

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NINTENDO CO., LTD., and NINTENDO OF AMERICA INC.,
Petitioners,

v.

ANCORA TECHNOLOGIES, INC.,
Patent Owner.

Case IPR2021-01338
U.S. Patent No. 6,411,941 B1

PATENT OWNER'S RESPONSE

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, Virginia 22313-1450

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. SUMMARY OF THE '941 PATENT	2
A. The Inventions Disclosed in the '941 Patent.....	2
B. The Challenged Claims	6
C. Prosecution History	7
1. Non-Final Office Action, December 20, 2000	7
2. Amendment, May 23, 2001	10
3. Final Office Action, June 22, 2001	10
4. Amendment and Request for Continued Examination, November 16, 2001	11
5. Non-Final Office Action, January 15, 2002	11
6. Amendment, February 5, 2002	17
7. Notice of Allowability, March 28, 2002	18
D. Prior Invalidity Challenges.....	20
1. Microsoft's Reexamination No. 90/010,560.....	20
2. Apple Litigation	26
3. HTC's Failed Petition for CBM Review	27
4. HTC's Failed Patentable Subject Matter Challenge	27
5. TCL's IPR Petition Asserting Hellman and Chou and the "Me-Too" Challengers.....	28
6. Samsung's Failed Inter Partes Review Petition.....	28
E. Level of Ordinary Skill in the Art	29

F.	Claim Interpretation	30
1.	“Agent”	32
III.	SUMMARY OF THE ASSERTED REFERENCES.....	36
A.	Hellman	37
B.	Chou	43
C.	Shneck	46
IV.	PETITIONER FAILS TO ESTABLISH OBVIOUSNESS BASED ON EITHER ASSERTED GROUND.....	48
A.	Governing Legal Principles.....	49
B.	Claim 1: The Petition Fails to Establish a Valid Motivation for Combining Hellman With Chou	52
1.	The Petition’s Asserted Motivations to Combine Hellman with Chou Are Logically Flawed.....	52
2.	Petitioners Do Not Offer Any Motivation for Storing Hellman’s Value “M” in the Memory Area “Used By the BIOS”	55
C.	Claim 1: Petitioners’ Combinations Fail to Identify the Claimed Step of Using an Agent to Set Up a Verification Structure in the Erasable, Non-Volatile Memory of the BIOS.....	56
1.	The Petition Wholly Fails to Identify Any Teaching in the Combined References of the claimed “Agent,” Which Requires an “OS-Level Software Program or Routine”	57
2.	The Petition Fails to Identify a Legally Sufficient Motivation For Modifying Hellman’s Update Unit 36 or Authorization and Billing Unit 13 to Be Software-Based	59
3.	The Petition Improperly Conflates the Memory of the BIOS and the Verification Structure, Which the Claim Requires to be Separate.....	62

D.	Dependent Claims 3, 8, 9, 14: The Allegedly Combined System Proposed in the Petition Would Not Function for its Intended Purpose	64
E.	Dependent Claims 2, 6–7, 10–13 Are Not Obvious For All of the Same Reasons as Claim 1	66
F.	Objective Evidence Establishes Non-Obviousness of the Claimed Invention	66
1.	A Leading BIOS Company Praised the Claimed Invention and Agreed to Commercialize	66
2.	Additional Licenses Establish the Significance of the Claimed Invention.....	69
V.	CONCLUSION	71

TABLE OF AUTHORITIES

	<u>Page</u>
Cases	
<i>Activevideo Networks, Inc. v. Verizon Commc'ns, Inc.</i> , 694 F.3d 1312 (Fed. Cir. 2012)	51
<i>Akzo N.V. v. United States ITC</i> , 808 F.2d 1471 (Fed. Cir. 1986)	50
<i>Ancora Techs., Inc. v. Apple, Inc.</i> , 744 F.3d 732 (Fed. Cir. 2014)	27, 35, 36, 70
<i>Ancora Techs., Inc. v. HTC Am., Inc.</i> , 908 F.3d 1343 (Fed. Cir. 2018)	27, 28
<i>Apple Inc. v. Samsung Elecs. Co., Ltd.</i> , 839 F.3d 1034 (Fed. Cir. 2016)	66, 67
<i>Bosch Automotive Service Solutions, LLC v. Matal</i> , 878 F.3d 1027 (Fed. Cir. 2017)	68
<i>Comcast Cable Commc'ns, LLC v. Rovi Guides, Inc.</i> , IPR2019-00299, Paper 45 (PTAB June 29, 2020)	52
<i>Custom Accessories, Inc. v. Jeffrey-Allan Indus., Inc.</i> , 807 F.2d 955 (Fed. Cir. 1986)	66
<i>Fox Factory Inc. v. SRAM LLC</i> , 944 F.3d 1366 (Fed. Cir. 2019)	67
<i>Graham v. John Deere Co. of Kansas City</i> , 383 U.S. 1 (1966)	50
<i>Henny Penny Corp. v. Frymaster LLC</i> , 938 F.3d 1324 (Fed. Cir. 2019)	67
<i>Impax Laboratories Inc. v. Lannett Holdings Inc.</i> , 893 F.3d 1372 (Fed. Cir. 2018)	69
<i>In re Dow Chem. Co.</i> , 837 F.2d 469 (Fed. Cir. 1988)	51, 62
<i>In re Evanega</i> , 829 F.2d 1110 (Fed. Cir. 1987)	50
<i>In re Gordon</i> , 733 F.2d 900 (Fed. Cir. 1984)	51
<i>In re Huang</i> , 100 F.3d 135 (Fed. Cir. 1996)	67
<i>In re NTP, Inc.</i> , 654 F.3d 1279 (Fed. Cir. 2011)	50, 61, 62

