

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SONY INTERACTIVE ENTERTAINMENT LLC,
Petitioner,

v.

BOT M8, LLC,
Patent Owner.

IPR2020-00922
Patent 8,078,540 B2

Before, KALYAN K. DESHPANDE, JAMES A. TARTAL, and
AMBER L. HAGY, *Administrative Patent Judges*.

TARTAL, *Administrative Patent Judge*.

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

We have jurisdiction to conduct this *inter partes* review under 35 U.S.C. § 6. This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a) (2018) and 37 C.F.R. § 42.73 (2019). For the reasons discussed below, we determine Sony Interactive Entertainment LLC (“Petitioner”)¹ has shown by a preponderance of the evidence that claims 1–6 (“the Challenged Claims”) of U.S. Patent No. 8,078,540 B2 (Ex. 1001, “the ’540 patent”) are unpatentable.

I. INTRODUCTION

A. Summary of Procedural History

Petitioner filed a Petition pursuant to 35 U.S.C. §§ 311–319 requesting an *inter partes* review of the Challenged Claims. Paper 2 (“Pet.”). We instituted an *inter partes* review of the Challenged Claims on all grounds of unpatentability asserted in the Petition. Paper 9 (“Inst. Dec.”). Bot M8, LLC (“Patent Owner”)² filed a Patent Owner Response. Paper 13 (“PO Resp.”). Petitioner filed a Reply to the Patent Owner Response. Paper 16 (“Pet. Reply”). Patent Owner filed a Sur-reply. Paper 17 (“PO Sur-reply”).

Oral argument was held and a transcript of the hearing appears in the record. Paper 25 (“Tr.”). Petitioner bears the burden of proving unpatentability of each claim challenged by a preponderance of the evidence, and the burden of persuasion never shifts to Patent Owner. *See* 35 U.S.C. § 316(e); 37 C.F.R. § 42.1(d); *Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015).

¹ Petitioner identifies itself and Sony Corporation, Sony Corporation of America, and Sony Interactive Entertainment, Inc., as real parties in interest. Pet. 70.

² Patent Owner identifies no additional real parties in interest. Paper 4, 1.

B. Related Proceedings

The parties identify *BOT M8, LLC v. Sony Corporation of America*, 3:19-cv-07027 (NDCA) as a related proceeding. Petitioner explains that the '540 patent was a subject of that case, but that it “has been dismissed due to failure to state of [sic] claim of infringement.” Pet. 70; Paper 4, 1. The parties identify no additional related matters. *See id.*

C. The '540 Patent

The '540 patent, titled “Gaming Machine, Gaming Information Authentication and Acquisition Device, and Gaming Information Acquisition Device,” issued December 13, 2011, from an application filed on February 16, 2010, which was a continuation of an application filed on April 17, 2006, that issued as U.S. Patent No. 7,693,282 (“the '282 patent”). Ex. 1001, codes (21), (22), (45), (54), (63). The '540 patent is directed to “a gaming machine, a gaming information authenticating and loading device, and a loading device for loading gaming information, whereby gaming information recorded on a portable storage medium can be authenticated.” *Id.* at 1:18–22.

The '540 patent explains, as background, that in “gaming machines” a “gaming program” is supplied “by means of a storage medium,” such as a “compact flash memory,” and is used to control “gaming information” consisting of the operations required for playing the game. *Id.* at 1:23–42. In prior systems, according to the '540 patent, the storage medium is detachable and at risk of “illegal actions, such as duplicating or manipulating the gaming information stored on the medium.” *Id.* at 1:43–48. The '540 patent also states that prior systems authenticated only the storage medium, not “the actual gaming information,” which may have been manipulated in some way. *Id.* at 2:14–20.

The '540 patent purports to resolve the issues identified with the prior systems by describing a system “whereby the gaming information stored on a storage medium can be authenticated.” *Id.* at 2:37–42; *see also* Pet. 1 (stating that according to the '540 patent “the prior art only authenticated the medium on which the game program was stored, not the game program itself,” and purports to solve this problem “by authenticating the game program, not just the storage medium, thereby preventing game-program modification”).

Figure 1 of the '540 patent is reproduced below.

Fig.1

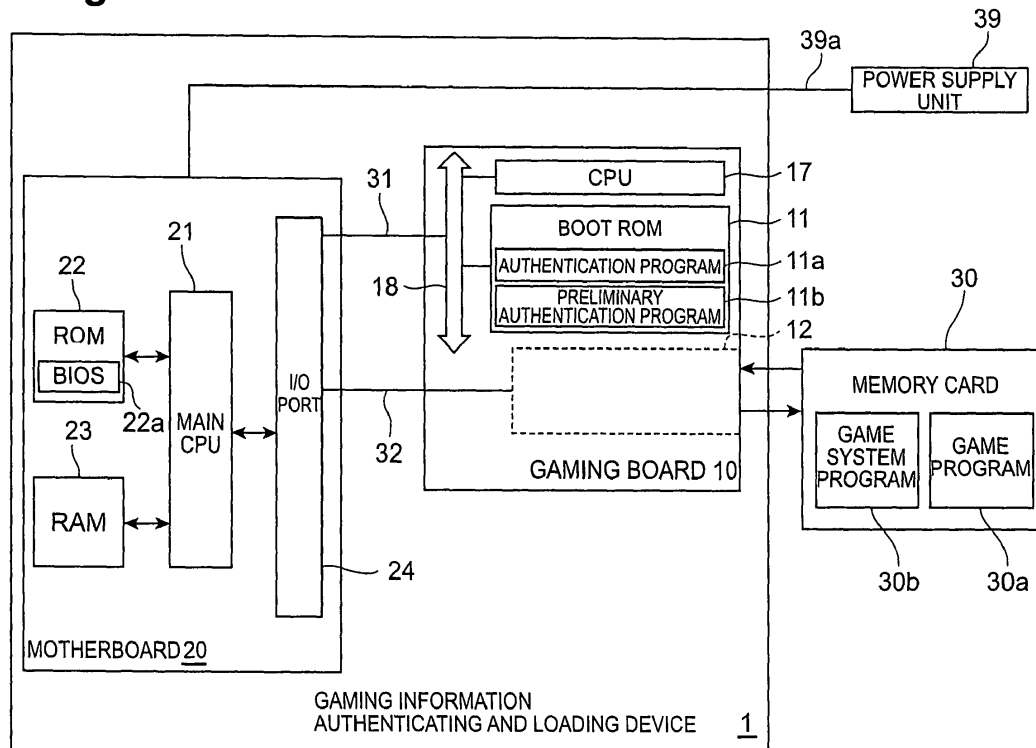


Figure 1 is “a block diagram showing the general constitution of a gaming information authenticating and loading device 1,” including gaming board 10 (a “loading device”) and motherboard 20 (a “processing device”). *Id.* at 5:25–30. Game program 30a and game system program 30b “form the

gaming information . . . and are stored on memory card 30.” *Id.* at 5:34–39. Gaming board 10, with card slot 12, CPU 17, and boot ROM 11, “loads a game program 30a and a game system program 30b . . . in the motherboard 20 from a memory card 30.” *Id.* at 5:40–46. PCI bus 31 transmits signals between motherboard 20 and gaming board 10. *Id.* at 5:49–52. “[C]ard slot 12 is connected to the motherboard 20 by means of an IDE bus 32.” *Id.* at 6:18–19.

“The boot ROM 11 stores an authentication program (first authentication program) 11a, a preliminary authentication program (second authentication program) 11b, . . . and a program (boot code) . . . for booting (starting up) the CPU 17 and the preliminary authentication program 11b.” Ex. 1001, 5:52–58. Authentication program 11a “states a procedure (authentication procedure) for authenticating the gaming information, in other words, checking and verifying that the gaming information, which is the object of the authenticating and loading process, has not been manipulated.” *Id.* at 61–65. “[P]reliminary authentication program 11b states a procedure (authentication procedure) for authenticating the authentication program 11a” to verify “that the authentication program 11a has not been manipulated.” *Id.* at 6:8–13.

Motherboard 20 is a “generic motherboard” and includes CPU (Central Processing Unit) 21, ROM (Read Only Memory) 22, RAM (Random Access Memory) 23, and I/O port 24 connected to PCI bus 31 and IDE bus 32. *Id.* at 6:25–32; *see also id.* at 11:12–17 (stating that “motherboard 20 is constituted by a commercially available generic motherboard, the motherboard 20 has highly generic characteristics and, consequently, it is possible to reduce manufacturing costs”). ROM 22 stores programs, such as BIOS 22a (standard BIOS on the motherboard 20), which

is executed by main CPU 21, and permanent data. *Id.* at 6:33–36. When BIOS (Basic Input/Output System) 22a, stored on ROM 22, is executed by CPU 21, “then prescribed initialization processing of the peripheral device is carried out, and a process for reading the game program 30a and the game system program 30b stored in the memory card 30 via the gaming board 10 is started.” *Id.* at 6:36–40. “RAM 23 can store, at the least, an authentication program 11a read out via the gaming board 10, and the game program 30a and the game system program 30b.” *Id.* at 6:45–48.

Petitioner summarizes the operation described by the ’540 patent as “authenticating the game program . . . by providing an ‘authentication program’ on a first circuit board with the game program and a processor on the motherboard that executes the authentication program to verify that the game program has not been modified.” Pet. 1 (citing Ex. 1001, 2:37–42, 5:40–6:43); *see also* PO Resp. 18 (describing the ’540 patent as providing, separate from motherboard 20, a gaming board 10 that “stores the gaming information until it can be verified and transferred to the motherboard, thereby adding a layer of protection between the gaming information and the motherboard”) (citing Ex. 1001, 10:1–11, 10:49–62), 19–22 (describing the “unique configuration by which the system has an ability to check the status of a board having a memory before any operating-system applications are permitted to access the memory” purportedly disclosed by the ’540 patent) (citations omitted), 22–25 (describing the “strong authentication sequence via the unique interfaces” purportedly disclosed by the ’540 patent).

D. Illustrative Claim of the ’540 Patent

Petitioner challenges claims 1–6 of the ’540 patent. Pet. 1. Claims 1 and 4 are independent, with claim 1 reciting “[a] gaming machine” and claim 4 reciting “[a] method of controlling a gaming machine.” Ex. 1001,

12:64–13:25, 13:47–14:24. Claims 2 and 3 depend from claim 1. Claims 5 and 6 depend from claim 4. *Id.* at 13:26–46, 14:25–49. Claim 1 is illustrative of the claimed subject matter and is reproduced below.

1. A gaming machine, comprising:
 - (i) a board including a memory in which a game program for executing a game and an authentication program for authenticating the game program are stored;
 - (ii) a motherboard which is different from the board and connects to the board, the motherboard including another memory which is different from the memory, said another memory configured to read out and store the game program stored in the memory; and
 - (iii) a CPU which is provided on the motherboard, for executing the game based upon the game program stored in said another memory, the CPU being configured to:
 - (a) read out the authentication program from the memory of the board, and then, store the read out authentication program in said another memory of the motherboard;
 - (b) execute the authentication program stored in said another memory in the process (a), and then, authenticate the game program in the memory of the board, based upon the executed authentication program;
 - (c) write the game program in the memory of the board, to said another memory of the motherboard, in a case where the game program in the memory of the board is authenticated as a result of the authentication process (b); and
 - (d) execute the game based upon the game program written to said another memory of the motherboard in the process (c).

Id. at 12:64–13:25.

E. References and Testimony

Below we provide an abbreviated summary of the qualifications of Dr. Andrew Wolfe, who provides testimony in support of Petitioner, and Dr. Long Yang, who provides testimony in support of Patent Owner. We also provide a table identifying the primary references relied upon by Petitioner, as well as the exhibits corresponding to the declarations and deposition testimony in the record for Dr. Wolfe and Dr. Yang.³

Dr. Wolfe received a B.S. degree in Electrical Engineering and Computer Science from the Johns Hopkins University, as well as a M.S. degree in Electrical and Computer Engineering and a Ph.D. degree in Computer Engineering from Carnegie Mellon University. Ex. 1003 ¶ 6. He was as Assistant Professor of Electrical Engineering at Princeton University from 1991 through 1997, where he taught “undergraduate and graduate-level courses in Computer Architecture, Advanced Computer Architecture, Display Technology, and Microprocessor Systems courses, as well as conducting sponsored research in the area of computer systems and related topics.” *Id.* ¶ 12. Dr. Wolfe states that he has “published more than 50 peer-reviewed papers in computer architecture and computer systems design,” has chaired conferences in microarchitecture and integrated circuit design, and is “a named inventor on at least fifty-six U.S. patents and thirty-seven foreign patents.” *Id.* ¶ 14.

Dr. Wolfe details various positions he has held in private industry. *Id.* ¶¶ 5, 7, 9–11, 13; *see also* Ex. 1004 (the curriculum vitae of Dr. Wolfe identifying prior employment). Dr. Wolfe states that he is currently a

³ The table provided identifies only a select number of documents. A complete identification of the papers and exhibits that form the record of this case is available in the docket of this proceeding.

consultant who provides “technical and business analysis to businesses on processor technology, computer systems, display technology, consumer electronics, software, design tools, data security, cryptography, and intellectual property issues,” as well as “a lecturer at Santa Clara University, teaching courses on Microprocessor Systems, Real-time Embedded Systems, and Mechatronics in the Electrical and Computer Engineering, Computer Science and Engineering, and Mechanical Engineering departments.” *Id.* ¶¶ 5, 12.

Dr. Yang received a Bachelor of Science degree in Electrical Engineering from National Taiwan University, as well as a Master of Science and a Doctorate Degree in Electrical and Computer Engineering from University of California, Santa Barbara. Ex. 2041 ¶ 3. Dr. Yang states that he “was Vice President of Engineering or Vice President of Research Development for many high-tech companies in Silicon Valley, including Toshiba America Electronic Components,” where he “led teams of scientists and engineers to develop new technologies/products in the field of optoelectronics, displays, telecommunication, and semiconductors.” *Id.* ¶ 4, App. A (curriculum vita of Dr. Yang identifying prior employment). Dr. Yang has “authored and co-authored over 46 publications in a large variety of technical fields, including semiconductor, laser, display, LEDs, epitaxy, etc.,” “was invited to give presentations in renowned international conferences,” and is a named inventor on 43 U.S. Patents “generally related to semiconductor laser, fiber optics, optical switches, visible LEDs, high-definition TV, and high voltage transistors.” *Id.* ¶ 5. Dr. Yang states that he is a registered patent agent and that he is currently employed as a scientific advisor at Kramer Levin Naftalis & Frankel LLP, the law firm representing Patent Owner in this proceeding. *Id.* ¶¶ 4, 7.

