
a. [Claim 15a]: “The mapping table of claim 1: wherein said memory store includes a set of solid state memory devices that are coupled to a set of buses and that are controlled by a set of FDEs; and”

277. It is my opinion that Bennett teaches all the elements of claim 1. *See supra* § X.B.1. It is also my opinion that the combination of Bennett and Bruce teach all of the elements of claim 15.

278. Bruce discloses a memory store that includes a set of solid state memory devices. For example, Bruce discloses multiple memory banks, each of which comprise a plurality of flash memory chips. Ex-1006 at 2:66-3:1; 5:31; 5:58-61. Each of these banks can be “separately accessed, allowing many flash operations to be performed in parallel.” *Id.* at 5:21-23. Accordingly, each of these memory banks comprise a set of solid state flash memory devices. *See supra* §§ X.B.2.a.ii, X.B.2.a.iii.

279. Bruce also discloses that the memory banks are coupled to a set of flash

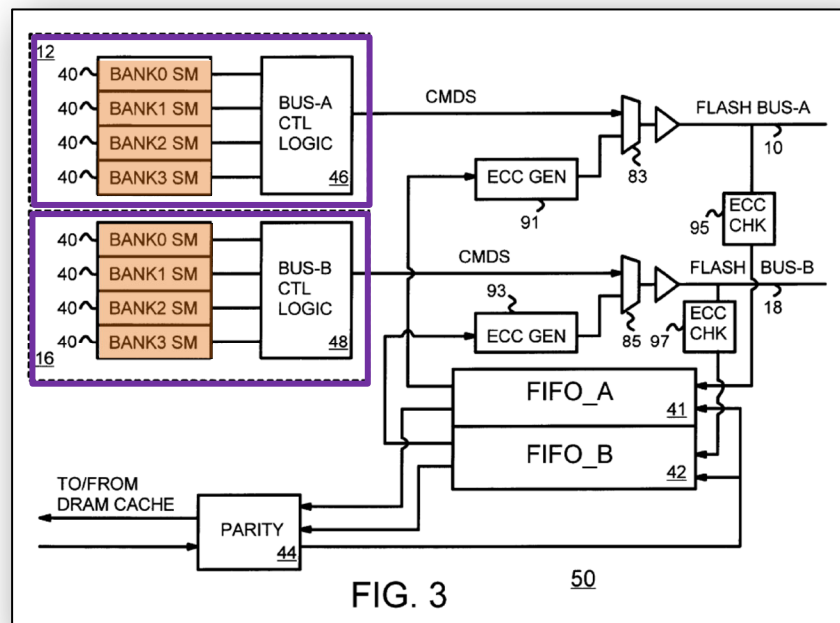
buses 10 and 18, which are “coupled to a flash-specific DMA controller” and have “shared lines that transmit the sequence of command bytes and address bytes from the flash-specific DMA controller.” *Id.* at 2:56-59; 3:62-64; 4:10-21; *see also supra* § X.B.2.a.iii. Annotated Figure 2, below, illustrates the set of **flash buses (bus 10/Flash Bus-A and bus 18/Flash Bus-B)** coupled to the **sets of solid state memory devices (flash memory banks)** through flash buffer chips:



280. Bruce further teaches that the sets of solid state memory devices are controlled by a set of FDEs. As explained above, Bruce teaches multiple DMA controllers containing four “state machines,” each of which control one of four banks of flash-memory chips. *Id.* at Abstract; *see also id.* at 11:23-25 (noting that four

state machines “are provided for each DMA controller to allow control of four separate devices”); 6:53-55. Having four DMA state machines “allows four flash-memory chips to be accessed at once for each flash bus, for a total of eight flash accesses.” *Id.* at 6:55-57. Accordingly, these state machines are “a component that controls multiple flash devices and facilitates high speed data transfers to and from a group of flash memory devices” pursuant to Petitioners’ proposed construction of FDE. *See supra* § X.B.2.a.ii.

281. Annotated Figure 3, shown below, illustrates the four **state machines** (identified as “**Bank0 SM**,” “**Bank1 SM**,” “**Bank2 SM**,” and “**Bank3 SM**”) within each flash specific **DMA controller**, thereby providing total of eight state machines:



282. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce's bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the mapping table of claim 1 "wherein said memory store includes a set of solid state memory devices that are coupled to a set of buses and that are controlled by a set of FDEs."

b. [Claim 15b]: "wherein said access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively controlled by different FDEs from said set of FDEs."