IPR2021-01338: Patent Owner Response

<i>In re NuVasive, Inc.</i> , 842 F.3d 1376 (Fed. Cir. 2016)	51, 59
<i>In re Young</i> , 927 F.2d 588 (Fed. Cir. 1991)	50
<i>Kaken Pharma. Co., Ltd. v. Iancu</i> , 952 F.3d 1346 (Fed. Cir. 2020)	passim
<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	49, 50, 51
<i>Monarch Knitting Mach. Corp. v. Sulzer Morat GmbH</i> , 139 F.3d 877 (Fed. Cir. 1998)	51
<i>Personal Web Techs., LLC v. Apple, Inc.</i> , 848 F.3d 987 (Fed. Cir. 2017)	50, 61
<i>Polaris Indus., Inc. v. Arctic Cat, Inc.</i> , 882 F.3d 1056 (Fed. Cir. 2018)	67
<i>Power Integrations, Inc. v. Fairchild Semiconductor Int’l, Inc.</i> , 711 F.3d 1348 (Fed. Cir. 2013)	66
<i>Rembrandt Wireless Techs., LP v. Samsung Elecs. Co., Ltd.</i> , 853 F.3d 1370 (Fed. Cir. 2017)	31, 33, 34
<i>Sony Mobile Commc’ns AB v. Ancora Technologies, Inc.</i> , IPR2021-00663, Paper 13 (April 23, 2021)	57
<i>Star Scientific, Inc. v. R.J. Reynolds Tobacco Co.</i> , 655 F.3d 1364 (Fed. Cir. 2011)	50
<i>Transocean Offshore Deepwater Drilling, Inc. v. Maersk Drilling USA, Inc.</i> , 699 F.3d 1340 (Fed. Cir. 2012)	69
<i>Vitronics Corp. v. Conceptronic, Inc.</i> , 90 F.3d 1576 (Fed. Cir. 1996)	30, 32
Regulations	
37 C.F.R. § 42.65(a)	52
37 C.F.R. 42.100(b)	30, 33

LIST OF EXHIBITS

EXHIBIT NO.	TITLE
1001	U.S. Patent No. 6,411,941 to Mullor et al. (“’941 Patent”)
1002	Image File Wrapper of U.S. Patent No. 6,411,941 (“File History”)
1003	Declaration of Andrew Wolfe, Ph.D. (“Wolfe Decl.”)
1004	U.S. Patent No. 4,658,093 (“Hellman”)
1005	U.S. Patent No. 5,892,906 (“Chou”)
1006	U.S. Patent No. 5,933,498 (“Schneck”)
1007	Reserved
1008	Reserved
1009	Reserved
1010	Reserved
1011	Claim Construction Order, <i>Ancora Techs., Inc. v. Apple Inc.</i> , No. 4:11-cv-06357 (N.D. Cal. Dec. 31, 2012) (ECF No. 107).
1012	Final Claim Constructions of the Court, <i>Ancora Techs., Inc. v. LG Elecs., Inc.</i> , No. 1:20-cv-00034 (W.D. Tex. June 2, 2020) (ECF No. 69).
1013	Supplemental Claim Construction Order, <i>Ancora Techs., Inc. v. LG Elecs., Inc.</i> , No. 1:20-cv-00034 (W.D. Tex. Aug. 19, 2020) (ECF No. 93).
1014	Civil Minutes re Telephonic Markman Hearing, <i>Ancora Techs., Inc. v. TCT Mobile (US), Inc.</i> , No. 8:19-cv-02192 (C.D. Cal. Nov. 12, 2020) (ECF No. 66) (attaching “The Court’s Final Ruling on Claim Construction (<i>Markman</i>) Hearing,” but also ordering further meet and confer on subject).
1015	Civil Minutes re Telephonic Markman Hearing, <i>Ancora Techs., Inc. v. TCT Mobile (US), Inc.</i> , No. 8:19-cv-02192 (C.D. Cal. Nov. 19, 2020) (ECF No. 69) (confirming no change to “The Court’s Final Ruling on Claim Construction (<i>Markman</i>) Hearing”).
1016	Decision Granting Institution of Inter Partes Review, <i>TCT Mobile (US) Inc. v. Ancora Technologies, Inc.</i> , No. IPR2020-01609 (Feb. 16, 2021) (Paper No. 7) (“TCL Institution Decision”).
1017	Decision Granting Institution of Inter Partes Review, <i>Sony Mobile Commc’ns AB v. Ancora Technologies, Inc.</i> , No.

	IPR2021-00663 (June 10, 2021) (Paper No. 17) (“Sony Institution Decision”).
2001	Microsoft Corporation’s Request for <i>Ex Parte</i> Reexamination Image File Wrapper, Control No. 90010560
2002	<i>Ancora Techs., Inc. v. Apple, Inc.</i> , 744 F.3d 732 (Fed. Cir. 2014)
2003	Deposition Excerpts of Jon Weissman, <i>Ancora Techs., Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919 (W.D. Wa. Sep. 9, 2019)
2004	Declaration of Ian Jestice, <i>Ancora Techs., Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919 (W.D. Wa. Aug. 26, 2019)
2005	Brief of Appellees HTC America, Inc. and HTC Corporation, <i>Ancora Techs., Inc. v. HTC America, Inc., HTC Corp.</i> , Case No. 18-1404 (Fed. Cir. Apr. 23, 2018)
2006	Declaration of Jon Weissman, <i>Ancora Techs., Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919 (W.D. Wa. Sep. 4, 2019)
2007	Terplan, Kornel, Morreale, Patricia, THE TELECOMMUNICATIONS HANDBOOK, CRC Press, 2000
2008	COMPUTER USER’S DICTIONARY, Microsoft Press, 1998
2009	MICROSOFT COMPUTER DICTIONARY FOURTH EDITION, Microsoft Press, 1999
2010	PC MAGAZINE ENCYCLOPEDIA, available at https://www.pcmag.com/encyclopedia (excerpt, definition of “Agent”)
2011	U.S. Patent No. 6,411,941 File History with Beeble White Paper
2012	Joint Claim Construction Chart, <i>Ancora Techs., Inc. v. TCT Mobile (US) Inc., Huizhou TCL Mobile Commc’n Co., Ltd., and Shenzhen TCL Creative Cloud Tech. Co., Ltd.</i> , Case No. 8:19-cv-02192 (Dkt. #49, 49-1, 49-2)
2013	Declaration of Dr. David Martin, Ph.D., <i>Sony Mobile Commc’ns AB, Sony Mobile Commc’ns, Inc., Sony Elecs. Inc., and Sony Corp. v. Ancora Techs., Inc.</i> , IPR2021-00663, Ex. 2015 (PTAB Apr. 23, 2021)
2014	U.S. Patent No. 6,189,146 (Misra)
2015	U.S. Patent No. 5,479,639 (Ewertz)
2016	Decision Denying Institution of Inter Partes Review, <i>Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. v. Ancora Technologies, Inc.</i> , IPR2020-01184