References and Witness Testimony	Date	Ex. No.
U.S. Patent No. 6,565,443 B1 (“Johnson”)	May 20, 2003	Ex. 1005
US 2003/0130032 A1 (“Martinek”)	July 10, 2003	Ex. 1006
US 2006/0101310 A1 (“Diamant”)	May 11, 2006	Ex. 1007
US 2004/0054952 A1 (“Morrow ’952”)	March 18, 2004	Ex. 1009
US 2003/0064771 A1 (“Morrow ’771”)	April 3, 2003	Ex. 1010
Declaration of Andrew Wolfe, PhD.	May 15, 2020	Ex. 1003
Deposition Transcript of Dr. Wolfe	Feb. 3, 2021	Ex. 2014
Declaration of Dr. Long Yang	Feb. 16, 2021	Ex. 2041
Deposition Transcript of Dr. Yang	May 4, 2021	Ex. 1041

F. Asserted Grounds of Unpatentability

Petitioner asserts that the Challenged Claims are unpatentable based on the following grounds:

Claims Challenged	35 U.S.C. §	References/Basis
1, 4	103(a)	Johnson, Martinek
2, 3, 5, 6	103(a)	Johnson, Martinek, Diamant
1, 4	103(a)	Morrow ’952, Morrow ’771, Diamant

Pet. 2–3.⁴ More specifically, Petitioner contends, for example, that claims 1 and 4 are obvious over Johnson “in view of” Martinek. *Id.* In its analysis, however, Petitioner makes clear that it relies on Martinek “[t]o the extent additional disclosure is required.” *See, e.g., id.* at 9. Similarly, for

⁴ The Leahy-Smith America Invents Act (“AIA”) included revisions to 35 U.S.C. §§ 102, 103 that became effective on March 16, 2013. Pub. L. No. 112-29, §§ 3(b)–3(c), 3(n)(1), 125 Stat. 284, 285–87, 293 (2011). Because there is no dispute that the ’540 patent claims have an effective filing date before March 16, 2013, we apply the pre-AIA versions of these statutes.

obviousness based on Morrow '952, Petitioner includes discussion of Morrow '771 and Diamant “[t]o the extent additional disclosure is needed,” or as an “alternative” or “addition” to what Morrow '952 suggests. *See, e.g., id.* at 57, 58, 67. Accordingly, our analysis of the grounds below includes consideration of whether Petitioner has shown obviousness based on Johnson, alone, or based on Morrow '952, alone.

II. ANALYSIS OF PATENTABILITY

A. Principles of Law

Petitioner contends under three grounds that claims of the '540 patent are unpatentable based on obviousness. Pet. 2–3. As set forth in 35 U.S.C. § 103(a),

[a] patent may not be obtained . . . if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.

The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of ordinary skill in the art; and (4) when in evidence, objective evidence of nonobviousness.⁵ *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966).

An obviousness analysis “need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007); *accord In re Translogic Tech., Inc.*, 504 F.3d 1249, 1259 (Fed. Cir. 2007).

⁵ Neither party addressed objective evidence of nonobviousness.

However, Petitioner cannot satisfy its burden of proving obviousness by employing “mere conclusory statements.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016). Instead, Petitioner must articulate a reason why a person of ordinary skill in the art would have combined the prior art references. *In re NuVasive*, 842 F.3d 1376, 1382 (Fed. Cir. 2016).

B. Level of Ordinary Skill in the Art

In determining the level of ordinary skill in the art, various factors may be considered, including the “type of problems encountered in the art; prior art solutions to those problems; rapidity with which innovations are made; sophistication of the technology; and educational level of active workers in the field.” *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995) (citation omitted). Petitioner contends that a person of ordinary skill in the art at the time of the invention would have had “the equivalent of at least an undergraduate degree in computer science, computer engineering, electrical engineering, or a similar technical field, and with one or more years of experience in the field of authentication, verification, and/or error detection in the context of computer hardware and/or software,” where “[a]dditional education may substitute for less experience and vice versa.” Pet. 2 (citing Ex. 1003 ¶¶ 50–52).

Patent Owner argues that Petitioner’s proposed level of ordinary skill is “incorrect because it fails to account for the requisite technical knowledge of computer system design.” PO Resp. 15 (citing Ex. 2041 ¶ 77).

According to Patent Owner, a person of ordinary skill would have had “the equivalent of at least an undergraduate degree in computer engineering, electrical engineering, or a similar technical field, and with three to five years of experience in the field of computer system designs, and also [would

have had] knowledge in authentication, verification, and/or error detection in the context of computer hardware and/or software.” *Id.* (citing Ex. 1041 ¶ 76). Patent Owner further states that a person of ordinary skill requires one who would have had “knowledge far beyond the curriculum of a typical undergraduate computer science program,” including “a clear understanding of computer system design and, in particular, how to interface with a daughterboard through a PCI interface.” *Id.* at 16 (citing Ex. 1041 ¶ 78). In short, Patent Owner proposes a higher level of ordinary skill than Petitioner.

During oral argument, Petitioner indicated that any difference between the levels of ordinary skill proposed by the parties was not material to their dispute. Tr. 16:13–17:9. Likewise, Patent Owner further confirmed that any disagreement with Petitioner’s proposed level of ordinary skill had “no material impact.” *Id.* at 42:12–22.

We apply Petitioner’s definition of the level of ordinary skill in the art. We determine the definition offered by Petitioner is consistent with the teachings of the ’540 patent and the prior art of record. *Cf. Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001) (noting that the prior art itself may reflect an appropriate level of skill in the art). Neither party contends the differences in their competing proposals are material to the issues before us and our adoption of Petitioner’s proposed definition does not reflect a view that adopting Patent Owner’s competing definition of the level of ordinary skill in the art would have any impact on the outcome of this proceeding. To the contrary, our conclusions would be the same under either party’s definition.

C. Claim Construction

“In an *inter partes* review proceeding, a claim of a patent . . . shall be construed using the same claim construction standard that would be used to

construe the claim in a civil action under 35 U.S.C. 282(b).” 37 C.F.R. § 42.100(b) (2019). That standard “includ[es] construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.” *Id.*; *see also Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc).

Petitioner states that the “ordinary and customary meaning,” as understood by a person of ordinary skill in the art at the time of the invention, applies in this case and identifies in the Petition no claim term that requires express construction. Pet. 3. Patent Owner specifically addresses “read” and “configured to” in the claim construction portion of its Patent Owner Response, which we discuss below and conclude do not require express constructions.

1. “read”

Claim 1, for example, recites “memory configured to read out and store the game program,” as well as “the CPU being configured to: (a) read out the authentication program from the memory of the board, and then, store the read out.” Ex. 1001, 12:64–13:25. The Challenged Claims do not use the term “load,” but the Specification discusses at length “authenticating and loading device 1” (*see, e.g., id.* at 5:25–39), including that “gaming board 10 is a device which loads a game program 30a and a game system program 30b . . . in the motherboard 20” (*id.* at 5:42–46). Petitioner states that a person of ordinary skill in the art “would have recognized that the disclosed loading processes are reading out of one memory and storing in another.” Pet. 21 (citing Ex. 1003 ¶¶ 46–48, 159–162).

According to Patent Owner, “read” and “load” have “different meanings in the context” of the ’540 patent. PO Resp. 26; *see also* PO Sur-

reply 1–4. Patent Owner directs us to the definitions provided by Dr. Wolfe on behalf Petitioner from the Microsoft Computer Dictionary. In his declaration, Dr. Wolfe explains as follows:

[T]he 2002 edition of the *Microsoft Computer Dictionary* defines the term “load” as follows:

load *vb.* To place information from storage into memory for processing, if it is data, or for execution, if it is program code.

Microsoft Computer Dictionary (Ex. 1011) at 315. The *Microsoft Computer Dictionary* similarly defines the term “read” as follows:

read *vb.* To transfer data from an external source, such as from disk or the keyboard, into memory or from memory into the Central Processing Unit (CPU).

Microsoft Computer Dictionary (Ex. 1011) at 440.

Ex. 1003 ¶ 46. Patent Owner purports to reproduce these definitions in its Response, but includes without identification or explanation modifications to the definition of “read” to show a division between parts Patent Owner labeled (i) and (ii) as follows:

read *vb.* To transfer data (i) from an external source, such as from disk or the keyboard, into memory or (ii) from memory into the Central Processing Unit (CPU).

PO Resp. 26 (misquoting Ex. 1003 ¶ 46 and Ex. 1011, 440).⁶ Patent Owner proceeds to argue that the ’540 patent uses “read” only “with the read (ii)

⁶ Dr. Yang suggests that “[o]riginally, Microsoft Computer Dictionary’s definition of ‘read’ had ‘(i)’ and ‘(ii)’ inserted in its definition of ‘read,’ which Petitioner apparently omitted.” Ex. 2041 ¶ 110. Dr. Yang does not identify any “original” version of the Microsoft Computer Dictionary to support his assertion and offers no explanation for why he suggests that Petitioner “apparently omitted” some portion of the definition where the definition provided by Dr. Wolfe is *verbatim* what appears in the exhibit Dr. Wolfe cites as support. Ex. 1003 ¶ 46 (quoting Ex. 1011, 440).

definition, . . . which is to transfer data from memory into the central processing unit.” *Id.* (emphasis omitted) (citing Ex. 2041 ¶ 113).

Petitioner argues in response, and we agree, that “Patent Owner never cites to any intrinsic evidence from the patent to support this construction.” Pet. Reply 2. Counsel for Petitioner further stated during oral argument that Petitioner does not believe there is any material dispute between the parties over the meaning of “read,” because Patent Owner did not argue “that the prior art fails to teach ‘read’ as construed.” Tr. 13:2–14:5.

Subsequently, during oral argument, counsel for Patent Owner explained that Patent Owner’s “concern in reference to the definition of ‘read,’ . . . has been really related to the fact that the gaming information cannot be read into RAM until after the authentication is complete,” and that “[a]pparently, that’s no longer in dispute.” *Id.* at 43:20–44:2; *see also id.* at 37:18–23 (counsel for Petitioner confirming that Petitioner has not disputed Patent Owner’s position that the Challenged Claims “preclude writing the game program into RAM prior to authentication”). The issue was further clarified during the oral argument in the following exchange:

JUDGE HAGY: Is there really a material dispute between the parties over the construction of “read”? In other words, does patent owner dispute that the prior art teaches transferring data from memory into the CPU, as is on your Slide 10?

MR. FRANKEL: Well, the prior art shows that at some point in the process gaming information is put into the -- from memory into the CPU, but the dispute is, when it goes into the memory of the motherboard -- which, again, now the parties agree is required by the claim, so I don't think there's a dispute before the Board anymore on the construction of “read.”

Id. at 45:3–14. Accordingly, we find on the full record that there is no material dispute between the parties with regard to the meaning of “read,” and, further, that an express construction of “read” is not necessary.

See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co., 868 F.3d 1013, 1017 (Fed. Cir. 2017); *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) (“[O]nly those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy.”).

2. “*configured to*”

Claim 1 recites, for example, “the CPU being configured to.” Ex. 1001, 12:64–13:25. Patent Owner argues that “[t]he Federal Circuit has long construed the claim term ‘configured to’ to generally mean ‘adapted to,’ ‘made to,’ or ‘designed to’ perform the stated function(s),” and that in this proceeding it should be construed to mean “requiring structure designed to or configured to accomplish the specified objective, not simply that they can be made to serve that purpose.” PO Resp. 27 (citations omitted); *see also* PO Sur-reply 5–8. Petitioner argues that “[t]he use of the term ‘configured to’ is not an invitation to read in limitations from the specification to require the specific hardware configuration described in exemplary embodiments in the specification,” and that “if the Board construes ‘configured to’ at all, it should confirm that ‘configured to’ does not limit the claim to only the specific structure disclosed in the ’540 Patent specification.” Pet. Reply 4–5. Further, according to Petitioner, “[b]eyond that issue, the meaning of the term ‘configured to’ is not otherwise in dispute between the parties and further construction is not necessary.” *Id.* at 5.

During oral argument, as with the claim term “read” discussed above, Patent Owner similarly explained that “[t]he argument was the CPU has to be configured such that it will not be pulling gaming information into RAM until after authentication is complete, which is not in dispute.” Tr. 43:3–44:2. Patent Owner further confirmed that the use of “configured to” in

independent claim 1 of the '540 patent was not a distinguishing characteristic for purposes of our patentability analysis over independent claim 4 of the '540 patent, which does not recite “configured to.”

Accordingly, we find on the full record that there is no material dispute between the parties with regard to the meaning of “configured to,” and, further, that an express construction of “configured to” is not necessary for purposes of this Decision.

3. Other Claim Terms

We find no other claim term requires an express construction for purposes of this Decision.

D. Scope and Content of the Prior Art

Petitioner relies on Johnson, Martinek, Diamant, Morrow '952, and Morrow '771, each of which we briefly summarize in relevant part below.

1. Summary of Johnson

Johnson, titled “System and Method for Verifying the Contents of a Mass Storage Device Before Granting Access to Computer Readable Data Stored on the Device,” states that it “relates to a general system and method for verifying the contents of a mass storage device before granting access to computer readable data stored on the device.” Ex. 1005, code (54), 1:9–12. More particularly, Johnson states that it “relates to a method and apparatus verifying the contents of a read-only mass storage device as part of a booting process of a computer system that is part of a computer controlled gaming machines before granting access to gaming application modules and data stored on the device.” *Id.* at 1:12–17. Johnson explains that “a procedure is needed to verify the approved media that is inspected by the regulators is used each time a computer controlled gaming machine operates.” *Id.* at 4:36–38.