283. It is my opinion that Bruce discloses this claim element. Bruce teaches that each individual bank of flash-memory chips "can be separately accessed, allowing many flash operations to be performed in parallel." Ex-1006 at 5:21-23; 6:55-57 (noting that "four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses"). Separate access is enabled by the fact that the flash memory chips connected to each flash buffer are "uniquely addressed." *Id.* at 7:62-67. Accordingly, to access the memory, the DMA controller sends specific device addresses for each chip to the flash buffer, which then decodes the address into chip selects. *Id.* at 8:46-49; 10:17-19 ("The second cycle of the new request phase sends a device address for one of the flash-memory chips connected

to the flash buffer chips.”); *see also id.* at 2:40-42 (noting that, in the invention, a DMA controller inputs “commands and addresses . . . to a flash-memory chip”). As also explained above, each FDE (e.g., state machine) controls an entire memory bank comprising a plurality of flash chips.

284. Accordingly, by teaching that each chip in a memory bank has a unique address sent by the DMA controller, Bruce teaches the use of group identifiers—namely, “an identifier for identifying one flash memory device from another flash memory device from a set of flash devices that are controlled by the same FDE” pursuant to Petitioners’ proposed construction. Moreover, Bruce discloses that each flash buffer chip may be connected to multiple banks of flash memory chips, each of which has its own state machine or FDE. Accordingly, the address (or group identifier) for the chips within each separate bank attached to flash buffer chip would identify “at least a portion of the set of memory devices” that are controlled by a different FDE from the set of FDEs.

285. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the mapping

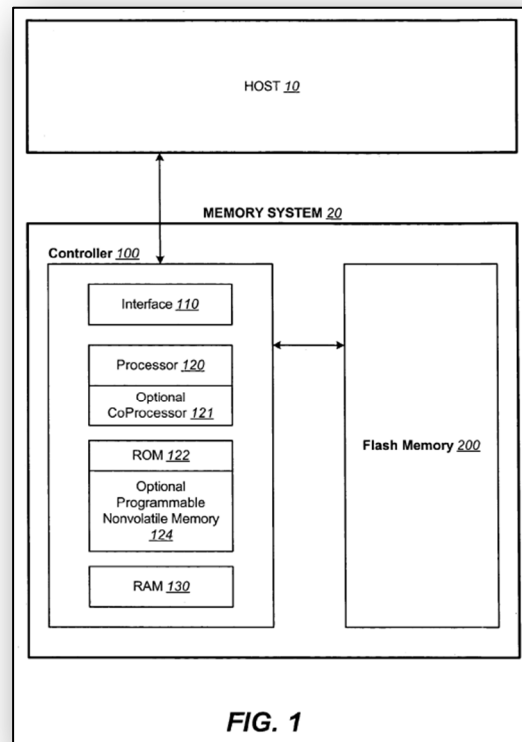
table of claim 1 “wherein said access parameter includes a group identifier for identifying at least a portion of said set of memory devices that are respectively controlled by different FDEs from said set of FDEs.”

9. Claim 32 Is Anticipated by Bennett

286. It is my opinion that Bennett teaches every element of claim 32, thereby anticipating this claim.

- a. **[Claim 32 pre]: “An electronic storage device that includes a memory system that uses a memory table for increasing the likelihood that an operational load imposed on the storage apparatus during the processing of an I/O transaction request will be optimally distributed across storage device resources, said I/O transaction request received by said storage device from a requesting host, said electronic storage device comprising:”**

287. To the extent the preamble is limiting, it is my opinion that Bennett discloses this claim element. Bennett discloses a memory system that operates with a host and can take the form of an electronic storage device such as a memory card or embedded memory system. Ex-1007 at 6:24-27. This system is illustrated in Figure 1, shown below:



288. Bennett discloses that this system uses tables for “[m]apping between logical groups and physical groups (metablocks).” *Id.* at 23:61-64; 7:9-11; 9:55-58. For example, Bennett discloses a Group Address Table—or “GAT”—that “is a list of [physical] metablock addresses for all logical groups of host data in the memory system.” *Id.* at 24:19-21. Bennett further discloses that this table increases the likelihood that an operational load will be optimally distributed by using “*parallelism* . . . as much as possible by arranging for multiple pages of information . . . to be *programmed in parallel*, and for multiple MEUs [minimum erasable units] to be *erased in parallel*.” *Id.* at 8:26-30; *see also id.* at 2:57-59 (“In

order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in *parallel*.”) (all emphasis added).¹³ This mapping table provides a more “efficient use of system resources” by allowing “multiple logical groups to be updated concurrently,” which “increases efficiency and reduces overhead.” *Id.* at 4:50-52.