IPR2021-01338: Patent Owner Response

2017	Declaration of Ian Jestice, <i>Ancora Technologies Inc. v. LG Electronics Inc., LG Electronics U.S.A. Inc., Samsung Electronics Co., Ltd., and Samsung Electronics America, Inc.</i> , Case No. 1:20-cv-00034 (Dkt. # 44-8)
2018	Declaration of Dr. David Martin (May 3, 2022)
2019	U.S. Patent No. 5,892,900 (Ginter)
2020	U.S. Patent No. 5,734,819 (Lewis)
2021	U.S. Patent No. 6,153,835 (Schwartz)
2022	<i>Ancora Techs. Inc. v. Apple, inc.</i> , Case No. 4:11-cv-06357 (Dkt. # 171-3) [Apple Inc.’s N.D. Cal. L.R. 3-3 (Invalidity) Disclosures]
2023	Petition, <i>HTC Corp. v. Ancora Techs. Inc.</i> , Case No. CBM2017-00054, Paper 1 (PTAB May 26, 2017)
2024	Institution Decision, <i>HTC Corp. v. Ancora Techs. Inc.</i> , Case No. CBM2017-00054, Paper 7 (PTAB Dec. 1, 2017)
2025	Croucher, “The BIOS Companion” (1997) (Excerpts)
2026	Transcript of Deposition of Dr. Andrew Wolfe (April 22, 2022)
2027	Joint News Release (February 14, 2005)
2028	March 25, 2022 Email from Board regarding IPR2021-01338 and IPR2021-01406
2029	[REDACTED] (Confidential)
2030	Declaration of Miki Mullor
2031	[REDACTED] (Confidential)
2032	[REDACTED] (Confidential)
2033	Reserved
2034	Reserved
2035	AIPLA statistics
2036	Reserved
2037	[REDACTED] (Confidential)
2038	Proposed Protective Order
2039	Redline Relative To Standard Protective Order
2040	Reserved
2041	Reserved
2042	Barron’s Dictionary of Computing and Internet Terms (5th Ed. 1996)

IPR2021-01338: Patent Owner Response

2043	Decision Granting Institution, Case No. IPR2020-01609, Paper 7 (PTAB Feb. 16, 2021)
2044	Free On-Line Dictionary of Computing (June 6, 1999), available at http://foldoc.org/bios
2045	U.S. Patent No. 5,684,951 (Goldman)
2046	Paul C. Kocher, <i>Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems</i> , Proceedings of 16 th Annual Int'l Cryptology Conf. (Aug. 18–22, 1996)

I. INTRODUCTION

Petitioners mischaracterize the '941 patent repeatedly by contending that the invention consists merely of storing a license record for a program in the BIOS memory. Petitioners ignore statements by both the applicants and the examiners in the '941 patent's prosecution history, distinguishing the cited art because it did not teach programs running at the OS level interacting with a program verification structure stored in the BIOS. These statements applied to all of the cited art during prosecution, through reexamination, and in multiple challenges to the validity of the '941 patent at the USPTO, district court, and the Federal Circuit. The references asserted in the Petition have the same shortcomings as the references already addressed during these earlier proceedings.

Because they ignore this prosecution and reexamination history, Petitioners do not apply the correct interpretation of the challenged claims. Interpreted properly, Petitioners fail identify any teaching in the art that would have rendered the claimed "agent" obvious. Neither Hellman, Chou, or Schneck; nor the Petitioners' combinations disclose programs running at the OS level interacting with a program verification structure stored in the BIOS.

Petitioners also fail to identify adequate evidence of a motivation to combine Hellman with Chou, and ignore important details in Hellman that render Hellman's

disclosed system unsuitable for its intended purpose when modified as Petitioners have proposed.

The non-obviousness of the challenged claims is further confirmed by objective evidence including industry praise and significant licensing related to the '941 patent. In particular, Patent Owner has licensed the '941 patent to multiple companies that asserted the very same prior art combinations in district court and the USPTO. This licensing success can only be attributed to the patent's strength over the asserted grounds and every other prior art reference known to the licensees.

II. SUMMARY OF THE '941 PATENT

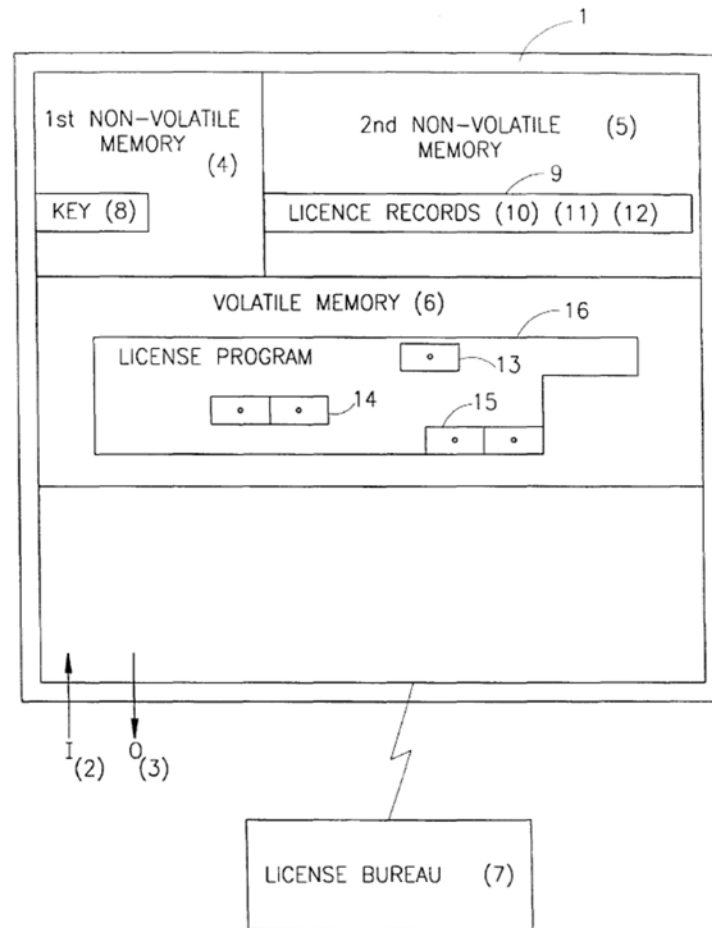
A. The Inventions Disclosed in the '941 Patent

The '941 patent discloses a specific technique for “identifying and restricting an unauthorized software program’s operation.” (Ex.1001 1:6–8.) It describes “illegally copied software,” which “represents billions of dollars in lost profits to commercial software developers.” (*Id.* at 1:14–18.) The '941 patent describes two known methods of verifying and restricting a software program on a computer: first involving software-based products, and second involving hardware-based products. (*Id.* 1:19–32.)