Johnson summarizes one embodiment as follows:

One such aspect of the present invention is a method for verifying the contents of a mass storage device attached to a computing system having a processor and system random access memory. The mass storage device has computer readable data stored therein. The method comprises first storing onto the mass storage device a mass storage verification module having a set of computer executable instructions for use in verifying the contents of the mass storage device. Next, the method stores a mass storage verification data for use in verifying the contents of the mass storage device onto the mass storage device. Before permitting an initial access to the computer readable data from the mass storage device, the method verifies the contents of the mass storage device by executing the instructions contained within the mass storage verification module to generate a set of computed verification data and comparing the set of computed verification data with the mass storage verification data. Finally, the method permits additional access to the computer readable data stored on the mass storage device only if the set of computed verification data matches the mass storage verification data.

Id. at 3:22–42.

Figure 1 of Johnson is reproduced below.

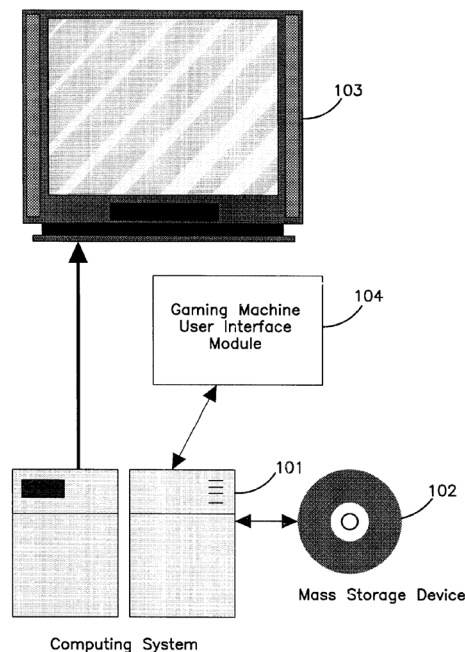


Figure 1 illustrates a computing system using a CD-ROM as a mass storage device, including “computing system 101 for use in storing application modules and data within a file system,” mass storage device 102, display device 103, and Gaming User Interface Module 104. *Id.* at 6:52–63. “The computing system 101 executes the application modules stored on the mass storage device 102 to generate images to be displayed on a display device 103, including but not limited to, a local video display, controlled by a Gaming Machine User Interface module 104, based upon instructions and data contained on the mass storage device 102.” *Id.*

Figure 2 of Johnson, reproduced below, shows computing system 101 “implemented using a computing system typically referred to as a personal computer.” *Id.* at 6:64–66.

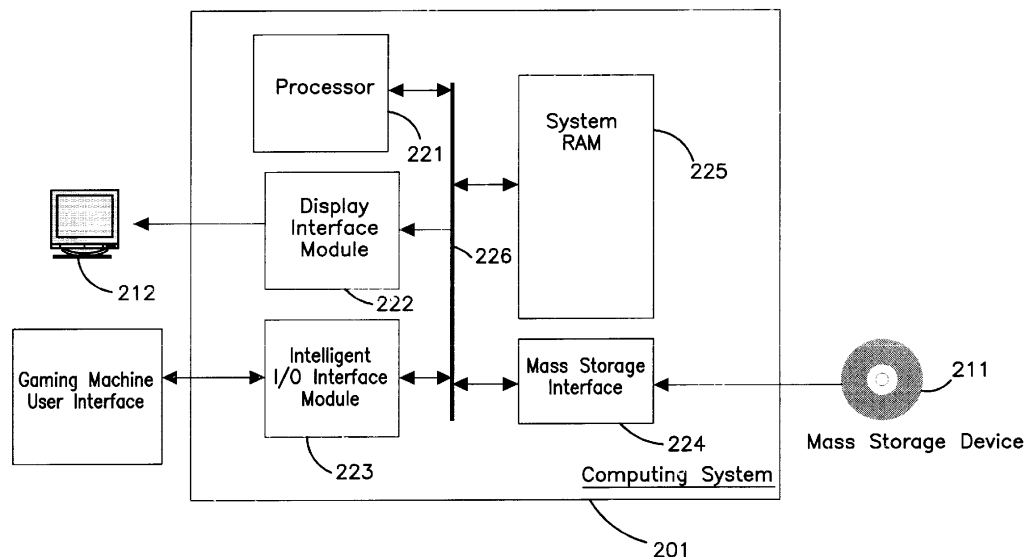


Figure 2 illustrates computing system 201 “used to implement a computer controlled gaming machine,” and includes main processing module 221, display interface module 222, intelligent input and output interface module (IIOB) 223, mass storage interface module 224, and system random access memory (RAM) 225. *Id.* at 7:2–9, 7:16–23. Also shown in Figure 1

are mass storage device 211 on which the operating system is stored and “gaming machine interface devices . . . such as a hopper, bill acceptor, meters, input panel, and the like,” which interface with computing system 201 through IIOB 223. *Id.* at 7:34–38.

Figure 3 of Johnson is reproduced below.

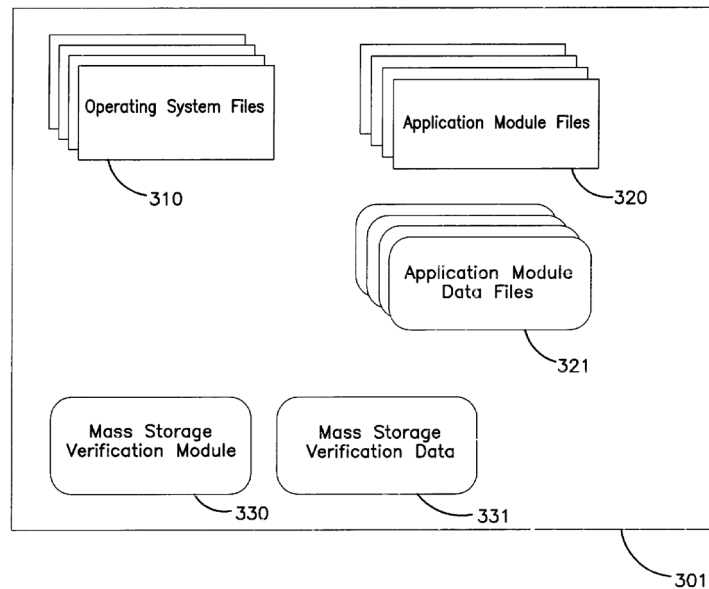


Figure 3 illustrates hierarchical file system 301 showing the contents of mass storage device 211, which organizes a plurality of different data modules and includes “one or more operating system files 310, one or more application module files 320, . . . one or more application module data files 321, . . . a mass storage verification module file 330 and a mass storage verification data file 331.” *Id.* at 7:39–50. Johnson describes hierarchical file system 301 as follows:

In the preferred embodiment, the application module file 320 and the application module data files 321 are the application program and data files used by the computing system 201 to generate the images, sounds, and user interactions needed to create a computer controlled gaming machine. The operating system files 310 include all of the files needed to implement an operating system on the computing system 201

that is needed to support the execution of the gaming application programs. The mass storage verification module 330 is an application program run before any data is permitted to be accessed by a gaming application to verify the contents of the mass storage device are valid. The mass storage verification data file 331 contains data utilized by the mass storage verification module 330 when performing its operations.

Id. at 7:51–65.

2. *Summary of Martinek*

Martinek, titled “Pass-Through Live Validation Device and Method,” relates to “computerized wagering game systems,” and “more specifically to use of a physical system for embedding a data verification device, component or a verification subcomponent in a gaming apparatus.”

Ex. 1006 ¶ 3. In one embodiment, Martinek describes an Integrated Device Electronics (IDE) system with “a Read Only Memory board that is pinned to plug into a hard drive port” that is “communicatively connected to a processing intelligence function (which may be a hard drive processor or other processor or microprocessor separate from the host computer).” *Id.*

¶¶ 29, 132. Figure 1 of Martinek is reproduced below.

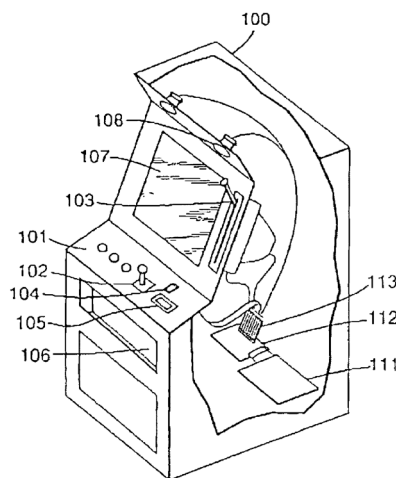


Fig. 1

Figure 1 illustrates gaming system 100 with computerized universal game controller 111 connected by wiring harness 113 to I/O interface 112. *Id.*

¶ 68. Universal game controller 111 may “control various gaming systems via I/O interface 112 designed to properly interface an input and/or output of the universal computerized game controller to the interface assemblies found within the various gaming systems.” *Id.* Petitioner notes that Martinek incorporates by reference Yoseloff⁷ “to provide exemplary game controller embodiments,” and states that in the game controller of Yoseloff “a ‘universal motherboard’ is used to replace a special-purpose motherboard.” Pet. 18; Ex. 1006 ¶ 68; Ex. 1008, Abstract, ¶¶ 30–33, 38.

Figure 14 of Martinek is reproduced below.

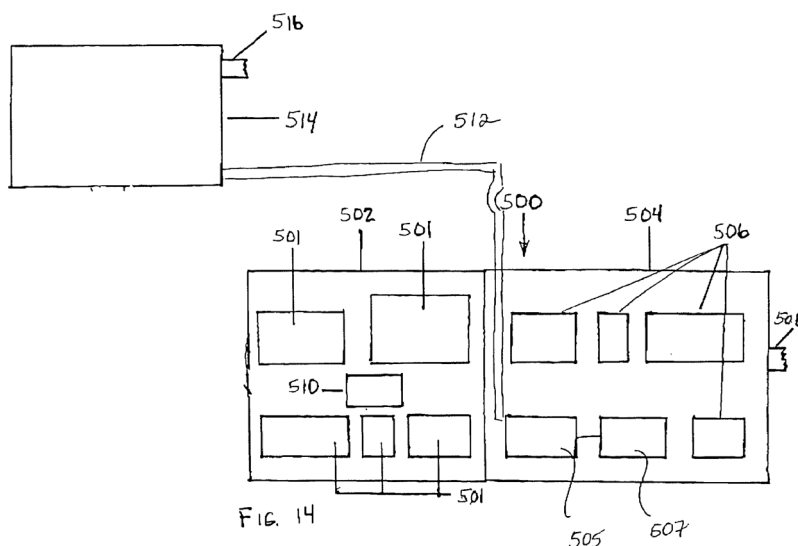


Figure 14 is a schematic of “third generation IVC [intelligent chip validation] system 500 [or IDE system 500] having the authentication program embedded outside of the game controller or computer.” Ex. 1006 ¶ 133. “IDE System 500 comprises a first board 502 having various

⁷ US 2001/0053712 A1, published December 20, 2001 (Ex. 1008, “Yoseloff”).

memory storage elements 501 (e.g., preferably non-writeable media such as ROM, EPROM, PROM and the like),” and “[a]nother board 504 which may be an extension or part of the first board 501,” with “its own processing intelligence.” *Id.* ¶¶ 133–134. “The processing intelligence is capable of authenticating data stored in memory elements 501 and 506, if present.” *Id.* ¶ 134. “A communication port 508 (I/O port with any communication link) carries information to and from the memory storage on the first and/or second board.” *Id.* ¶ 135. “Another communication link 512 to a host processor 514 with its own communication link 516 is shown in communicative connection with the second board 504 including intelligence 505 and associated memory elements 507.” *Id.*

3. Summary of Diamant

Diamant, titled “Device, System and Method for Verifying Integrity of Software Programs,” describes “a first verification algorithm or program to evaluate the integrity or authenticity of a second verification program.” Ex. 1007, codes (54), (57). Figure 1 of Diamant is reproduced below.

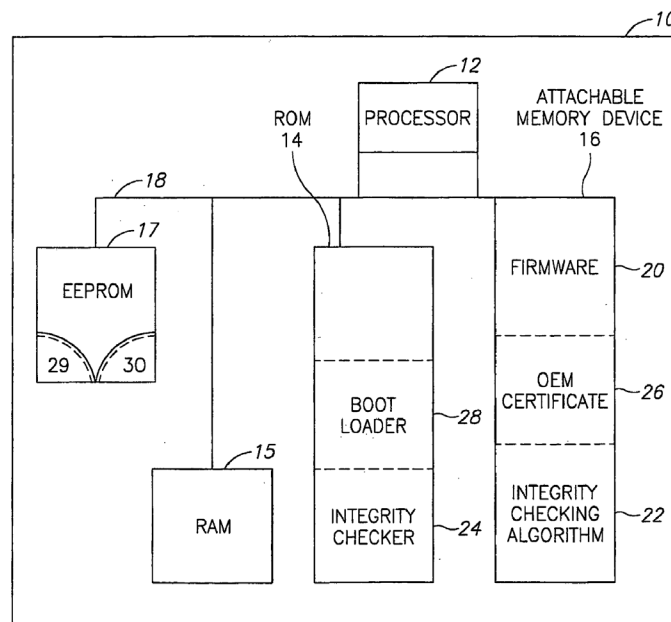


FIG.1

Figure 1 is a schematic of device 10, including processor 12, ROM 14 (a “data storage unit”), random access memory (RAM) 15, memory device 16 (such as “an attachable memory device or another suitable storage device”), electrical erasable read only memory (EEPROM) 17, and buses 18 connecting various components and memory units. *Id.* ¶ 10. “[M]emory device 16 may be included in a remote device, or may be integral to device 10, and need not be attachable.” *Id.* ROM 14 may store boot loader 128 as well as “one or more authenticity or integrity checking algorithms such as . . . integrity checker 24 that may . . . verify the integrity of an integrity checking algorithm 22 or other integrity checking functions that may be stored for example on a memory device 16 such as . . . an attachable memory device or in other data storage units that may be operably connected to device 10.” *Id.* ¶ 11. Memory device 16 may “include an integrity checking algorithm 22 or other device, code or testing mechanism that may check . . . the integrity of . . . firmware 20 . . . included in . . . memory device 16 to confirm that firmware 20 has not been miscopied, compromised, altered without authorization, corrupted or otherwise not authorized for use with device 10.” *Id.* ¶ 12.