289. Bennett further discloses that the mapping table optimizes memory operations in response to processing of an I/O transaction with a host. For example, Bennett discloses that the host “accesses the memory” and reads or writes “to the memory system in unit[s] of logical clusters, each consisting of one or more logical sectors.” *Id.* at 6:53-59; *see also id.* at 6:61-65 (explaining that the host “issues a command to the memory system 20 to read or write a segment containing a string of logical sectors of data with contiguous addresses”). As a result, the system disclosed

¹³ As explained above, the ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *Supra* n. 3. Likewise, the district court has held that examples of optimization “include[e] contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one access parameter.” Ex-1008 at 30. Bennett teaches each of these examples. *Infra* § X.B.9.b, X.B.9.c, X.B.9.e.

by Bennett “maps the logical address from the host to a physical memory location.”
Id. at 9: 20-22.

290. Bennett also discloses “contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one parameter,” *infra* §§ X.B.9.b, X.B.9.c, X.B.9.e, which the district court has identified as examples of optimized memory operations. Ex-1008 at 30.

291. For at least these reasons, it is my opinion that Bennett discloses “[a]n electronic storage device that includes a memory system that uses a memory table for increasing the likelihood that an operational load imposed on the storage apparatus during the processing of an I/O transaction request will be optimally distributed across storage device resources, said I/O transaction request received by said storage device from a requesting host.”

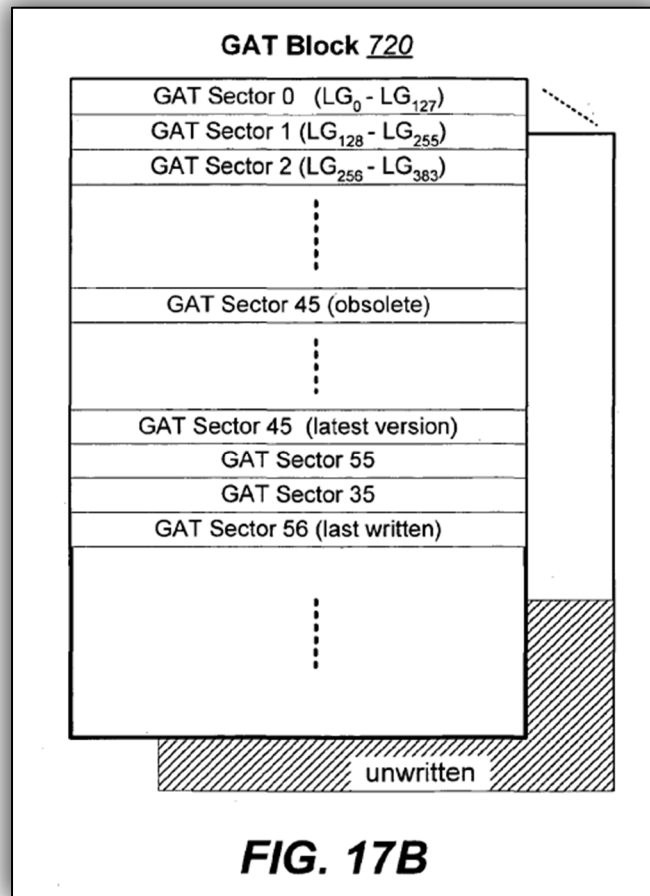
b. [Claim 32a]: “a mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said first and second LBA sets including respective sets of consecutive LBAs;”

292. It is my opinion that Bennett discloses this claim element. As explained above, Bennett discloses tables for “[m]apping between logical groups and physical groups (metablocks).” Ex-1007 at 23:61-64; 7:9-11; 9:55-58. For example, Bennett discloses a Group Address Table—or “GAT”—that “is a list of [physical] metablock addresses for all logical groups of host data in the memory system.” *Id.* at 24:19-21.

In particular, the GAT “contains one entry for each logical group, ordered sequentially according to logical address.” *Id.* at 24:21-22; *see also id.* at 25:1-3 (“GAT sectors within a GAT block each contain logical to physical mapping information for 128 contiguous logical groups.”); 9:55-58 (explaining that the GAT tracks “mapping between logical groups of sectors and their corresponding metablocks”).

293. Bennett further explains that the GAT comprises a “set of sectors (referred to as GAT sectors) with entries defining metablock address for every logical group in the memory system.” *Id.* at 24:24-27. Moreover, each GAT sector “contain[s] logical to physical mapping information for 128 contiguous logical groups.” *Id.* at 25:1-3; *see also id.* at 24:21-22 (explaining that the GAT “contains one entry for each logical group, ordered sequentially according to logical address”); 24:66-25:1 (“[A] GAT block contains entries for logically contiguous set of groups in a region of logical address space.”).