The '941 patent identifies problems with both. First, the software-based products “validate authorized software usage by writing a license signature onto the computer’s volatile memory (e.g., hard disk).” (*Id.* 1:19–21.) The license signature,

however, is vulnerable to “hackers” and to “physical instabilities” of the hardware. (*Id.* 1:21–26.) Second, the hardware-based products required “a dongle that is coupled e.g. to the parallel port of the P.C.” (*Id.* 1:27–29.) The dongles, however, are “expensive, inconvenient, and not particularly suitable for software that may be sold by downloading (e.g., over the internet).” (*Id.* 1:29–32.)

In contrast to these systems, the '941 patent describes using existing computer hardware with “a conventional BIOS module in which a key was embedded at the ROM section thereof.” (*Id.* 1:45–47.) The BIOS module further includes “another (second) non-volatile section of the BIOS, e.g., E²PROM (or the ROM)” in which “memory may optionally be erased or modified (using E²PROM manipulation commands).” (*Id.* 1:66–2:4.) FIG. 1 illustrates a first non-volatile memory 4 (e.g., ROM section of the BIOS) including a key 8, and a second non-volatile memory 5 (e.g., E²PROM section of the BIOS) with a license-record area 9 that includes license records 10–12. (Ex. 1001 5:10–27.)



(Ex. 1001 FIG. 1.) BIOS memory space of this type was and is typically used for storing programs that assist in the start-up of a computer. (Ex. 2018 ¶62.) The '941 patent explains that two-part BIOS modules (ROM + E²PROM) were known. (Ex. 1001 3:21–24 (“a processor normally includes a first non-volatile memory, a second non-volatile memory, and data linkage access to a volatile memory.”); *see also* Ex. 2018 ¶63.)

A license record is created in the second non-volatile section of the BIOS during “an initial license establishment procedure” that sets up a “verification structure” using E²PROM manipulation commands that enable the method to “add,

modify or remove licenses” in the BIOS. (*See* Ex. 1001 1:33–2:9.) Separately, the “key” stored in the BIOS is “a unique identification code for the host computer,” which is embedded “during manufacture” and “cannot be removed or modified.” (*Id.* 1:46–52.) In one example, setting up the verification structure includes “establishing at least one license-record location in the first or second nonvolatile memory area.” (*Id.* 6:20–21.) License records can then be created and stored in the verification structure. With reference to FIG. 1, license record fields (13–15) appended to a licensed program 16 are encrypted to form license records (10–12). (*Id.* 5:27–43.) The key stored in the ROM portion of the BIOS is used to encrypt the license records, such that the license record is rendered useless if copied to a second computer with a different key. (*See id.* 2:27–59; 6:22–26, Ex. 2018 ¶64.) The encrypted license records are then established in the license record locations (*e.g.*, 10–12 in FIG. 1). (Ex. 1001 6:26–28.)

Verifying the license requires a license verifier application, which “accesses the program under question, retrieves therefrom the license record, encrypts the record utilizing the specified unique key (as retrieved from the ROM section of the BIOS) and compares the so encrypted record to the encrypted records that reside in the E²PROM.” (*Id.* 2:10–19; 6:29–39.) The program is allowed to run if the encrypted records match or halted if they do not. (*Id.* 2:19–26; 6:40–52.)

The ’941 patent explains the advantages of the disclosed method:

An important advantage in utilizing non-volatile memory such as that residing in the BIOS is that the required level of system programming expertise that is necessary to intercept or modify commands, interacting with the BIOS, is substantially higher than those needed for tampering with data residing in volatile memory such as hard disk. Furthermore, there is a much higher cost to the programmer, if his tampering is unsuccessful, i.e. if data residing in the BIOS (which is necessary for the computer's operability) is inadvertently changed by the hacker. This is too high of a risk for the ordinary software hacker to pay. Note that various recognized means for hindering the professional-like hacker may also be utilized (e.g. anti-debuggers, etc.) in conjunction with the present invention.

(Ex. 1001 3:4–17.)

B. The Challenged Claims

Petitioners assert that claims 1–3, 6–14, and 16 were obvious. Each depends from claim 1, which recites:

1. A method of restricting software operation within a license for use with a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area; the method comprising the steps of:

selecting a program residing in the volatile memory,

using an agent to set up a *verification structure* in the *erasable, non-volatile memory of the BIOS*, the verification structure accommodating *data that includes at least one license record*,

verifying the program using at least the verification structure from the erasable non-volatile memory of the BIOS, and

acting on the program according to the verification.

(Ex. 1001 6:59–7:4.)

C. Prosecution History

The '941 patent issued from Application No. 09/164,777, filed on October 1, 1998, and claimed priority to Israeli Application No. 124571, filed on May 21, 1998.

1. Non-Final Office Action, December 20, 2000

In a first office action, the Examiner rejected the then-pending claims over U.S. Patent No. 5,892,900 by Ginter et al. (hereinafter “Ginter”) and U.S. Patent No. 5,684,951 by Goldman et al. (hereinafter “Goldman”). (Ex. 2011 ANCC000077–83.) The examiner relied primarily on Ginter, asserting that Ginter anticipated many of the pending claims. (*Id.* ANCC000077–79.) For certain dependent claims, the examiner asserted that the pending claims were obvious based on Ginter modified with Goldman’s disclosure of pseudo unique keys. (*Id.* ANCC000080–83.)

a) Ginter

Ginter discloses “a distributed virtual distribution environment (VDE) that may enforce a secure chain of handling and control, for example, to control and/or meter or otherwise monitor use of electronically stored or disseminated information.” (Ex. 2019 Abstract; *see also id.* 2:20–32, 6:29–67.)