EEPROM 17 may “store the expected results of integrity checker’s 24 evaluation of the integrity of an integrity checking algorithm 22 and other identification information for firmware 20 that may be used to confirm that such firmware 20 or attached memory device 16 is compatible with or authorized for use on device 10.” *Id.* ¶ 13. The results “generated by the execution of integrity checker 24 in its evaluation of integrity checking algorithm 22 may be checked against the expected result 30” stored in EEPROM 17 “to confirm the integrity or authorized status of integrity checking algorithm 22.” *Id.* ¶ 14. EEPROM 17 “may also store a device-

specific value 29 such as . . . a key, random number or other symbol that may . . . be used or accepted as an input in the execution of integrity checker 24.” *Id.*

Figure 2 of Diamant is reproduced below.

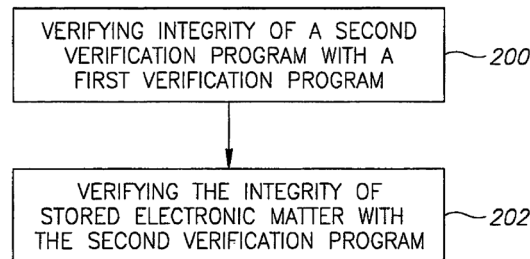


FIG.2

Figure 2 “is a flow diagram depicting a method of verifying the integrity of a program” in accordance with Diamant. *Id.* ¶ 24. In block 200, a first verification program stored on a memory unit of the device that is “separate from the memory unit that stores the second verification program and the program or firmware to be run on the device” verifies “the integrity of a second verification program.” *Id.* ¶ 24. The “expected result that is to be yielded by the first verification program’s check of the second verification program may be stored in a third memory unit [such as an EEPROM unit] that may be read by a processor or by a memory unit that stores the first verification program.” *Id.* ¶¶ 25–26. “If the first verification program confirms the integrity of the second verification program, the method may proceed to block 202.” *Id.* ¶ 27. “In block 202, the second verification program may verify the integrity of a functional program or other stored electronic matter that may have been downloaded into a memory of the device or that may have been included in . . . a memory unit of a device, such as . . . an attachable memory device.” *Id.* ¶ 28.

4. *Summary of Morrow '952*

Morrow '952, titled “Device Verification System and Method,” is directed to “verifying a device by verifying the components of that device,” such as “processors, persistent storage media, volatile storage media, random access memories, read-only memories (ROMs), erasable programmable ROMs, [and] data files.” Ex. 1009, code (54), ¶ 6. The device may be “a gaming machine, wherein the verification of the gaming machine is performed before game play is allowed.” *Id.* ¶ 19.

In an embodiment, device 10 includes “one or more processors 62, persistent storage media 80 and 90, volatile storage media such as random access memories (RAMs) 76, read-only memories (ROMs) 77 or electrically erasable programmable ROMs (EEPROMs) such as basic input/output systems (BIOS) 64.” *Id.* ¶ 37. Device 10 also includes data files 54, such as software program files 92–96 and operating system files 98. *Id.* Further, “[e]ither within the device 10, or in the diagnostic system 140 attachable to the device 10, are executable instructions or a software program 70 for verification of the components (verification software 70), which may itself be one of the components 50 to verify if it is internal to the device 10.” *Id.* ¶ 38. “Preferably, the verification software 70 is stored in a basic input/output system (BIOS) 64 device or chip.” *Id.* This “mak[es] it hard to bypass the verification process” “because the code in the BIOS 64 is usually the first code executed upon boot or start-up of the device 10.” *Id.* Thus, “operating system files 98 may be verified before loading or booting, or before any software program 92 is run from the persistent storage media 90. This makes the verification software 70 completely independent of data files 54 stored on the persistent storage media 90 which are being verified.” *Id.* ¶ 64

5. *Summary of Morrow '771*

Morrow '771, titled Reconfigurable Gaming Machine, is directed to “gaming machines having the ability to reconfigure entire games, pay tables and/or artwork.” Ex. 1010, code (54), ¶ 1. In an embodiment, a gaming platform 70 “enables casino owners to draw off of the large library of casino game functions available in a traditional master processing unit (MPU) stand-alone platform, while adding the graphics and sound capabilities of a personal computer.” *Id.* ¶ 29. Gaming platform 70 includes processor 90 that has a CD-ROM drive for storing “graphics, sound files, presentation software for at least one game, and [a] basic operating system.” *Id.* ¶ 31. Processor 90 also has a “customized BIOS chip, referred to as BIOS +, which provides typical PC boot functions, as well as verification and decryption algorithms.” *Id.* “The gaming platform 70 performs many verification processes during boot-up and game operation,” and “an algorithm that originates on the BIOS+ conducts verification of all files on the CD-ROM.” *Id.* ¶ 37.

E. Alleged Obviousness over Johnson, Alone, and in View of Martinek

Petitioner contends that the subject matter of independent claims 1 and 4 of the '540 patent would have been obvious over Johnson, alone, and in view of Martinek. Pet. 4–36; Pet. Reply 4–15. Petitioner’s contentions are supported by Dr. Wolfe. Ex. 1002 ¶¶ 123–195. Patent Owner disputes these contentions, primarily with regard to whether the asserted references teach or suggest “(1) the unique authentication process using unique hardware configurations without first storing the data in the memory on the motherboard, and (2) the unique hardware configurations that enable the board to halt the booting process of the motherboard,” and to whether Petitioner has shown a sufficient rationale in support of the asserted

combination. PO Resp. 39–50; PO Sur-reply 9–17; *see also* Ex. 2041 ¶¶ 145–181 (Dr. Yang’s testimony in support of Patent Owner).

We consider and address below the arguments and evidence of each party and determine for the reasons provided that Petitioner shows by a preponderance of the evidence that Johnson teaches or suggests every limitation of claims 1 and 4 of the ’540 patent. We further conclude that Petitioner has shown by a preponderance of the evidence that claims 1 and 4 are unpatentable over Johnson, and that, to the extent Petitioner relies on Martinek as further express support for elements Johnson suggests, but does not expressly disclose, Martinek is persuasive evidence of what a person of ordinary skill in the art would have known.

1. Differences Between the Subject Matter of Independent Claim 1 and the Teachings of Johnson, Alone and in View of Martinek

Petitioner provides a detailed explanation of how Johnson, alone, teaches or suggests each limitation of claim 1, with further support cited in Martinek, which we analyze below, along with Patent Owner’s arguments in opposition.

A gaming machine, comprising:

(i) a board including a memory in which a game program for executing a game and an authentication program for authenticating the game program are stored;

To the extent the preamble is limiting, Petitioner shows that Johnson teaches a “gaming machine.” Pet. 6 (citing, e.g., Ex. 1005, 6:45–48) (describing a “preferred embodiment” as “a computer controlled gaming machine”). With regard to the recited “board including a memory” that stores a game program and an authentication program, Petitioner shows that Johnson teaches a “mass storage device” that stores a “gaming application module,” corresponding to the recited “game program,” and a “verification

module” corresponding to the recited “authentication program.” *Id.* at 6–9 (citing, e.g., Ex. 1005, 7:39–65, Fig. 3). Petitioner acknowledges that the “mass storage device” of Johnson is not “explicitly” included on a “board.” *Id.* at 6. Petitioner shows that such an implementation would have been obvious to a person having ordinary skill in the art based on the teachings of Johnson, alone, as well as in combination with Martinek. Specifically, Johnson teaches that “the mass storage device may be ‘any mass storage device attached to any computing system.’” *Id.* at 9 (quoting Ex. 1005, 6:42–51). More fully, Johnson states as follows:

In the preferred embodiment, the computing system is a computer controlled gaming machine in which the only mass storage device is a read-only drive such as a CD-ROM or DVD-ROM drive. One skilled in the art will recognize that the present invention may be used with any mass storage device attached to any computing system without deviating from the spirit and scope of the present invention.

Ex. 1005, 6:42–51. Petitioner shows, and Patent Owner does not dispute, that “[i]t was well known to a [person having ordinary skill in the art] that circuit board-based mass storage devices were ubiquitous long before 2005, with well-known examples including USB flash drives, SD cards, [and] Compact Flash cards.” Pet. 9 (citing Ex. 1003 ¶¶ 141–142); PO Resp. 41–50. Thus, Petitioner has shown by a preponderance of the evidence that it would have been obvious to implement the “mass storage device” of Johnson “with any of the well-known and commercially available circuit board-based memory devices,” also referred to by Petitioner as a Read Only Memory Board,” corresponding to the claimed “board including a memory.” *Id.*

Petitioner also shows that Martinek teaches: (1) “a Read Only Memory board 502 that is different from and connects to the motherboard

via connection 508” and includes memory storage elements 501 that store “game code;” (2) board 504 that includes memory element 507 for storing an authentication program; and, (3) that board 502 and board 504 may be implemented as part of the same board. *Id.* at 9–11 (citing Ex. 1006 ¶¶ 29, 45, 79, 114, 133–135, Fig. 14; Ex. 1003 ¶¶ 80–85, 87–89, 143–148).

Martinek evidences what a person of ordinary skill in the art would have known with regard to the “mass storage device” taught by Johnson, and supports Dr. Wolfe’s opinion that “[a]t and well before the time of alleged invention of the ’540 Patent, numerous forms of mass storage devices were well known to a [person having ordinary skill in the art] that comprised a board including memory,” including “USB flash drives, SD cards, [and] Compact Flash cards.” Ex. 1003 ¶ 142.

(ii) a motherboard which is different from the board and connects to the board, the motherboard including another memory which is different from the memory, said another memory configured to read out and store the game program stored in the memory; and

Corresponding to the recited “motherboard” and “another memory,” Petitioner shows that Johnson teaches “a block of system random access memory (RAM) 225 for use in storing application modules and data for use when implementing a gaming machine,” but does not expressly state that RAM 225 is on a “motherboard.” Pet. 14, 20–21 (quoting Ex. 1005, 7:16–19). Petitioner explains that RAM 225 is different than the “mass storage device” of Johnson, and that a person having ordinary skill in the art “would have understood that Johnson’s computing system contemplates a motherboard to which the processor and system RAM are connected.” Pet. 14–15 (citing Ex. 1003 ¶¶ 57–59, 150–152). In support, Dr. Wolfe explains that “[u]pon reading the disclosures of Johnson, a [person of

ordinary skill in the art] would have recognized that Johnson discloses that the electronics of the computing system, including main processing unit (CPU) 221, RAM 225, I/O interface module 223, and BIOS ROM, are situated on a motherboard and that the mass storage device is connected to the motherboard (via the mass storage device interface module 224).”

Ex. 1003 ¶ 59; *see also id.* ¶ 149 (acknowledging that Johnson “does not specifically state that the system RAM is provided on a motherboard (using that specific term),” and stating that it would have been obvious to a person of ordinary skill in the art to include a motherboard with System Ram 225 on it “as it was widely known and also taught in Martinek”).

Martinek evidences what a person of ordinary skill in the art would have known with regard to the arrangement of components taught by Johnson, including the use of a “motherboard,” and supports Dr. Wolfe’s opinion that “[i]t would have been obvious to a [person having ordinary skill in the art] that a general-purpose computer system, such as disclosed in both Johnson and Martinek, would include a motherboard upon which the processor and RAM (among other components) are situated.” Ex. 1003 ¶ 154 (citing Ex. 1006 ¶ 131 (noting that the second generation IVC system “may be inserted into the gaming apparatus, for example, connected to a motherboard”)); *see also id.* ¶ 156 (stating that “[w]ell prior to the time of the alleged invention of the ’540 Patent, motherboards were a common, well-known, and well-understood method for aggregating and electrically coupling different electronic components in a processing device” (citing Ex. 1008 ¶ 38 (stating “[i]t is well within the skill of those in the video wagering game art to construct motherboards, particularly PC motherboards (e.g., with Intel 8086-compatible processors, memory, and nonvolatile storage such as EPROM or disk storage)")))).

Thus, Petitioner shows that using a motherboard to implement Johnson would have been obvious,” because “motherboards were ubiquitously utilized in computing devices, including for interconnecting components in computing systems like the one in Johnson,” and the ’540 patent acknowledges that “generic motherboards” were “commercially available.” Pet. 16 (citing Ex. 1001, 6:25–30, 11:13–16; Ex. 1003 ¶¶ 29–38, 153); PO Resp. 41–50. Petitioner also shows that Johnson teaches “loading executable programs from the mass storage device into RAM,” and that such loading processes correspond to the recited reading out of one memory and storing in another. Pet. 20–21 (citing, e.g., Ex. 1003 ¶¶ 46–49, 159–162); *see also id.* at 22 n.4 (explaining that consistent with the ’540 patent, a person of ordinary skill would have understood that “the CPU executes the loading (read out and write) process that results in ‘another memory’ storing the information”) (citing Ex. 1001, 9:60–67, Fig. 5; Ex. 1003 ¶¶ 159–162). Thus, Petitioner shows by a preponderance of the evidence that it would have been obvious to implement a “motherboard,” as claimed, to which a processor and RAM 225 of Johnson are connected, and to configure the memory of the motherboard to read out and store a game program from the memory of a mass storage device, such as a board.

(iii) a CPU which is provided on the motherboard, for executing the game based upon the game program stored in said another memory,

Petitioner shows that main processing module 221 of Johnson corresponds to the recited CPU. Pet. 23–24 (citing Ex. 1005, 7:8–15).⁸ Although Johnson does not state expressly that main processing module 221 is on a “motherboard,” Petitioner shows that a person of ordinary skill in the art would have understood that main processing module 221 would have been provided on a motherboard, as well as that it would have been obvious to implement Johnson in such a way as to provide main processing module 221 on a motherboard. Pet. 23–24; Ex. 1003 ¶¶ 165–167. Dr. Wolfe explains that “even though Johnson does not specifically state that the CPU is provided on a motherboard (in those terms), a [person of ordinary skill in the art] would have recognized that the computer-controlled gaming machine described in Johnson would include a motherboard upon which the CPU is provided.” Ex. 1003 ¶ 164; *see also id.* ¶¶ 149–157 (discussing Martinek’s express teaching of arranging a CPU and memory on a motherboard).