294. Annotated Figure 17A, shown below, illustrates sets of consecutive logical groups within each GAT sector. For example, GAT Sector 0 contains consecutive logical groups 0-127, GAT Sector 1 contains consecutive logical groups 128-255, etc.:



295. Bennett also teaches that each logical group comprises a set of logical sectors, and that the “host system addresses data in units of logical sectors where, for example, each sector may contain 512 bytes of data.”¹⁴ *Id.* at 6:54-57; *see also id.* at 6:66-7:2 (explaining that the controller “manage[s] the storage and retrieval of

¹⁴ The '740 patent confirms that an LBA typically has a “size equal to 512 bytes.” Ex-1001 at 7:22-25.

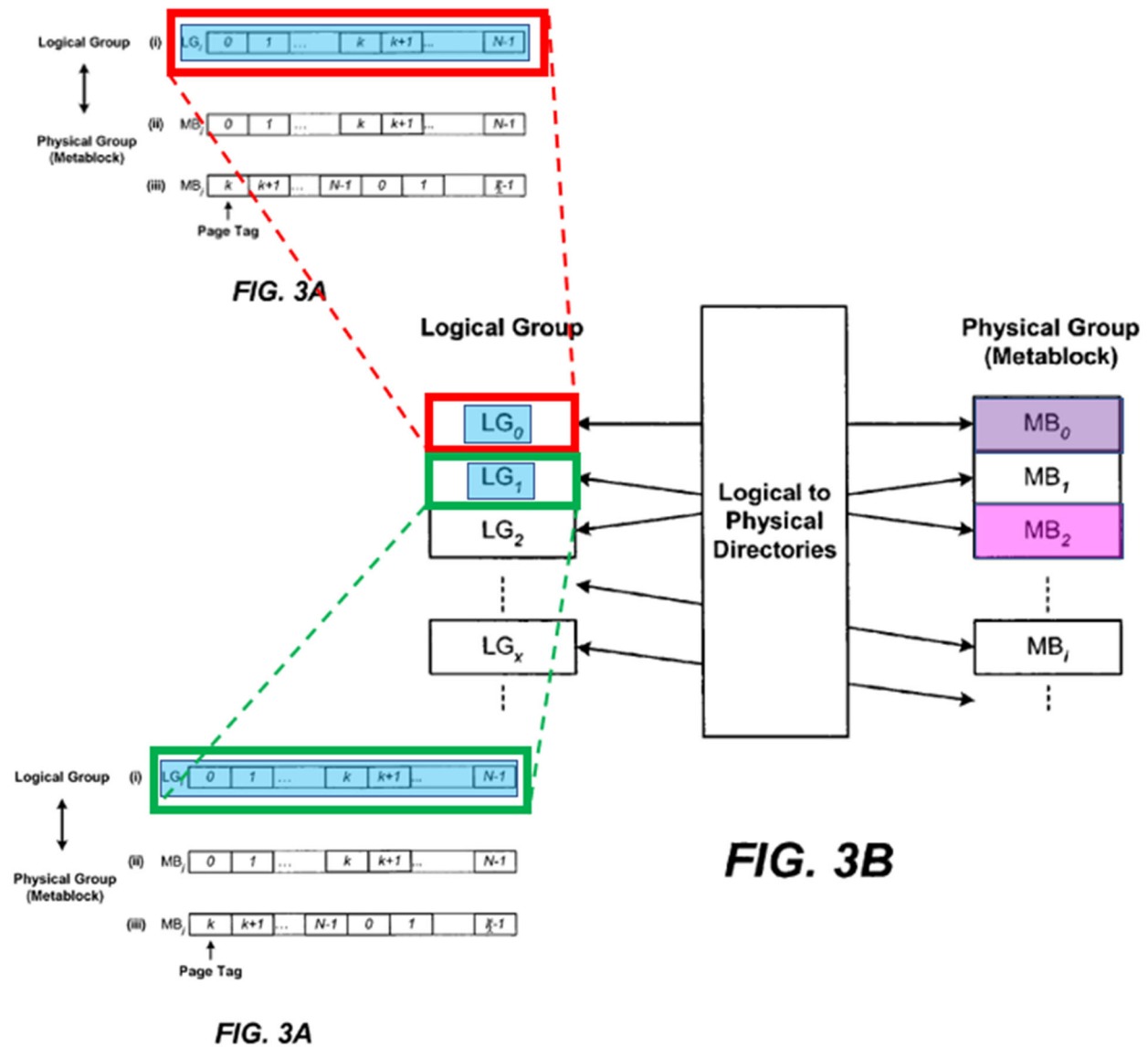
the data of host logical sectors”). As Bennett explains, the logical sectors within the logical group “are in **contiguous** logical order 0, 1, . . . , N-1.” *Id.* at 7:13-15; 6:61-65 (noting that the host issues a command “to read or write a segment containing a string of logical sectors of data with **contiguous addresses**”); 11:6-9 (explaining that “each host write is a segment of one or more sector with **contiguous** logical addresses”) (all emphasis added). Accordingly, each of Bennett’s logical sectors is an LBA—or “an address that is part of a logical addressing system used by a host”—pursuant to Petitioners’ proposed construction. Further, by disclosing multiple logical groups that each contain contiguous LBAs (sectors), Bennett teaches that each logical set (logical group) comprises consecutive LBAs.