Ginter discloses an “electronic appliance” with a secure processing unit (SPU) capable of operating within the VDE. (*Id.* 60:7–15; *see also id.* 3:67–4:4, 20:47–59.) Ginter explains that “tamper resistance and concealment of VDE control process execution and related data storage activities” is used to improve the overall security of the VDE system. (*Id.* 21:22–33.) In an example illustrated in FIG. 8, Ginter describes the electronic appliance as a computer, comprising one or more SPU 500. (*Id.* 62:31–63, 63:67–65:15.) Alternatively, “SPU 500 may be integrated together with one or more other CPU(s) (e.g., a CPU 654 of an electronic appliance) in a single component or package.” (*Id.* 65:21–24.) The SPU, as illustrated in Ginter’s FIG. 9, “may comprise a combination of a masked **ROM 532a** and an **EEPROM and/or equivalent “flash” memory 532b.**” (*Id.* 70:39–71:15, FIG. 9.)

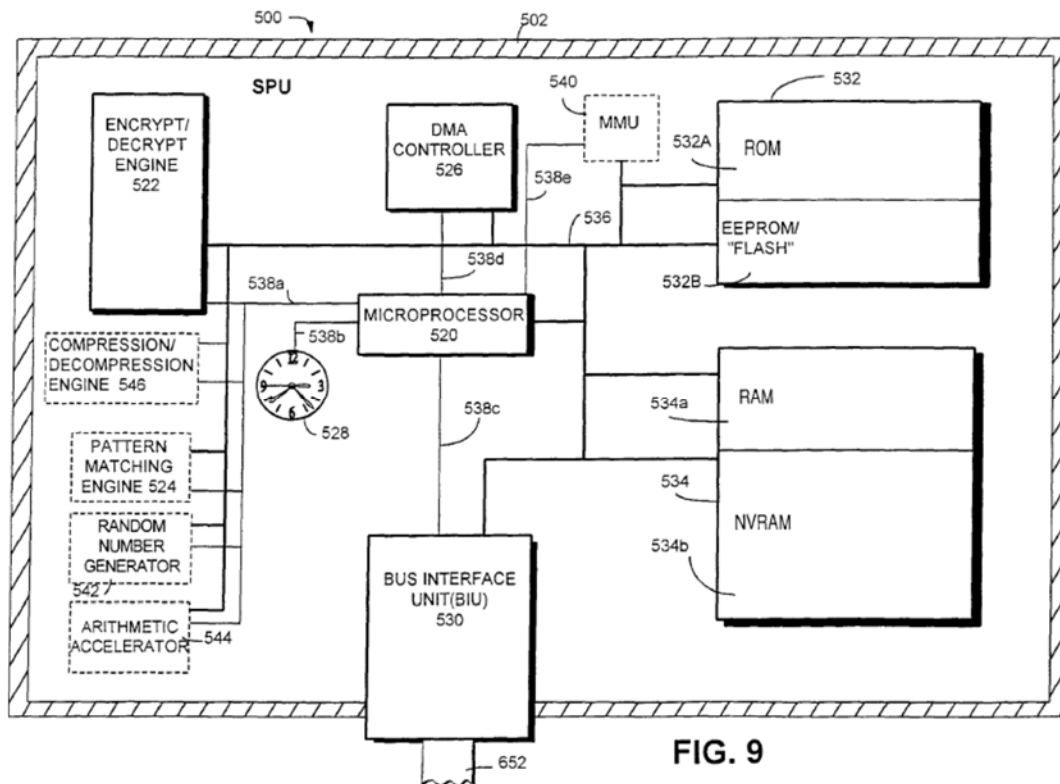


FIG. 9

(*Id.* FIG. 9.) Separately, Ginter discloses “cryptography employing keys and/or authentication values hidden in *normally inaccessible locations* in the appliance 600,” including: “[n]on-volatile, writable, storage in the appliance or its components, such as that used for . . . *standard BIOS software*, etc.” (Ex. 2011 ANCC000111 (citing Ex. 2019 245:55–246:24).) Ginter does not disclose or suggest an OS-level agent capable of setting up a verification structure in memory of the BIOS.

b) Goldman

Goldman discloses a system for “user validation with respect to a multi-user computer system.” (Ex. 2045 1:8–10.) In this system, a computer system 112 is connected via an internet interface 110 to an application system 310. (*Id.* FIG. 1,

4:59–5:4.) As shown in Figure 3A, the system provides a user validation system 310a communicating with the computer system 112. (*Id.* 5:34–40.)

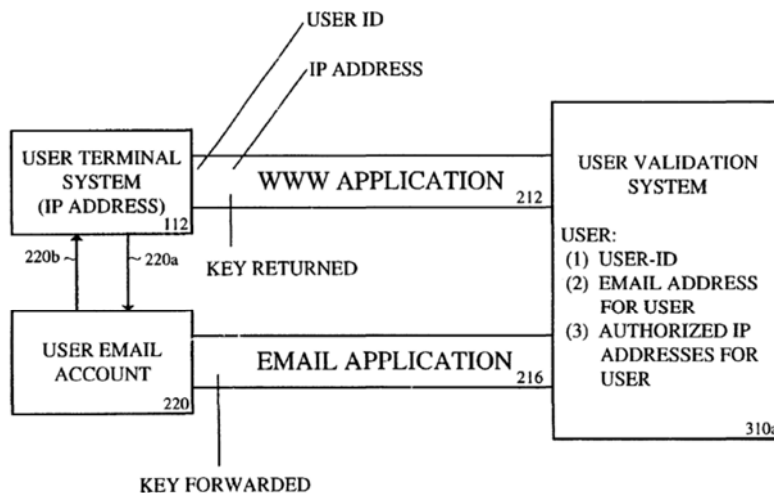


FIG. 3A

(*Id.* FIG. 3A.) In certain situations, the validation system generates a pseudo unique key from a userID, current IP address, and a secret code, which can be used to verify the user when the user is logged in from a new IP address. (*Id.* Abstract, 8:42–54, 9:26–33.)

2. Amendment, May 23, 2001

Applicants initially contended that Ginter does not disclose “setting up a verification structure and verifying the program using the verification structure as recited in the rejected claims.” (Ex. 2011 ANCC000093.)