Additionally, Petitioner shows, and Patent Owner does not dispute, that a person of ordinary skill would have understood that Johnson’s main processing module 221 executes the game based on the game program stored in random access memory (RAM) 225. *Id.* at 23–24 (citing Ex. 1003 ¶¶ 165–167; Ex. 1005, 5:14–18, 7:16–19, 7:51–56, 9:14–17, Fig. 2). Thus,

⁸ As explained above, Johnson identifies “main processing module 221” and “display interface module 222.” Ex. 1005, 7:2–6, Fig. 2. In certain places, however, both Johnson and Petitioner in the Petition mistakenly refer to “main processing module 222,” which we understand to be intended to be a reference to “main processing module 221.” Pet. 23; Ex. 1005, 7:9–11.

Petitioner has shown by a preponderance of the evidence that it would have been obvious to provide a “CPU” on a “motherboard” with memory to execute the game program stored in that memory.

the CPU being configured to:

- (a) read out the authentication program from the memory of the board, and then, store the read out authentication program in said another memory of the motherboard [(the “Read Out and Store Limitation”)];*
- (b) execute the authentication program stored in said another memory in the process (a), and then, authenticate the game program in the memory of the board, based upon the executed authentication program [(the “Execute and Authenticate Limitation”)];*
- (c) write the game program in the memory of the board, to said another memory of the motherboard, in a case where the game program in the memory of the board is authenticated as a result of the authentication process (b) [(the “Write Limitation”)]; and*
- (d) execute the game based upon the game program written to said another memory of the motherboard in the process (c) [(the “Execute the Game Limitation”)].*

With regard to the Read Out and Store Limitation, Petitioner shows that in Johnson main processing module 221 (corresponding to the recited “CPU”) loads verification module 330 (corresponding to the recited “authentication program”) from mass storage device 102 (corresponding to the recited “memory of the board”). Pet. 25–27 (citing Ex. 1003 ¶¶ 169–173; Ex. 1005, 8:64–9:2, 9:23–26, Fig. 5). Petitioner explains that the ’540 patent “describes the ‘read out’ and ‘then store’ process as a ‘loading process.’” Pet. 27 (citing Ex. 1001, 9:32–34, Figs. 5, 6; Ex. 1003 ¶¶ 46–49, 170–173). Petitioner further explains that “[w]hen implementing Johnson with a Read Only Memory board,” it would have been obvious to a person of ordinary skill “to configure the CPU to load (i.e., read out and store) the

authentication program from memory of the Read Only Memory board into RAM of the motherboard,” because it would have been “substantively no different from programming the CPU to read data from a mass storage device communicatively coupled to the CPU.” *Id.* (citing Ex. 1003 ¶174).

With regard to the Execute and Authenticate Limitation, Petitioner shows that Johnson’s main processing module 221 (corresponding to the recited “CPU”) is configured to execute verification module 330 (corresponding to the recited “authentication program”) stored in RAM (corresponding to “said another memory”) and to then “authenticate the game program” by application module 320. Pet. 28–31 (citing Ex. 1003 ¶¶ 66–70, 175–179; Ex. 1005, 4:47–51, 5:8–18, 7:59–65, 8:1–21, 8:31–44, 8:64–9:2, 9:29–59, Figs. 4, 5). Further, Petitioner explains that when using a Read Only Memory board as the mass storage device in Johnson, it would have been obvious to a person having ordinary skill in the art to use “the CPU on the motherboard (e.g., as opposed to a separate ‘processing intelligence’ on Martinek’s Read Only Memory board) to execute the verification module to authenticate the contents of the relevant memory element(s).” Pet. 30–31 (citing Ex. 1003 ¶¶ 87–89, 130–133, 179; Ex. 1006 ¶¶ 26, 29). Dr. Wolfe provides a detailed explanation of how “Johnson teaches that the verification module stored on the mass storage device verifies that the contents of the mass storage device are valid.” Ex. 1003 ¶¶ 66–70 (explaining how Johnson “uses file names, file starting locations, file sizes, and a CRC checksum for each file on the mass storage device to generate ‘computed verification data’ as part of the verification procedure” (citing Ex. 1005, 8:14–21, Fig. 4)).

With regard to the Write Limitation, Petitioner shows that a person of ordinary skill in the art “would have understood that Johnson writes game

programs in the system RAM only after verification of the files on the mass storage device is successfully completed.” Pet. 31–33 (citing Ex. 1003

¶¶ 180–183; Ex. 1005, Abstract, 3:53–42, 7:16–19, 9:6–21, 10:24–26, Fig. 5). Johnson expressly states as follows:

Before permitting an initial access to the computer readable data from the mass storage device, the method verifies the contents of the mass storage device by executing the instructions contained within the mass storage verification module to generate a set of computed verification data and comparing the set of computed verification data with the mass storage verification data. Finally, the method permits additional access to the computer readable data stored on the mass storage device only if the set of computed verification data matches the mass storage verification data.

Ex. 1005, 3:33–42. Petitioner also shows that Johnson expressly teaches loading “only enough of the operating system that is needed to access the files stored on the mass storage device 211 and perform the verification operations,” which would have been understood not to include game programs. Pet. 32–33 (quoting Ex. 1005, 9–12). As Dr. Wolfe explains, a person of ordinary skill in the art would have understood from Johnson that “that the main processing module is configured to execute a gaming application module only after the mass storage device passes the verification procedure,” and that “at least for the embodiment where the entire operating system is not loaded into system RAM until the verification procedure successfully completes . . . the application module files (i.e., game program)

are also not loaded or written into RAM until the verification procedure successfully completes.”⁹ Ex. 1003 ¶¶ 182–183 (emphasis omitted).

With regard to the Execute the Game Limitation, Petitioner relies on the same arguments and evidence discussed above with regard to the recitation in claim 1 of “a CPU which is provided on the motherboard, for executing the game based upon the game program stored in said another memory.” Pet 34–35 (citing, e.g., Ex. 1003 ¶ 187). For the same reasons we found that Johnson renders obvious the recited CPU on a motherboard for executing the game, as discussed above, we likewise find Petitioner has shown that Johnson teaches executing the game program, as required by the Execute the Game Limitation.

Patent Owner advances several arguments in opposition, which we find unavailing for the reasons provided below. First, Patent Owner argues that Johnson fails to disclose the Execute and Authenticate Limitation (referred to by Patent Owner as “Element 1(iv)(b)”). PO Resp. 42–46; PO Sur-reply 9–12. In particular, Patent Owner argues that Johnson fails to

⁹ Petitioner also argues that Martinek teaches transferring “gaming system operating code and game data sets” to volatile memory for faster access, and teaches that it is advantageous to perform “security and validation” to data “prior to loading into various memory devices in the gaming machine (such as RAM and NVRAM).” Pet. 33 (quoting Ex. 1006 ¶¶ 95, 114); *see also id.* (citing Ex. 1003 ¶¶ 184–186). As discussed below, Patent Owner does not dispute that Johnson teaches the Write Limitation, itself, but focuses instead on whether the game program is authenticated prior to being written to the memory of the motherboard, as required by the Execute and Authenticate Limitation. *See* PO Resp. 41–50 (arguing, for example, that Johnson “requires moving the [game] program from the mass storage device into RAM prior to authentication”). Because we find the Write Limitation would have been obvious over Johnson, alone, we do not reach Petitioner’s alternative contention based on the combination of Johnson and Martinek as to this limitation.

authenticate the game program in the memory of the board (i.e., in Johnson's mass storage device), but, instead, "requires moving the program from the mass storage device into RAM prior to authentication." PO Resp. 42–43 (citing Ex. 1003 ¶ 177; Ex. 1005, Fig. 5; Ex. 2041 ¶ 157). Patent Owner's argument is fundamentally at odds both with what claim 1 requires and with what Johnson teaches.

In particular, Patent Owner argues that Johnson uses operating system 503 to access the mass storage device, and a person of ordinary skill in the art would have understood that Johnson "relies on the file access program of the operating system, and in such an operation, the data in the mass storage device 21 from the files is loaded to RAM 225 to be processed by the CPU." PO Resp. 43 (citing Ex. 1005, 9:3–10; Ex. 2041 ¶ 159). The cited portion of Johnson states as follows:

In the preferred embodiment, the Load OS module 503 will load the entire operating system into the system RAM 225 before the Load Verification module 504 will begin operation. Of course, this process may be modified to load only enough of the operating system that is needed to access the files stored on the mass storage device 211 and perform the verification operations before the verification process begins.

Ex. 1005, 9:3–10. Dr. Yang states that from this disclosure a person of ordinary skill in the art would have understood that Johnson "is relying on the file access program of the operating system, and in such an operation, the data in the mass storage device 21 from the files is loaded to RAM 225 to be processed by the CPU," but offers no further explanation or support for his opinion. Ex. 2041 ¶¶ 159–160.

Patent Owner further discusses how the '540 patent performs a verification process, but does not suggest claim 1 recites that specific process. PO Resp. 43–46 (stating, for example, that the authentication

program of the '540 patent “reads the game program data at the byte-level without using the file access program in the operating system (which Johnson uses)” (citing Ex. 1001, 9:42–67; Ex. 2041 ¶ 164).

Patent Owner seeks to read into claim 1 a requirement that *nothing* related to, or any portion of, the gaming information be read into RAM from the mass storage device of Johnson prior to authenticating the game program. Claim 1, however, merely states that the game program is authenticated and then written to the memory of the motherboard; claim 1 does not broadly preclude data or files other than the game program from being written to RAM to facilitate the authentication of the game program. Moreover, Johnson expressly states that access to the game program is not allowed until the game program is authenticated. *See, e.g.*, Ex. 1005, code (57). Dr. Wolfe explains how Johnson does this by using “file names, file starting locations, file sizes, and a CRC checksum for each file on the mass storage device to generate ‘computed verification data’ as part of the verification procedure.” Ex. 1003 ¶¶ 66–70 (citing Ex. 1005, 8:14–21, Fig. 4). Petitioner asserts, and we agree, that claim 1 does not require a “byte-level read” of the game program files during authentication. Pet. Reply 9. Petitioner further explains persuasively, with support, as follows:

[T]he claim language “in the memory of the board” simply refers to the location of the game program (i.e., “the game program in the memory of the board”). It does not refer to where the authentication takes place. The authentication, of course, takes place in the CPU, which has to access the game program from the memory board to perform authentication. *See, e.g.*, '540 Patent, 9:53-57 (“the main CPU 21 performs an authentication process f2 with respect to the read-out game system program 30b”). The authentication is, of course, not literally performed

within the memory of the board itself. *See id.*; *see also* Yang Tr. 19:8–23.

Pet. Reply 10 (emphasis omitted).

In its Sur-reply, Patent Owner directs us to cross-examination testimony by Dr. Wolfe that purportedly “concedes that the data being authenticated in Johnson is copied to the system RAM.” PO Sur-reply 9 (citing Ex. 2014, 45:15–46:11). The testimony regarding the authentication process of Johnson at issue is as follows:

Q. Can you explain the process for how that data is located and actually verified by the program?

A. It doesn’t get into all the details because these are very ordinary methods, but it talks about doing a file by file authorization, or verification. Normally that would be done by file name or file number, doing that in a main loop. And then it does it using file signature comparison. That again would be an ordinary process where you would compute the signature of a file and compare it to a reference signature.

Q. So in this loop that you just mentioned, are the data files still on the mass storage device at that point or have they been transferred to the RAM when those calculations are made, or does Johnson not say?

A. He does say that in a preferred embodiment he copies them to the RAM.

Q. Where does it say that?

A. I’m sorry, wait a minute. Let me read that more carefully. It may just be that the reference data, the reference signatures are copied to the RAM. I don’t think he’s specific as to whether or not the application files and application data are copied to the RAM.

Ordinarily in performing any analysis of something that’s on mass storage, you would copy at least, piece by piece you would copy into RAM.

Ex. 2014, 45:2–46:11. In the preceding testimony, Dr. Wolfe does not state that in Johnson the game program is transferred to RAM prior to

authentication, which, as explained above, would be contrary to the express disclosures of Johnson.

Patent Owner's argument appears to be that claim 1 precludes any data from the mass storage device from being loaded into RAM prior to authentication of the game program. PO Sur-reply 10–12 (arguing that Patent Owner “distinguished Johnson on that [sic] basis Johnson authenticates data located in the system RAM, not in the memory of another board, and does not disclose any structure even capable of meeting the claim language,” and that “Petitioner's expert acknowledged that Johnson's technique works by authenticating information that is moved from the mass storage device into the system RAM”). Claim 1, however, includes no such broad limitation, and instead only requires that the game program in the memory of the board first be authenticated before the *game program* is written to motherboard. Patent Owner's arguments do not rebut the express disclosure of Johnson and we credit the declaration testimony of Dr. Wolfe over Dr. Yang in this regard as Dr. Wolfe's testimony is consistent with and supported by Johnson and the language of claim 1.

Second, Patent Owner broadly argues that “Johnson in view of Martinek fails to teach or suggest the CPU being configured as required by [the Read Out and Store Limitation, the Execute and Authenticate Limitation, the Write Limitation, and the Execute the Game Limitation],” referred to by Patent Owner as “Elements 1(iv)(a) through 1(iv)(d).” PO Resp. 46–50. Patent Owner, however, fails to identify any claimed limitation not taught by the asserted art. For example, Patent Owner suggests that the '540 patent “discloses that its hardware configuration enables the CPU of the motherboard to access the memory of the board before the boot process of the motherboard is complete.” *Id.* at 46 (citing

Ex. 1001, 9:42–67). Claim 1, however, includes no such limitation directed to “the boot process.” Patent Owner also argues that the ’540 patent requires that a “board with a memory must be connected to the motherboard in a special configuration so that the recited functions can be executed in its prescribed order.” *Id.* at 46–47 (citing Ex. 1001, 9:42–67). Claim 1 does not recite a “special configuration.”