296. Bennett also discloses mapping between the LBA sets and PBAs. For example, Bennett discloses metablocks that correspond to physical memory locations, explaining that “[t]he physical address space of the flash memory is treated as a set of metablocks.” Ex-1007 at 8:17-19; *see also id.* at 4:13-15 (noting that “[e]ach physical group (metablock) is erasable as a unit and can be used to store a logical group of data”); *id.* at 23:64-66 (stating that “[a]n address table is maintained in flash memory, containing a metablock address for every logical group in the memory system.”). Accordingly, Bennett’s metablocks are “unique addressable physical memory locations” pursuant to Petitioners’ proposed

construction of “PBA.”

297. Bennett’s table also associates the first and second LBA sets with first and second PBAs (metablocks). As Bennett explains, “[e]ach logical group is mapped to a unique metablock,” and the “mapping information is maintained in a set of logical to physical directories.” Ex-1007 at 7:38-44; *see also id.* at 23:61-64 (explaining that the GAT maps “between logical groups and physical groups (metablocks).”); *id.* at 9:56-58 (“The GAT 120 keeps track of the mapping between logical groups of sectors and their corresponding metablocks.”); *id.* at 23:64-66 (“An address table is maintained in flash memory, containing a metablock address for every logical group in the memory system.”).

298. Annotated Figures 3A and 3B, shown below, illustrate the **first logical field** (LG₀) comprising **consecutive LBAs** associated with the **first PBA** (MB₀), and the **second logical field** (LG₁) comprising **consecutive LBAs** associated with the **second PBA** (MB₂):



299. For at least these reasons, it is my opinion that Bennett teaches “a mapping table which maps a first LBA set to a first PBA, and a second LBA set to a second PBA, said first and second LBA sets including respective sets of consecutive LBAs.”

- c. [Claim 32b]: “a memory store that includes a set of flash devices, said set of flash devices includes a first

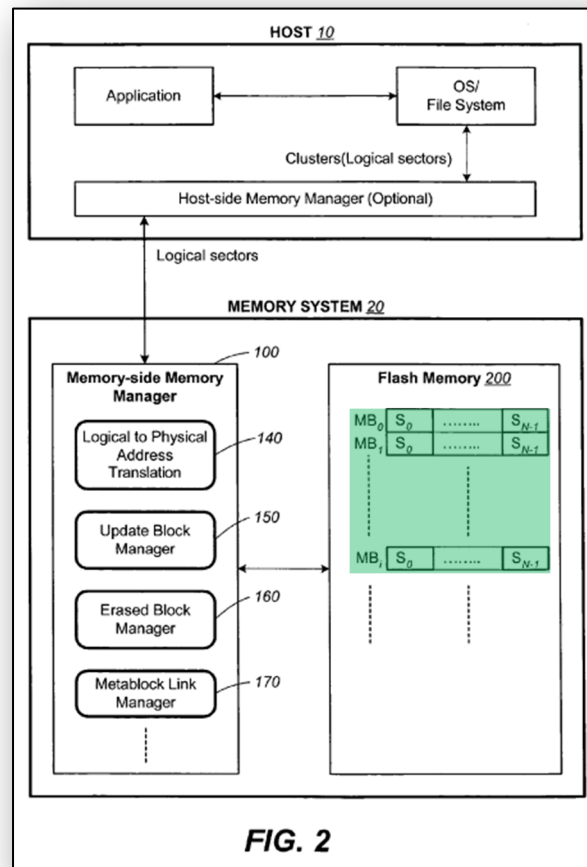
physical memory location and a second physical memory location that are addressable by said first and second PBAs respectively;”

300. It is my opinion that Bennett discloses this claim element. Bennett discloses a memory store in the form of flash memory 200, which comprises an array of memory cells that may be “distributed over one or more integrated circuit chip[s].” Ex-1007 at 6:27-31; *see also id.* at 8:34-38 (explaining that a “memory array” can be partitioned into planes that “may be distributed among one or more memory chips”). Each of these flash chips has a specific memory location. *See id.* at Abstract (noting that the “non-volatile memory system is organized in physical groups of physical memory locations”).