3. Final Office Action, June 22, 2001

In a second office action, the Examiner maintained the same rejections (Ex. 2011 ANCC000110–16), disagreeing with applicant’s argument that Ginter does not

teach “setting up a verification structure and verifying the program using the verification structure” (Ex. 2011 ANCC000106).

4. Amendment and Request for Continued Examination, November 16, 2001

Applicants amended claim 1 to recite “using an agent to setting up verification structure in the ~~second~~ erasable, non-volatile memory of the BIOS, the ~~verification~~ verification structure accommodatinges data that includes at least one license record.” (Ex. 2011 ANCC000130.) The response further noted that “a description of a specific embodiment of the invention is attached hereto.” (*Id.* ANC000129.) The file history includes a white paper, authored by the inventors of the ’941 patent. (*See id.* ANC000178–85.) The white paper explains that, “[i]n order for a user to access the BIOS E2PROM[,] *proprietary software*” needed to be developed. (*Id.* ANC000180 (emphasis added).) The document explains that the “Beeble License Manager is a windows control panel applet that allows users to self manage Beeble licenses installed on their machine.” (*Id.* ANC000182.) The control panel applet was responsible for “installation and removal of Beeble licenses.” (*Id.*) Skilled artisans would recognize the described control panel applet as an example of an OS-level software agent. (Ex. 2018 ¶88.)

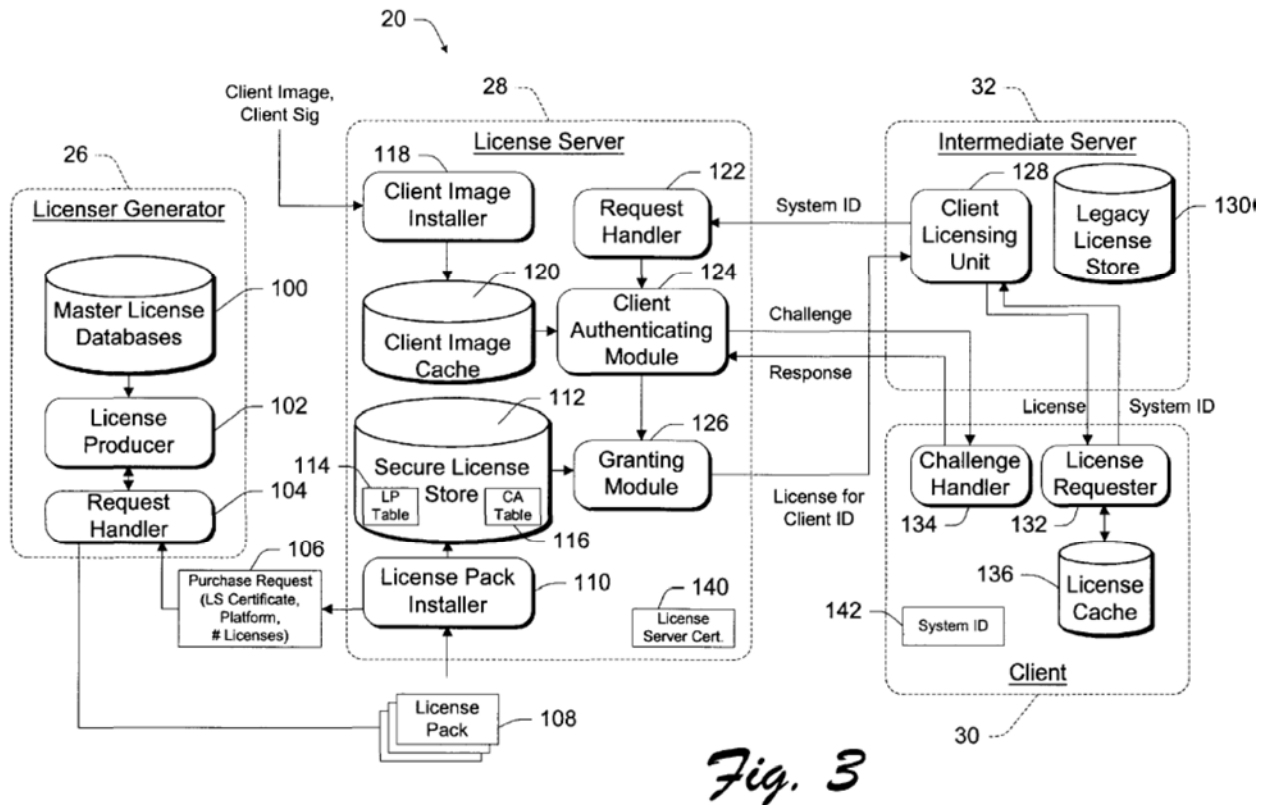
5. Non-Final Office Action, January 15, 2002

In a third office action, the Examiner relied on new references, U.S. Patent No. 6,189,146 to Misra et al. (hereinafter “Misra”) and U.S. Patent No. 5,479,639 to

Ewertz et al. (hereinafter “Ewertz”). (*See* Ex. 2011 ANC000143–45.) The examiner rejected all of the then-pending claims as obvious over Misra, Goldman, and Ewertz. (Ex. 2011 ANCC000143–45.) The examiner relied primarily on Misra, equating “programs” disclosed by Misra to the claimed agent. According to the examiner, “Misra teaches . . . programs (“agents”) used in the license collation process that belong to various parties.” (*Id.* ANCC000144–45.) The examiner asserted that Misra would have been modified in view of Ewertz’s teaching of expanding BIOS memory to store identification or configuration data such as software licenses. (*Id.*)

a) Misra

Misra discloses “a system and method for enforcing software licenses.” (Ex. 2014 1:7–8.) Misra’s architecture included four components: license generator 26, license server 28, intermediate server 32, and client 30, as illustrated in FIG. 3. (*Id.* 6:21–24.)