Patent Owner’s arguments appear to be directed to showing that the Challenged Claims of the ’540 patent preclude writing the game program into RAM prior to authentication, a point not in dispute. *See supra* § II.C.1; PO Resp. 44–45; *see also* Tr. 42:7–11 (counsel for Patent Owner stating that “I planned to spend most of my time talking about the requirement that the gaming information not be read into RAM until after authentication. It’s apparently not in dispute, so unless the Board has questions on that, I’ll just move on.”). More specifically, Patent Owner argues that “Petitioner needs to demonstrate that the structural elements in the asserted references, *e.g.*, memory and the CPU, are configured to accomplish specific functional objectives as claimed.” PO Resp. at 47. According to Patent Owner, Johnson teaches a “standard mass storage device,” which a person of ordinary skill in the art would have understood to employ “standard interfaces for ‘any’ computing system,” not the “non-standard configuration” of the ’540 patent. *Id.* at 47–48 (citing Ex. 2041 ¶ 174) (emphasis omitted); *see also id.* at 48 (arguing that Martinek does not cure this defect). Patent Owner proceeds to describe a “special combination of signal paths” not taught by either Johnson or Martinek that Patent Owner argues is needed to enable “the board to halt the boot process of the motherboard.” *Id.* at 48–49 (citing, *e.g.*, Ex. 2041 ¶¶ 178–179). Claim 1,

however, does not recite any “special combination of signal paths” or require halting “the boot process.”

Fundamentally, Patent Owner’s argument once again revolves around whether Johnson discloses authenticating the game program before loading the game program into the memory of the motherboard. *See, e.g.*, Tr. 40:10–14 (Patent Owner stating that it “is not asserting that the claims require the specific PCI/IDE architecture” and “is not contending that the claims require the specific byte-by-byte or any particular file path”). During oral argument Patent Owner was asked how its position is consistent with the express disclosure of Johnson that the gaming program is not written to RAM until after authentication. Tr. 46:24–47:5 (referencing Petitioner’s demonstrative slide 16, which quotes Ex. 1005, 8:38–40 (providing that “[t]he gaming machine will not be permitted to access the programs and data on the mass storage device unless it passes the verification procedure”). In response, Patent Owner disregards the express disclosure and instead asserts that “there’s nothing in Johnson that explains how you could get gaming information to the CPU without putting it into RAM first.” Tr. 47:11–48:13. Further in its response, Patent Owner drew attention to the reference to “additional access” in the Abstract of Johnson as “a really important sentence” (Tr. 47:11–48:13) with the full Abstract of Johnson providing as follows:

A system and method for verifying the contents of a mass storage device attached to a computing system having a processor and system random access memory. The computing system is preferably used as part of a computer controlled gaming machine. The mass storage device has computer readable data stored therein. First, a mass storage verification module having a set of computer executable instructions for use in verifying the contents of the mass storage device is storing onto the mass

storage device. Next, the method mass storage verification data for use in verifying the contents of the mass storage device is also stored onto the mass storage device. *Before permitting an initial access to the computer readable data from the mass storage device contents of the mass storage device is verified* by executing the instructions contained within the mass storage verification module to generate a set of computed verification data and comparing the set of computed verification data with the mass storage verification data. Finally, additional access to the computer readable data stored on the mass storage device is allowed only if the set of computed verification data matches the mass storage verification data.

Ex. 1005, code (57) (“Abstract”) (emphasis added).

Patent Owner’s argument is not supported by the express disclosure Patent Owner directs us to, which states that “[b]efore permitting an initial access to the computer readable data from the mass storage device contents of the mass storage device is verified.” *Id.* Moreover, we credit the testimony of Dr. Wolfe, who explains in detail how Johnson verifies the data on the mass storage device prior to loading the game program into RAM.

Ex. 1003 ¶¶ 66–70.

2. *Differences Between the Subject Matter of Independent Claim 4 and the Teachings of Johnson and Martinek*

Independent claim 4 is directed to “a method for controlling a gaming machine,” as opposed to the “gaming machine” of claim 1, but otherwise recites features substantively the same as claim 1. Ex. 1001, 12:64–13:25, 13:48–14:24. Petitioner relies on substantially the same evidence and arguments to show that each of the limitations of both claims 1 and 4 is taught by the combination of Johnson and Martinek. *See* Pet. 35–36; Ex. 1003 ¶¶ 188–194. Patent Owner does not raise any arguments directed to claim 4 beyond the arguments raised with respect to claim 1. *See* PO Resp. 39–50. For the same reasons provided above with respect to claim 1,

we determine that Petitioner shows by a preponderance of the evidence that Johnson teaches or suggests every limitation of claim 4.

3. Rationale in Support of Obviousness

We find that every limitation of claims 1 and 4 would have been obvious over Johnson, alone, and therefore we need not address the arguments of the parties concerning the rationale for combining Johnson and Martinek. In particular, as discussed above, Johnson teaches the use of a mass storage device, but does not expressly identify a “board” as such a device. Johnson also teaches various components as part of its gaming system, including a CPU and memory, but does not expressly state that they are located on a “motherboard.” Petitioner has shown that such an implementation of Johnson would have been well within the knowledge of a person of ordinary skill in the art. Specifically, Petitioner shows, and Patent Owner does not dispute, that a person of ordinary skill in the art “would have been motivated to implement Johnson’s mass storage device in the form of a Read Only Memory board, such as the one suggested in Martinek,” and “would have recognized this to be a simple design choice and would have been motivated to utilize a Read Only Memory board because, for example, such a board is easily expandable,” as well as “a combination of known elements (i.e., circuit boards and common computing memory storage elements) according to known methods (i.e., electrically connecting components on circuit boards) to obtain a predictable result—that the components would have worked together to allow the gaming machine to load programs from a read only memory board into RAM of the motherboard”). Pet. 11–14 (citing Ex. 1003 ¶¶ 130–133, 143–148; Ex. 1006 ¶¶ 25, 133).

To the extent Petitioner relies on Martinek as further express support for these elements Johnson suggests, but does not expressly disclose, we find Martinek is persuasive evidence of what a person of ordinary skill in the art would have known, as explained above. We have also considered Patent Owner’s arguments that a person of ordinary skill in the art would not have been motivated to combine Johnson and Martinek. PO Resp. 39–41. Patent Owner suggests “there is no motivation . . . to combine two distinct, differing schemes” from Johnson and Martinek. *Id.* at 41 (citing Ex. 2041 ¶ 152). Patent Owner, however, acknowledges that Petitioner cites Martinek primarily for its express disclosure of a read-only memory board. *Id.* Patent Owner does not dispute that such a board was a well-known type of mass storage device, as taught by Johnson, but instead suggests that the system taught by Martinek “does not teach an authentication program executed by the CPU on the motherboard,” while “Johnson teaches its mass storage verification module being executed by its process on the motherboard.” *Id.* at 40–41. As to claims 1 and 4, Petitioner is not proposing to combine two “schemes” from Johnson and Martinek, and there is no suggestion that using a board as a mass storage device would require any modification of the scheme taught by Johnson.

F. Alleged Obviousness over Johnson, Martinek, and Diamant

Petitioner contends that the subject matter of claims 2 and 3, which depend from claim 1, and claims 5 and 6, which depend from claim 4, would have been obvious over the combination of Johnson, Martinek, and Diamant. Pet. 36–52. Petitioner’s contentions are supported by Dr. Wolfe. Ex. 1003 ¶¶ 196–238.

Patent Owner argues that “Johnson in view of Martinek and Diamant fails to disclose the hardware (CPU) configuration [of claim 2],” and “also

fails to teach or suggest systems that provide for the authentication procedure to be triggered when power is first supplied to the device,” as required by claim 3. PO Resp. 50–58. Patent Owner also argues that Petitioner’s contentions are “tainted by hindsight bias.” *Id.* at 56. Below we address how Petitioner contends the asserted references teach the additional limitations of claims 2, 3, 5, and 6, along with Patent Owner’s arguments in opposition. Petitioner shows that the asserted references teach or suggest each additional limitation of claims 2, 3, 5, and 6, and provides a sufficient reason in support of the rationale for the asserted combination. We conclude that Petitioner has shown by a preponderance of the evidence that claims 2, 3, 5, and 6 would have been obvious over Johnson in view of Martinek and Diamant.

1. Differences between the Subject Matter of Dependent Claims 2 and 5 and the Teachings of Johnson, Martinek, and Diamant

Claim 2, depending from claim 1, further requires the following:

- a preliminary authentication program for authenticating the authentication program is further stored in the memory of the board and another CPU which is different from the CPU, said another CPU configured to execute the preliminary authentication program, is provided on the board, said another CPU being configured to, prior to performing the process (a):
 - (e) execute the preliminary authentication program stored in the memory of the board, and then, authenticate the authentication program stored in the memory of the board, based upon the preliminary authentication program.

Ex. 1001, 13:26–38. Claim 5, depending from claim 4, contains substantively the same requirements. *Id.* at 14:25–39. Petitioner relies on the same evidence and argument with respect to both claim 2 and claim 5. Pet. 38–47, 51–52.

Claims 2 and 5 require, in short, an additional CPU on the board (other than the motherboard), which executes first in the process an additional program to authenticate the authentication program. Johnson does not disclose “another CPU” on a board (not the motherboard) or a “preliminary authentication program.”

As to the “preliminary authentication program,” Petitioner relies on Diamant’s teaching of a first verification program, integrity checker 24, as the recited “preliminary authentication program.” Pet. 39–41 (citing Ex. 1007 ¶¶ 15, 19, 24, 30, Figs. 2, 3). Diamant expressly provides that “integrity checker 24 may verify or confirm that a verification program such as for example integrity checking algorithm 22 is valid for use on device 10 and has not been corrupted or compromised.” Ex. 1007 ¶ 19.

As to the “another CPU,” Petitioner relies on Martinek’s teaching that a board (not the motherboard) on which an authentication program is stored may include “its own processing intelligence,” such as “a processor, and software” to execute an authentication program. Pet. 41–43; Ex. 1006 ¶¶ 133–135. Petitioner further explains that in the proposed combined system, “the processor 505 of Martinek’s Read Only Memory board is configured to execute the preliminary authentication program (Diamant’s first verification program) stored in the memory of the board, and then, authenticate the authentication program (Johnson’s verification module) also stored in the memory of the board.” Pet. 45 (citing Ex. 1003 ¶¶ 220, 221, 206–219). Subsequent to this additional process, the system functions in accordance with the process of claim 1, as taught and suggest by Johnson as explained above. That is, after the authentication program is authenticated, it is loaded into RAM and the CPU on the motherboard executes the authentication program to authenticate the game program stored on the

board. We find that Petitioner shows by a preponderance of the evidence how the asserted combination teaches or suggests every limitation of claims 2 and 5.

We find unavailing Patent Owner’s argument that the combination of asserted references “fails to disclose the hardware (CPU) configuration” recited in claim 2. PO Resp. 51. Patent Owner stresses that each of the asserted references “only describe one single CPU operation,” whereas claim 2 requires two CPUs performing separate operations. *Id.* at 52–53. For example, according to Patent Owner, Diamant “only has one processor 12 to execute both the integrity checker and the integrity verification algorithm.” PO Resp. 52 (citing Ex. 1007 ¶ 15). We note, however, that Diamant does not specify that only a single processor may be used. To the contrary, Diamant states that “a processor *such as for example* processor 12 may call and execute integrity checker 24,” and that “[i]f integrity checker 24 verifies that integrity checking algorithm 22 is valid, a processor *such as for example* processor 12 may load and execute integrity checking algorithm 22.” Ex. 1007 ¶ 15 (emphasis added). Diamant also expressly states that “processor 12 may be for example a central processing unit of a personal computer or another processor, controller or execution unit suitable for example of running code, software or instructions for a device 10.” *Id.* ¶ 18. From this, Dr. Wolfe states that Diamant “specifically teaches that ‘another processor’ (besides the motherboard CPU) can execute the first verification program (e.g., integrity checker 24).” Ex. 1003 ¶ 214. Patent Owner takes issue with this, arguing that Diamant “only expressly teaches one CPU or processor, controller, etc. for a device 10.” PO Resp. 54 (citing Ex. 1007 ¶ 18).

Both parties are correct in-part, in that Dr. Wolfe has shown that Diamant contemplates a processor other than the “central processing unit” (or motherboard) that executes the preliminary authentication program, and Patent Owner is correct in that Diamant does not expressly disclose using a different processor to execute the preliminary authentication program from the processor used to execute the authentication program. Regardless, Petitioner relies on both a first processor on the motherboard, as taught by Johnson, and a second processor on a board separate from the motherboard, as taught by Martinek. We disagree with Patent Owner’s contention that any feature of claim 2 is absent from Petitioner’s asserted combination.

2. Differences between the Subject Matter of Dependent Claims 3 and 6 and the Teachings of Johnson, Martinek, and Diamant

Claim 3 depends from claim 2 and further recites as follows:

a power unit which is provided in a casing of the gaming machine and connects to the board and the motherboard, wherein:

in a case where the board and the motherboard are activated based upon power supply from the power unit, the CPU and said another CPU execute the processes (a) to (e) as triggered by the activation.

Ex. 1001, 13:39–46. Claim 6 depends from claim 5 and contains substantively the same requirements as claim 3. *Id.* at 14:40–49.

Petitioner acknowledges that Johnson does not expressly state the system includes “a power unit,” but instead reasons that such a “power unit” “would have been obvious as the functionality described in Johnson would not be possible without a unit to provide power to the electrical components.” Pet. 47–50 (citing Ex. 1003 ¶¶ 226–231). There is no dispute that, as Dr. Wolfe explains, “dating back to early computing systems from the 1970s and, certainly well prior to the alleged invention the ’540 Patent in 2005, power units were standard in personal computers, including power

units provided in the casing of the machine and that connect to the motherboard.” Ex. 1003 ¶ 228. Petitioner further shows as follows:

Johnson discloses initiating the disclosed authentication process steps as part of the “boot procedure,” which “typically occurs when a computing system boots.” [Ex. 1005] at 8:45–9:2; *id.* at 3:13–19 (“The present invention solves the above described problems by providing a method and apparatus verifying the contents of a read-only mass storage device . . . as part of a booting process of a computer system where the computing system is part of a computer controlled gaming machine”); *id.* at Fig. 5. A [person having ordinary skill in the art] would have understood that the computer system in Johnson will “boot” when power is applied to the components of the system. [Ex. 1003] ¶ 233.