301. Bennett also discloses that the first and second physical locations are addressable by first and second PBAs (i.e., physical sectors). Bennett teaches that “[t]he physical address space of the flash memory is treated as a set of metablocks.” *Id.* 8:17-19. Further, each metablock represents “a group of memory locations, e.g., sectors that are erasable together.” *Id.* at 8:15-16; *see also id.* at 6:50-52 (noting that “each metablock is a group of physical sectors $S_0 \dots S_{n-1}$ ”). As such, a first physical memory location would be addressable by a first PBA and a second memory location would be addressable by a second PBA.

302. Annotated Figure 2 below illustrates memory store 200 with physical

memory locations represented by **metablocks (MB₀ . . . MB_i)** comprising a series of physical sectors (S₀ . . . S_{N-1}):



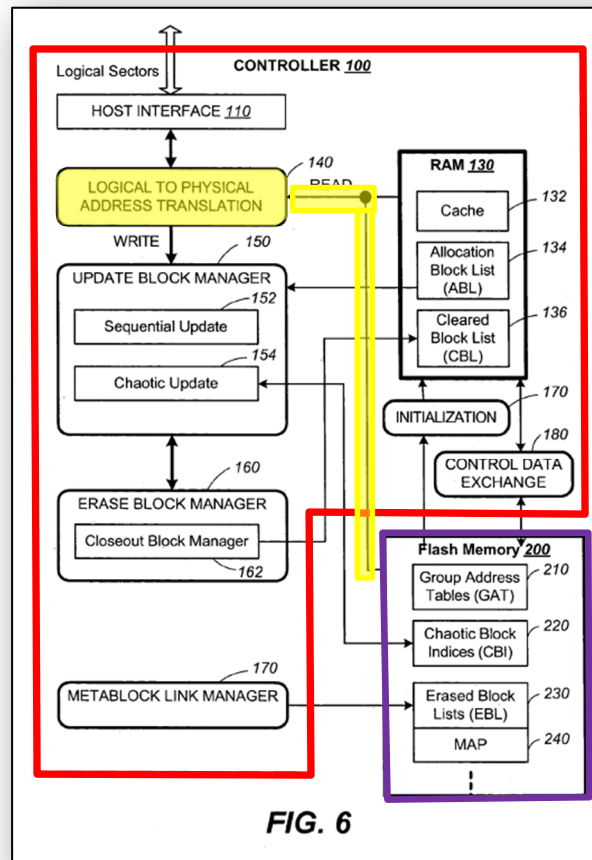
303. For at least these reasons, it is my opinion that Bennett teaches “a memory store that includes a set of flash devices, said set of flash devices includes a first physical memory location and a second physical memory location that are addressable by said first and second PBAs respectively.”

- d. **[Claim 32c]: “a memory system coupled to said memory store and disposed to use said mapping table**

during a memory operation;”

304. It is my opinion that Bennett discloses this claim element. For example, Bennett discloses a memory system¹⁵ comprising a controller 100 that is coupled to the memory store. The controller includes a logical to physical address translation module 140 that utilizes the GAT to “map[] the logical address from the host to a physical memory location.” Ex-1007 at 9:20-22. Annotated Figure 6, shown below, illustrates **controller 100** coupled to **flash memory 200**, as well as the logical to physical address translation module 140 (**highlighted in yellow**) that is connected to and uses information from the GAT during a memory operation:

¹⁵ The '740 patent describes the “memory system ” as comprising a DMA controller and mapping table. *See* Ex-1001 at 3:20-31; Figure 1 (illustrating Memory System 4).



305. For at least these reasons, it is my opinion that Bennett teaches “a memory system coupled to said memory store and disposed to use said mapping table during a memory operation.”

- e. **[Claim 32d]: “wherein said first PBA includes a first set of access parameters and said second PBA includes a second set of access parameters, and at least one difference between said first and second PBAs;”**

306. It is my opinion that Bennett discloses this claim element. For example, Bennett explains that the GAT comprises data fields for each sector, which include

the “metablock number, the page tag . . . and a flag indicating whether the metablock has been relinked.” *Id.* at 24:39-42. In this manner, Bennett teaches that each PBA (metablock) is “disposed to represent a set of access parameters” representing the physical memory location, such as identifiers for a metablock, page tag, and relinking flag. *See id.* at 24:39-42; Figures 3A, 3B, and 17A. Bennett also teaches at least one difference between these PBAs in the form of a different Metablock number (i.e., MB₀ and MB₂).