(*Id.* FIG. 3.) Misra explains that the clients 30 can be “many different kinds of computers, including a desktop personal computer, a workstation, a laptop, a notebook computer, a handheld PC, and so forth;” or a “terminal device, which is a low cost machine.” (*Id.* 5:13–22.) As shown in FIG. 3, the client 30 includes a client system ID 142 that “is a unique identifier of the client computer” and based on information such as hard disk volume numbers, network cards, registered software, video cards, microprocessor identifiers, total RAM, and floppy disk drive configuration. (*Id.* 12:50–63.) Misra also describes a license ID, which is a unique identifier assigned to a software license when the software license is issued to a client device. (*Id.* 11:9–12.) The license ID may be a digital certificate indicating the right

to use the particular software at issue. (*Id.* 10:60–67.)

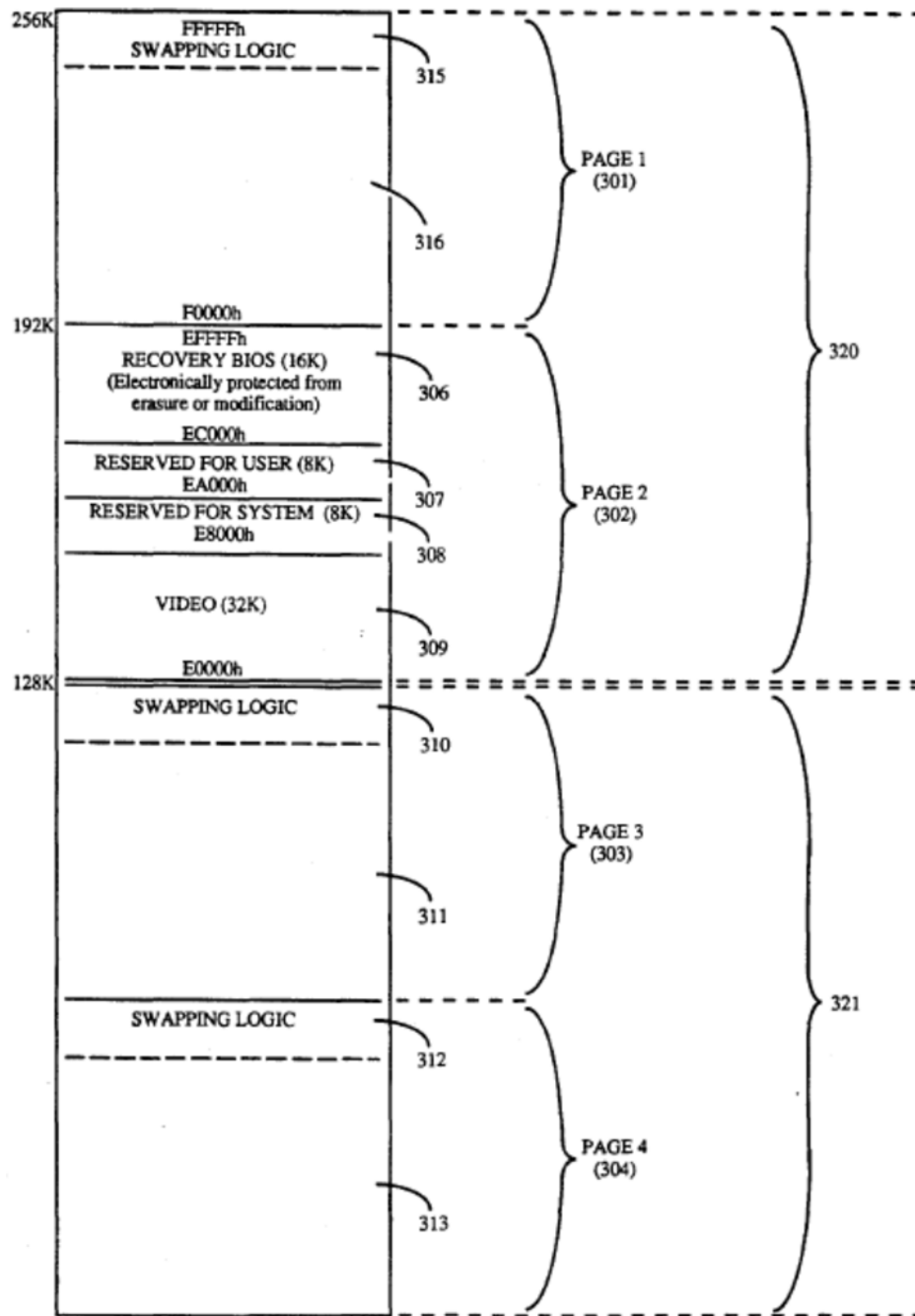
Misra does not, however, teach using the BIOS of a computer to store the license ID. (Ex. 2011 ANCC000145 (Examiner noted that Misra does not teach “constructing license records within a computer BIOS”).)

b) Ewertz

Ewertz discloses “a paging technique . . . used to expand the usable non-volatile memory capacity beyond a fixed address space limitation.” (Ex. 2015 2:24–26.) Ewertz explains that “ROM devices with a BIOS contained therein are typically constrained to a specific address range within the address space available.” (*Id.* 1:56–59, 1:62–67; *see also id.* 5:15–17.) According to Ewertz, developments at the time made it “increasingly unfeasible to fit all desired BIOS features within the 128K boundary of the IBM PC AT architecture.” (*Id.* 2:6–12.)

Ewertz illustrates the preferred embodiment of a paged *flash* memory 103. (E.g., *Id.* 4:33–35.)

FIGURE 2



(*Id.* FIG. 2.) Ewertz describes the contents of the 128K region 320 as follows:

The upper region 301 is used for storage of the normal system BIOS while the lower region 302 is used for storage of other logic and data

such as overflow BIOS code and/or data, video or other BIOS's, set-up code or data, and *other information* or logic.

(*Id.* 5:20–25 (emphasis added).) Ewertz further describes a technique for swapping certain “pages” of the flash memory. (*Id.* 3:3–5, 5:30–38.) This approach would have had at least two advantages recognizable to persons skilled in the art. (Ex. 2018 ¶¶99; Ex. 2015 12:8–14, 10:14–20.)