Pet. 50–51.

Patent Owner argues that claim 3 requires two CPUs and that “[c]onventionally, only the motherboard is activated at power up.” PO Resp. 57–58 (citing Ex. 2034, 1). According to Patent Owner, “[s]ince there are two CPUs in the inventions of Claim 3, Claim 3 recites activating the boot sequences of both the CPU and another CPU, with proper timing, in order to execute the process (a) to (e).” *Id.* at 58 (citing Ex. 1001, 12:3–11).

Petitioner responds that claim 3 requires no more than that “the CPU and said another CPU execute the processes (a) to (e) in claims 1 and 2 where the board and the motherboard are activated based on power from a power supply.” Pet. Reply 20. Petitioner asserts that claim 3 does not recite any “proper timing” requirement or “separate booting up sequences.” *Id.*

We agree with Petitioner. Claim 3 merely recites “where the board and the motherboard are activated based upon power supply from the power unit.” Even if only the motherboard is activated at “power up,” claim 3 does not require both to be activated at the same time at “power up,” much less

require any particular “proper timing.” For the preceding reasons, Petitioner has shown by a preponderance of the evidence that Johnson suggests to a person of ordinary skill in the art the additional requirements of a “power unit” as recited by claims 3 and 6.

3. Rationale in Support of Obviousness

In support of the asserted combination, Petitioner offers multiple reasons why a person of ordinary skill in the art would have had reason to combine the teachings of Johnson, Martinek, and Diamant, as proposed by Petitioner. Pet. 37–47. Petitioner first notes that all three references are “in the same field of endeavor” and “reasonably pertinent to certain problems addressed by the ’540 Patent . . . and also to certain proposed solutions.” Pet. 37.

In the process recited in claim 1, the authentication program is loaded into RAM without authentication. Claim 2 requires, in part, that the authentication program, itself, be authenticated. Petitioner explains that a person of ordinary skill in the art “would have been motivated to implement” the combined system proposed by Petitioner “to ensure that all data loaded into RAM, including Johnson’s verification module (i.e., authentication program . . .), is authenticated before being loaded into RAM and executed by the motherboard processor, thereby protecting the motherboard RAM and CPU from malicious code.” Pet. 42 (citing Ex. 1003 ¶¶ 201–204, 212–219), 46–47. Diamant, itself, makes clear the reason for including a preliminary authentication program it describes, explaining that “integrity checker 24 [(a preliminary authentication program)] may verify or confirm that a verification program such as for example integrity checking algorithm 22 is valid for use on device 10 and has not been corrupted or compromised.” Pet. 40 (emphasis omitted) (quoting Ex. 1007 ¶ 19).

Petitioner also explains that Martinek provides “the express motivation for implementing the system in this way.” *Id.* at 42–43 (citing Ex. 1003 ¶ 212; Ex. 1006 ¶¶ 14, 26). Dr. Wolfe explains that Martinek teaches the use of Read Only Memory Board that includes a processor to authenticate data stored in the memory of the board. Ex. 1003 ¶¶ 83, 87, 88, 211. Dr. Wolfe states as follows:

A [person having ordinary skill in the art] would have understood that implementing a separate “preliminary” verification program as taught by Diamant, using processor 505 of Martinek to first authenticate Johnson’s verification module before loading it into RAM accomplishes the goal expressly described in Martinek of acting as a gate to “allow data to enter a host computer only after validation.” Martinek [(Ex. 1006)] at [0114]; *see also id.* at [0026] (describing that “security and validation [are] performed advantageously to data prior to loading [the data] into various memory devices in the gaming machine (such as RAM and NVRAM)”). A [person having ordinary skill in the art] would have been motivated to provide a specific mechanism in the form of a “preliminary” verification program as taught by Diamant in the system of Johnson to accomplish the benefit expressly taught in Martinek. Providing this “preliminary” verification program would enhance the overall security of the system by ensuring that Johnson’s verification module was not compromised before being loaded into main memory and before it was executed by the motherboard CPU to authenticate the remainder of the mass storage device content.

Id. ¶ 212 (third and fourth alteration in original). Dr. Wolfe also provides reasons why in the proposed combination the CPU on the motherboard suggested by Johnson would continue to be used execute the verification program, as required by the claim. *Id.* at 215. In sum, “[e]xecuting verification module 330 on the motherboard CPU is expressly described in Johnson, and a [person having ordinary skill in the art] would see no reason to alter this disclosure and no benefit in doing so,” because “maintaining this

functionality in the motherboard carries no drawback because the verification module would have already been authenticated before loading to the motherboard.” *Id.*; *see also id.* ¶ 216 (summarizing the reasons and motivations for the asserted combination); Pet. 42–47 (discussing reasons in support of combination).

After all of the above, Dr. Wolfe also states as follows:

I also note that, as there are only two processors in the combined system (i.e., Johnson’s motherboard processor and Martinek’s Read Only Memory board processor) and two authentication programs (i.e., Diamant’s preliminary authentication program and Johnson’s authentication program), it certainly would have been obvious to a [person of ordinary skill in the art] to try executing the various authentication programs on either of the two processors with more than a reasonable expectation of success given the predictable nature of motherboard and other board processors and authentication programs and the fact that both Martinek and Diamant disclose that the various authentication programs can be run on any available processor.

Id. ¶ 217 (citing Ex. 1006 ¶ 29; Ex. 1007 ¶¶ 15, 18); *see also* Pet. 45 (after discussing numerous reasons in support of the proposed combination, stating “or at a minimum obvious to try the architecture proposed above”). We draw attention to this because, in opposition, Patent Owner argues “it would not have been obvious for a [person of ordinary skill in the art] to try,” and that “one must be motivated to do more than merely to vary all parameters or try each of numerous possible choices until one possibly arrived at a successful result.” PO Resp. 52 (emphasis omitted) (quoting *In re Stepan Co.*, 868 F.3d 1342, 1347 (Fed. Cir. 2017)). Petitioner notes that “Dr. Wolfe’s obvious to try opinion (in paragraph 217) was just one alternative motivation to combine opinion proffered by Dr. Wolfe,” and summarizes the various other reasons Dr. Wolfe provided in support of the

asserted combination. Pet. Reply 18–19. We do not rely on “obvious to try” as the rationale that supports the asserted combination, and instead find for the reasons explained above that the express disclosures in the asserted references provide reasons and motivations that support the asserted combination.

Patent Owner also emphasizes, without cited support, that a person of ordinary skill in the art “would not want to combine these references, to achieve two CPUs and two authentication programs when the references each only describe one single CPU operation.” PO Resp. 52–53. We disagree that the obviousness of claims 2 and 5 turns on whether a single reference discloses two CPUs. The modification proposed by Petitioner above explains in detail how it applies the teachings to arrive at a system with two CPUs, where Johnson uses a CPU on a motherboard and Martinek uses a CPU on another board, and details persuasive reasons why a person of ordinary skill in the art would have understood a benefit of combining those teachings to arrive at a process that uses two CPUs.

We have considered each of Patent Owner’s additional arguments and find they do not counter the rationale provided by Petitioner in support of the combination, nor do they show that Petitioner’s contentions are infected with improper hindsight bias. PO Resp. 53–57. Patent Owner argues that “when adding Diamant to the combination of Johnson and Martinek, the processor 12 of Diamant must be associated with either processor 221 of Johnson or processing intelligence 201 of Martinek.” *Id.* at 53. As Petitioner explains, its asserted combination “does not implement Diamant’s processor 12.” Pet. Reply 16–17. Patent Owner argues that “Martinek specifically prevents the CPU on the motherboard from being involved in the authentication process.” PO Resp. 54. As Petitioner explains, in its

combination “Martinek’s ROM board CPU is only used to execute the preliminary authentication program (Diamant’s first verification program),” and “[a]fter authentication succeeds, Martinek opens its ‘gate’ and allows Johnson’s motherboard CPU to access the authenticated data stored on the ROM board, namely Johnson’s verification module 330.” Pet. Reply 17–18.

G. Alleged Obviousness over Morrow ’952, Morrow ’771, and Diamant

Petitioner contends that the subject matter of independent claims 1 and 4 of the ’540 patent would have been obvious over the combination of Morrow ’952, Morrow ’771, and Diamant. Pet. 52–69; Pet. Reply 20–24.¹⁰ Petitioner’s contentions are supported by Dr. Wolfe. Ex. 1003 ¶¶ 239–307. Patent Owner disputes these contentions, primarily with regard to whether the asserted combination teaches the Execute and Authenticate Limitation or what Patent Owner characterizes as the claimed configuration between the motherboard and the board. PO Resp. 58–64; PO Sur-reply 20–22.

We consider and address below the arguments and evidence of each party and determine for the reasons provided that Petitioner shows by a preponderance of the evidence that Morrow ’952 teaches or suggests every limitation of claims 1 and 4 of the ’540 patent. We conclude that Petitioner has shown by a preponderance of the evidence that claims 1 and 4 are unpatentable over Morrow ’952.

1. Differences Between the Subject Matter of Independent Claim 1 and the Teachings of Morrow ’952, Morrow ’771, and Diamant

Petitioner provides a detailed explanation of how Morrow ’952, alone, teaches or suggests each limitation of claim 1, with further support cited in

¹⁰ Petitioner refers to Morrow ’952 (Ex. 1009) as “Morrow.” To avoid confusion with Morrow ’771 (Ex. 1010) we use “Morrow ’952.”

Morrow '771 and Diamant, which we analyze below, along with Patent Owner's arguments in opposition.

A gaming machine, comprising:

(i) a board including a memory in which a game program for executing a game and an authentication program for authenticating the game program are stored;

To the extent the preamble is limiting, Petitioner shows that Morrow '952 teaches a "gaming machine." Pet. 53–54 (citing, e.g., Ex. 1009 ¶ 55, Fig. 4; Ex. 1003 ¶¶ 102–108, 252. With regard to the recited "board including a memory" that stores a game program and an authentication program, Petitioner shows that Morrow '952 teaches persistent storage media 90 (corresponding to a "board") on which application program 92 (corresponding to a "game program") and verification software 90 (corresponding to an "authentication program") are stored. *Id.* 54–57 (citing, e.g., Ex. 1009 ¶¶ 2–4, 11, 38, 40, 45, 52, 55, 65, Fig. 1). Dr. Wolf explains that Morrow expressly describes persistent storage media 90 as including "a removable flash memory device," and that a person of ordinary skill in the art "would have recognized that such removable flash memory device includes, for example, SD cards and Compact Flash cards," and would have known "that SD cards and Compact Flash cards (both removable flash memory devices) comprised a board including memory." Ex. 1003 ¶ 260 (citing, e.g., Ex. 1009 ¶¶ 13, 40).

(ii) a motherboard which is different from the board and connects to the board, the motherboard including another memory which is different from the memory, said another memory configured to read out and store the game program stored in the memory; and

Corresponding to the recited "motherboard" and "another memory," Petitioner shows that gaming machine 10 of Morrow '952 includes CPU 60

and “volatile storage media such as random access memories (RAMs) 76,” a different memory from persistent storage media 90. Pet. 58 (citing Ex. 1009 ¶ 37, Fig. 1). Although Morrow ’952 does not explicitly use the term “motherboard,” Dr. Wolfe states that both the gaming machine of Morrow ’952 and the motherboard described in the ’540 patent are “composed of commercially available equipment with highly generic characteristics,” and explains that a person of ordinary skill in the art would have understood Morrow ’952 “to disclose that the electronics of the gaming machine 10, including processor (CPU) 60 and system RAM 76, among other components, are situated on a motherboard,” and would also have recognized that “persistent storage media 90 is connected to the motherboard, thereby allowing the CPU 60 to load data files from persistent storage media 90 into RAM 76.” Ex. 1003 ¶¶ 266, 267 (citing, e.g., Ex. 1001, 6:25–30, 11:13–16; Ex. 1009 ¶¶ 37, 63, Fig. 1). Thus, Petitioner shows that a person of ordinary skill in the art “would have recognized that the described gaming device would include a motherboard,” because data files are loaded from persistent storage media 90 to RAMs 76 and, consistent with a gaming machine “composed of components with highly generic characteristics that are interconnected,” CPU 60 and RAMs 76 would have been situated on a motherboard. *Id.* (citing Ex. 1003 ¶¶ 109–111, 265–267; Ex. 1008, 37, 63, Fig. 1).¹¹

¹¹ Alternatively, Petitioner contends Morrow ’771 expressly teaches a gaming machine with a processor and RAM that uses a “motherboard.” Pet. 58–59. Because we find the recited “motherboard” would have been obvious over Morrow ’952, alone, we do not reach Petitioner’s alternative contention based on the combination of Morrow ’952 and Morrow ’771.

Petitioner also shows that Morrow's express disclosure that "[f]or faster access, the processor 60 may move the contents of the file allocation structure 99 into a RAM 76," supports Petitioner's contention that loading game program 92 of Morrow '952 into RAM 76 of the motherboard would have been obvious to a person of ordinary skill in the art. *See* Pet. 60–61 (quoting Ex. 1009 ¶ 63).