307. Combined Figures 3A and 17A, shown below, illustrate the **PBA fields** for the **first PBA** (MB₀) and the **second PBA** (MB₂):

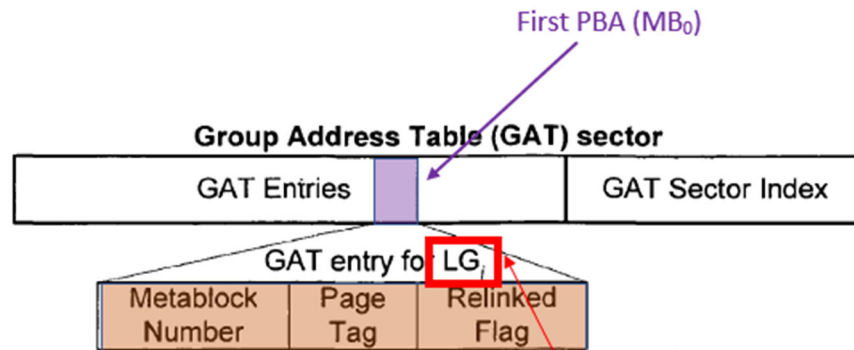


FIG. 17A

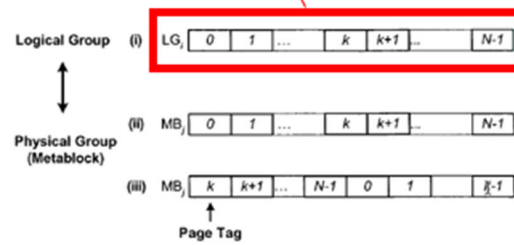


FIG. 3A

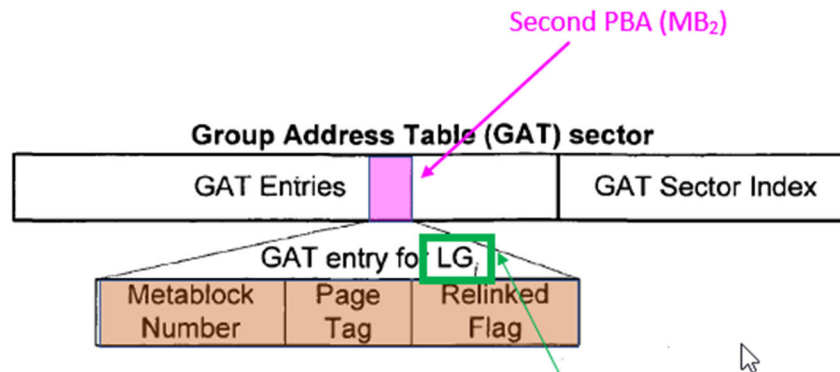


FIG. 17A



FIG. 3A

308. For at least these reasons, it is my opinion that Bennett discloses “wherein said first PBA includes a first set of access parameters and said second PBA includes a second set of access parameters, and at least one difference between said first and second PBAs.”

- f. **[Claim 32e]: “and wherein said mapping table increases the likelihood that that the operational load imposed on the storage apparatus during the processing of the I/O transaction request will be optimally distributed across storage device resources.”**

309. It is my opinion that Bennett discloses this claim element. Bennett

discloses that “[i]n order to maximize programming speed and erase speed, ***parallelism*** is exploited as much as possible by arranging for multiple pages of information . . . to be ***programmed in parallel***, and for multiple MEUs [minimum erasable units] to be erased in parallel.” Ex-1007 at 8:26-30;¹⁶ *see also id.* at 2:57-59 (“In order to improve read and program performance, multiple charge storage elements or memory transistors in an array are read or programmed in ***parallel***.”) (all emphasis added). Thus, the mapping table disclosed in Bennett provides a more “efficient use of system resources” by allowing “multiple logical groups to be updated concurrently,” which “increases efficiency and reduces overhead.” *Id.* at 4:50-52.

310. Bennett further discloses that the mapping table optimizes memory operations in response to processing of an I/O transaction with a host. For example,

¹⁶ As explained above, the ’740 patent states that parallel memory operations are one example of “optimized memory operations.” *Supra* n. 3. Likewise, the district court has held that examples of optimization “include[e] contiguous LBA sets, adjacent LBA sets, [and] adjacent PBA’s differing by at least one access parameter.” Ex-1008 at 30. Bennett teaches each of these examples. *Supra* § X.B.9.b, X.B.9.c, X.B.9.e.

Bennett discloses that the host “accesses the memory” and reads or writes “to the memory system in unit[s] of logical clusters, each consisting of one or more logical sectors.” *Id.* at 6:53-59; *see also id.* at 6:61-65 (explaining that the host “issues a command to the memory system 20 to read or write a segment containing a string of logical sectors of data with contiguous addresses”). As a result, the system disclosed by Bennett “maps the logical address from the host to a physical memory location.” *Id.* at 9:20-22.

311. For at least these reasons, it is my opinion that Bennett discloses a mapping table that “increases the likelihood that that the operational load imposed on the storage apparatus during the processing of the I/O transaction request will be optimally distributed across storage device resources.”

10. Claim 34 Is Rendered Obvious by Bennett in View of Bruce

312. It is my opinion that Bennett in combination with Bruce renders obvious every element of claim 34.

- a. **[Claim 34a]: “The electronic storage device of claim 32, wherein the processing of the I/O transaction request includes a first memory operation performed using said first PBA and a second memory operation performed using said second PBA;”**

313. It is my opinion that the combination of Bennett and Bruce teach all of the elements of claim 32. *See supra* § X.B.9. It is also my opinion that the

combination of Bennett and Bruce teach all of the elements of claim 34.

314. As explained above, Bruce discloses uniquely addressable memory locations (or PBAs) within different memory banks, which are associated with different FDEs, different buses, and different device addresses. *See supra* §§ X.B.3, X.B.4, X.B.9.e. Accordingly, the banks of flash-memory chips “can be separately accessed, allowing many flash operations to be performed in parallel.” Ex-1006 at 5:21-23; *see also id.* at 6:55-57 (noting that “four flash-memory chips [can] be accessed at once for each flash bus, for a total of eight flash accesses”); 10:17-19 (“The second cycle of the new request phase sends a device address for one of the flash-memory chips connected to the flash buffer chips.”). This simultaneous or parallel access to different unique memory locations (PBAs) illustrates a first memory operation using a first PBA and second memory operation using a second PBA.

315. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce’s bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious the “electronic storage device of claim 32, wherein the processing of the I/O transaction request

includes a first memory operation performed using said first PBA and a second memory operation performed using said second PBA.”

- b. **[Claim 34b]: “said first and second PBAs respectively associated with a first set of access parameters and a second set of access parameters, and said set of first and second set of access parameters differing by at least one access parameter.”**

316. It is my opinion that Bruce discloses this claim element. As discussed above, Bruce discloses uniquely addressable memory locations (or PBAs) within different memory banks, which are associated with different access parameters, including different FDEs, different buses, and different device addresses. *See supra* §§ X.B.3, X.B.4, X.B.9.e. Thus, for example, a first and second PBA can be identified using the following access parameters:

	Access Parameters		
	Bus Identifier	FDE Identifier	Group Identifier
First unique physical location (PBA)	Bus 10/Bus-A	Bank0 SM	Specific address of first chip
Second unique physical location (PBA)	Bus 18/Bus-B	Bank3 SM	Specific address of second chip

317. As a result, Bruce teaches that each of the PBAs is associated with a set of access parameters, in which the first set of access parameters (bus identifier, FDE

identifier, and group identifier) differ by at least one access parameter. Indeed, as illustrated above, Bruce demonstrates that each of the access parameters for a PBA may differ.

318. As discussed above, it is my opinion that it would have been obvious for a person of ordinary skill in the art in to combine Bennett with Bruce, and to include Bruce's bus identifiers, FDE identifiers, and group identifiers in the mapping table disclosed by Bennett. *See supra* ¶¶ 230-239. For at least these reasons, it is my opinion that Bennett in combination with Bruce renders obvious electronic storage device of claim 32 wherein "said first and second PBAs respectively associated with a first set of access parameters and a second set of access parameters, and said set of first and second set of access parameters differing by at least one access parameter."

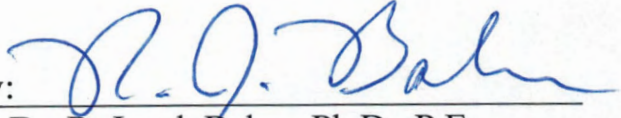
XI. CONCLUSION

319. For the foregoing reasons, it is my opinion that claims 1, 9-15, 32, and 34 of the '740 patent are anticipated or rendered obvious by Sukegawa, Bennett, and Bruce.

Declaration of R. Jacob Baker
Inter Partes Review of U.S. Patent No. 8,010,740

320. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: March 22, 2023

By: 
Dr. R. Jacob Baker, Ph.D., P.E