(iii) a CPU which is provided on the motherboard, for executing the game based upon the game program stored in said another memory,

Petitioner shows that CPU 60 of Morrow '952 corresponds to the recited CPU. Pet. 61–63 (citing Ex. 1009 ¶¶ 37, 45, Fig. 1). Although Morrow '952 does not state expressly that CPU 60 is on a "motherboard," Petitioner shows that a person of ordinary skill in the art would have understood that CPU 60 would have been placed on a motherboard for the same reasons discussed above in regard to the motherboard limitation of claim 1. *See also* Ex. 1003 ¶ 279 (Dr. Wolfe stating a person of ordinary skill "would have recognized that the described gaming machine [of Morrow '952] would include a motherboard upon which the CPU is provided," and that such an implementation would have been obvious "as the use of a motherboard was widely known and also taught in Morrow '771"). Having shown that it would have been obvious to store the game program in memory, as explained above, Petitioner likewise shows that it would have been obvious to execute the game program stored in memory. Pet. 62–63. In this regard, Dr. Wolfe explains that Morrow '952 states that "[f]or faster access, the processor 60 may move the contents of the file allocation structure 99 into a RAM 76," suggesting to a person of ordinary skill in the art that "the CPU 60 is configured to execute the game

based upon the gaming application program 92 stored in RAM.” Ex. 1003 ¶ 281 (quoting Ex. 1009 ¶ 63).

the CPU being configured to:

- (a) read out the authentication program from the memory of the board, and then, store the read out authentication program in said another memory of the motherboard [(the “Read Out and Store Limitation”)];*
- (b) execute the authentication program stored in said another memory in the process (a), and then, authenticate the game program in the memory of the board, based upon the executed authentication program [(the “Execute and Authenticate Limitation”)];*
- (c) write the game program in the memory of the board, to said another memory of the motherboard, in a case where the game program in the memory of the board is authenticated as a result of the authentication process (b) [(the “Write Limitation”)]; and*
- (d) execute the game based upon the game program written to said another memory of the motherboard in the process (c) [(the “Execute the Game Limitation”)].*

With regard to the Read Out and Store Limitation, Petitioner shows that in Morrow ’952 verification software 70 (an “authentication program”) may be stored on persistent storage media 90. Pet. 63 (citing Ex. 1009 ¶ 56). Petitioner acknowledges that Morrow ’952 does not expressly state that verification software 70 is read out and stored into RAM. *Id.* Petitioner explains, instead, that it would have been obvious to do so, as it was well-known and, particularly, in light of the disclosure in Morrow ’952 of the benefit of storing a program in RAM “to allow the motherboard CPU to access the verification program code more quickly during operation.” Pet. 64 (citing Ex. 1003 ¶ 287; Ex. 1009 ¶ 63). As Dr. Wolfe explains, a person of ordinary skill in the art “would have found it obvious to implement the system, such that the CPU is configured to load (i.e., read out and then

store) the verification software program from the memory of the board (i.e., *Morrow's* persistent storage media 90) into RAM of the motherboard (in the same way, and for the same reasons, that *Morrow* teaches such loading in connection with application programs),” as this “was the conventional way to execute application programs stored on a persistent storage device at and before 2005.” Ex. 1003 ¶¶ 287–289. Dr. Wolfe’s opinion is supported, as he explains, by the express disclosure in *Diamant* that a verification program (integrity checking algorithm 22) may be loaded from memory device 16 into RAM 15 to be executed by processor 12, which we view as evidence of what a person of ordinary skill in the art would have known. Pet. 64; Ex. 1003 ¶ 288; Ex. 1007 ¶¶ 11, 15.

With regard to the Execute and Authenticate Limitation, with the authentication program stored in RAM, as suggested by *Morrow '952* as explained above in regard to the Read Out and Store Limitation, Petitioner further shows that *Morrow '952* teaches CPU 60 executes the authentication program to authenticate data files of media 90, including the game program. Pet. 65–66 (citing Ex. 1009 ¶¶ 48, 49). Dr. Wolfe explains that “verification software is executed by processor (CPU) 60 to read identification numbers and/or digital signatures of hardware and software components (including data files 54 and firmware of the hardware components), to analyze the information in comparison to information stored in database 74 (which may be stored in BIOS+ 64, *see* [0040], Fig. 1), and to generate a tilt condition message when any software or hardware component fails verification.” Ex. 1003 ¶ 118–119 (citing Ex. 1009 ¶¶ 41, 42); *see also id.* ¶ 120 (explaining that “[i]n the context of verifying the digital contents of the persistent storage media device 90, for example, *Morrow [’952]* discloses use of Digital Signature Algorithm (DSA) signature generation and

verification or Rivest-Shamir-Adleman (RSA) Algorithm verification,” or “the verification software 70 may use the Rivest-Shamir-Adleman (RSA) algorithm to verify the components 50”) (citing Ex. 1009 ¶¶ 48–51).

Patent Owner argues that Morrow ’952 fails to disclose the Execute and Authenticate Limitation. PO Resp. 58–61; PO Sur-reply 20–22. According to Patent Owner, Morrow ’952 uses file allocation reader 76 “to provide the ability to access the mass storage media prior to executing the operating system.” PO Resp. 59 (citing Ex. 1009 ¶ 63; Ex. 2041 ¶ 223). Petitioner states that “[f]or faster access, the processor 60 may move the contents of the file allocation structure 99 into a RAM 76,” and the “processor 60 may then process the file allocation structure 604 to provide access to files stored in the storage device.” *Id.* at 60 (citing Ex. 1006 ¶ 63). Patent Owner reasons that the ’540 patent “teaches a detailed approach to avoid the use of RAM when authenticating the game program 30a and the game system program 30b.” *Id.* at 60–61 (citing Ex. 1001, 9:42–67). Patent Owner proceeds to discuss “the byte-level reading operation” of the ’540 patent and concludes that “Morrow ’952 does not teach [the Execute and Authenticate Limitation].” *Id.* at 61. We find Patent Owner’s argument unpersuasive because, as Petitioner notes, claims 1 and 4 do not recite any “byte-level reading operation.” Pet. Reply 21. More broadly, contrary to Patent Owner’s assertion, claims 1 and 4 do not preclude all data from being written to RAM. Petitioner explains as follows:

Morrow ’952’s file allocation reader 76 allows files to be accessed for validation even before the operating system (let alone an application program such as a game) is loaded and without the need for storing files in RAM. Petition, 66 (citing Morrow ’952, [0063], Fig. 1). As explained in the Petition, a [person of ordinary skill in the art] would have understood that Morrow ’952 teaches that the game program is not loaded into

RAM (nor is the operating system) unless and until the data files, including the game program, pass verification. Petition, 67 (citing Wolfe Decl., ¶¶294-295)). Morrow ['952] does so by only accessing the file allocation structure 99, which is separate from the actual application programs 92, as shown in Fig. 1. Morrow '952, [0063], Fig. 1. Further, Morrow ['952] teaches that verification can be accomplished by verifying abbreviated bit strings rather than actual files, which also avoids the need to load the program itself into RAM. *Id.* at [0065]; *see also id.* at [0064].

Pet. Reply 22–23. Patent Owner responds that it does not contend that a “byte-level reading operation” is required by claims 1 and 4, but maintains that Morrow '952 “first moves the data into RAM before verification.” PO Sur-reply 21. As explained above, claims 1 and 4 do not address all data, only that the “game program” be written to RAM after authentication.

Patent Owner further argues that we should “ignore Petitioner’s new, unsupported, and technically inaccurate attorney argument that ‘Morrow teaches that verification can be accomplished by verifying abbreviated bit strings rather than actual files, which also avoids the need to load the program itself into RAM.’” PO Sur-reply 22 (citing Pet. Reply 22–23). First, the argument Patent Owner refers to is not new. *See* Pet. 65–66 (addressing in detail the use by Morrow '952 of “an abbreviated bit string” in regard to the Execute and Authenticate Limitation) (citing Ex. 1006 ¶¶ 48–49, Fig. 1; Ex. 1003 ¶¶ 290–292). Second, Petitioner’s argument is not “unsupported” as the Petition cites Dr. Wolfe’s testimony, and Dr. Wolfe explains in detail how “Morrow ['952] discloses authenticating each of the game data files 54 on media 90,” using an “abbreviated bit string.” Pet. 65–66; Ex. 1003 ¶ 292. Lastly, we disagree with Patent Owner’s characterization of Dr. Wolfe’s testimony as “technically inaccurate.” Patent Owner argues that the description in Morrow '952 of using

abbreviated bit strings for verification still requires that the “abbreviated bit strings” are “computed by hashing the entire file and are not merely portions of the file as Petitioner’s attorney argument suggests.” PO Sur-reply 22. Patent Owner’s argument is, itself, unsupported attorney argument, and we credit the testimony of Dr. Wolfe over the attorney argument offered by Patent Owner. Dr. Wolfe’s testimony is also persuasive because it is consistent with Morrow ’952, which states, for example, that “the present invention provides an improved method and system for verifying a device, having components, *before* or during use,” and “the components 50 of the gaming machine 10 are verified at the network server 402 after the gaming machine 10 transmits the identification numbers, hash values, etc., to the network Server 402.” Ex. 1009 ¶¶ 1, 55 (emphasis added).

With regard to the Write Limitation, Petitioner shows Morrow ’952 teaches writing application program 92 (a game program) from the memory of the board (media 90) into RAM 76 (another memory). Pet. 66. Specifically, Petitioner shows that Morrow “teaches that ‘the loading operation of the device’ may be prohibited if a data file fails verification,” that “CPU 60 is configured to execute a gaming application module only after media 90 passes the verification procedure without an indication of tampering,” and that a “‘file allocation reader 76 stored in the BIOS’ may be used to access files for purposes of validation without the need for loading the operating system from media 90 and without the need for storing files in RAM (other than the file allocation structure 99 of media 90).” *Id.* at 66 (citing Ex. 1003 ¶ 294; Ex. 1009 ¶¶ 14, 15, 49, 50, 52, 63, Figs. 1–3).

With regard to the Execute the Game Limitation, Petitioner relies on the same arguments and evidence discussed above with regard to the recitation in claim 1 of “a CPU which is provided on the motherboard, for

executing the game based upon the game program stored in said another memory.” Pet. 68 (citing, e.g., Ex. 1003 ¶¶ 273–281, 293–299). For the same reasons we found that Morrow ’952 renders obvious the recited CPU on a motherboard for executing the game, as discussed above, we likewise find Petitioner has shown that Morrow ’952 teaches executing the game program, as required by the Execute the Game Limitation.

Patent Owner more broadly argues that Morrow ’952 fails to disclose claimed “configuration between the motherboard and the board,” including, for example, a “unique hardware configuration” and “a special signal path to the motherboard.” PO Resp. 62–64. We agree with Petitioner that Patent Owner’s arguments are untethered to the language of claims 1 and 4 and beyond their scope. *See* Pet. Reply 23–24.

2. Differences Between the Subject Matter of Independent Claim 4 and the Teachings of Morrow ’952, Morrow ’771, and Diamant

Independent claim 4 is directed to “a method for controlling a gaming machine,” as opposed to the “gaming machine” of claim 1, but otherwise recites features substantively the same as claim 1. Ex. 1001, 12:64–13:25, 13:48–14:24. Petitioner relies on substantially the same evidence and arguments to show that each of the limitations of both claims 1 and 4 is taught by Morrow ’952. Pet. 68–69; Ex. 1003 ¶¶ 300–307. Patent Owner does not raise any arguments directed to claim 4 beyond the arguments raised with respect to claim 1. *See* PO Resp. 58–64. For the same reasons provided above with respect to claim 1, we determine that Petitioner shows by a preponderance of the evidence that Morrow ’952 teaches or suggests every limitation of claim 4.

3. Rationale in Support of Obviousness

We find that every limitation of claims 1 and 4 would have been obvious over Morrow '952, alone, and therefore we need not address the arguments of the parties concerning the rationale for combining Morrow '952, Morrow '771, and Diamant. In particular, as discussed above, Morrow '952 teaches various components as part of its gaming system, including a CPU and memory, but does not expressly state that they are located on a "motherboard." Morrow '952 also does not expressly state that its verification software is read out and stored to RAM. Petitioner has shown that such an implementation of Morrow '952 would have been well within the knowledge of a person of ordinary skill in the art. To the extent Petitioner relies on Morrow '771 and Diamant as further express support for these elements Morrow '952 suggests, but does not expressly disclose, we find Morrow '771 and Diamant are persuasive evidence of what a person of ordinary skill in the art would have known.

III. CONCLUSION¹²

Claims 1–6 of the '540 patent were shown to be unpatentable by a preponderance of the evidence, as summarized in the table below.

Claim(s)	35 U.S.C. §	Reference(s)	Claim(s) Shown Unpatentable	Claims Not Shown Unpatentable
1, 4	103(a)	Johnson, Martinek	1, 4	
2, 3, 5, 6	103(a)	Johnson, Martinek, Diamant	2, 3, 5, 6	
1, 4	103(a)	Morrow '952, Morrow '771, Diamant	1, 4	
Overall Outcome			1–6	

IV. ORDER

Upon consideration of the record before us, it is:

ORDERED that claims 1–6 of U.S. Patent No. 8,078,540 B2 have been proven by a preponderance of the evidence to be unpatentable; and

FURTHER ORDERED that, as this is a Final Written Decision, a party seeking judicial review of the Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

¹² Should Patent Owner wish to pursue amendment of the challenged claims in a reissue or reexamination proceeding subsequent to the issuance of this decision, we draw Patent Owner's attention to the April 2019 Notice Regarding Options for Amendments by Patent Owner Through Reissue or Reexamination During a Pending AIA Trial Proceeding. *See* 84 Fed. Reg. 16,654 (Apr. 22, 2019). If Patent Owner chooses to file a reissue application or a request for reexamination of the challenged patent, we remind Patent Owner of its continuing obligation to notify the Board of any such related matters in updated mandatory notices. *See* 37 C.F.R. § 42.8(a)(3), (b)(2).

IPR2020-00922
Patent 8,078,540 B2

FOR PETITIONER:

Eric A. Buresh
Jason R. Mudd
Kathleen D. Fitterling
ERISE IP, P.A.
eric.buresh@eriseip.com
jason.mudd@eriseip.com
kathleen.fitterling@eriseip.com

FOR PATENT OWNER:

James Hannah
Jonathan S. Caplan
Jeffrey H. Price
KRAMER LEVIN NAFTALIS & FRANKEL LLP
jhannah@kramerlevin.com
jcaplan@kramerlevin.com
jprice@kramerlevin.com