Ewertz also discloses processing logic for updating the flash memory device. (*Id.* FIG. 8, 10:21–26.) Ewertz explains, however, that this processing logic “resides in the system BIOS.” (*Id.* 10:26–30.) Doing so allows the update routines “to accommodate hardware specific operations with non-volatile memory in particular computer systems.” (*Id.* 10:45–48.) For example, Ewertz discloses that identification information including software license numbers “may be stored in a page of non-volatile memory.” (*Id.* 3:15–22.) A PHOSITA reading Ewertz would understand that writing to the BIOS, as shown in Ewertz’s Figure 8, would have been done using software running as part of the BIOS code. (Ex. 2018 ¶¶101.) Ewertz explains that the steps in Figure 8 are started when the BIOS is running. (*Id.*) After block 812 at the end of the procedure, “[n]ormal BIOS processing” is resumed. (Ex. 2015 11:12–13.)

6. Amendment, February 5, 2002

Applicants traversed the rejection,¹ emphasizing that the claimed invention “proceed[ed] against conventional wisdom in the art” because:

BIOS is a configuration utility. Software license management applications, *such as the one of the present invention, are operating system (OS) level programs*. Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, *BIOS and OS level programs are normally mutually exclusive*.

(Ex. 2011 ANCC000153 (emphasis added).) Applicants further noted that “the systems described in Misra and Ewertz are an OS program and a BIOS program, respectively.” (*Id.* ANCC000154.)

Applicants also argued that the invention “proceeds against conventional wisdom in the art” because “[t]he BIOS area is not considered a storage area for computer applications.” (*Id.* ANCC000153–54.) Applicants identified two reasons why the PHOSITA would not look to the BIOS memory as a storage medium:

¹ The applicant also amended the claims that issued as claims 12 and 18 to address informalities.

First, OS does not support this functionality and is not recognized as a hardware device like other peripherals. . . . There is no OS support whatsoever to write data to the system BIOS. Therefore, an ordinary person skilled in the art would not consider the BIOS as a possible storage medium. Furthermore, it is common that all peripheral devices in the PC are listed and recognized by the OS except for the BIOS. This supports the fact that the BIOS is not considered a peripheral device. Accordingly, an ordinary person skilled in the art would not consider the BIOS for any operation, including writing to the BIOS.

Second, no file system is associated with the BIOS. Every writable device connected to the PC is associated with an OS file system to arrange and manage data structures. . . . No such file system is associated with the BIOS. This is further evidence that OS level application programmers would not consider the BIOS as a storage medium for license data.

(*Id.*) In short, because of the prior art status quo of “mutual[] exclusiv[ity]” between BIOS and OS level programs, and lack of interaction from the OS to information stored in the BIOS, “[u]sing BIOS to store application data such as that stored in Misra’s local cache for licenses [wa]s not obvious.” (*Id.*)

7. Notice of Allowability, March 28, 2002

The examiner allowed the claims, explaining that they recited a program (i.e.,

the claimed agent) running at the “OS level” interacting with a verification structure in the BIOS—an approach that was never previously contemplated or disclosed in the art cited during examination. The examiner explained in the “Reasons for Allowance”:

[T]he key distinction between the present invention and the closest prior art, is that the Misra et al., and Ginter et al. systems and the Ewertz et al. system run at the operating system level and BIOS level, respectively. More specifically, the closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level interacting with a program verification structure stored in the BIOS to verify the program using the verification structure and having a user act on the program according to the verification. Further, it is well known to those of ordinary skill of the art that a computer BIOS is not setup to manage a software license verification structure. The present invention overcomes this difficulty by using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.

(Ex. 2011 ANCC000162.) In short, the term “agent” was added during prosecution to specifically elucidate that “the closest prior art systems, singly or collectively, do not teach licensed programs *running at the OS level* interacting with a program verification structure stored in BIOS.” (*Id.* (emphasis added).)

D. Prior Invalidity Challenges

The '941 patent has withstood validity challenges in multiple hotly litigated cases since 2008 including serial challenges before the Board, in the federal district courts, and at the Federal Circuit. In all cases, the cited art does not teach or suggest programs *running at the OS level* interacting with a program verification structure stored in BIOS.

1. Microsoft's Reexamination No. 90/010,560

In 2009, Ancora defended the '941 patent in *Ex Parte* Reexamination No. 90/010,560. The Examiner found claims 1–19 patentable without amendment. (Ex. 1001 9 (*ex parte* reexamination certificate).)

a) Request for Reexamination, May 28, 2009

On May 28, 2009, Microsoft Corporation filed a request for reexamination of the '941 patent based on U.S. Patent No. 6,153,835 to Schwartz (hereinafter “Schwartz”) and U.S. Patent No. 5,734,819 to Lewis (hereinafter “Lewis”). (Ex. 2001 17–40, 112-151.)

(1) Lewis

Lewis discloses “a computer system having a non-volatile memory and a way of detecting when that information has been altered so as to prevent operation of the computer system once tampering has been detected.” (Ex. 2020 1:11–15.) Lewis describes license managers for computer systems, in which a unique software key

corresponds to the machine's serial number. (*Id.* 1:18–23.) Rather than storing the electronic serial number in programmable memory (*id.* 1:37–41), Lewis discloses a hardware device 16 with a chip ID register 18 for storing a unique chip ID (*id.* 4:33–34). The computer further includes a non-volatile memory (NVM) 20 that also stores the chip ID, along with other information about the device, as illustrated in Figure 2. (*Id.* 4:40–44.)

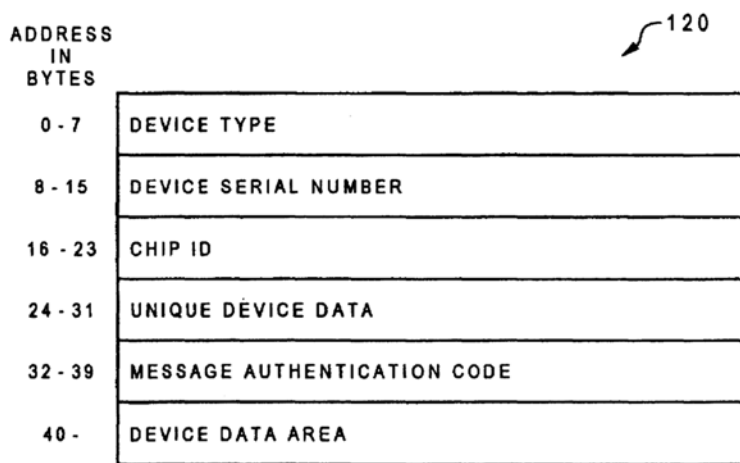


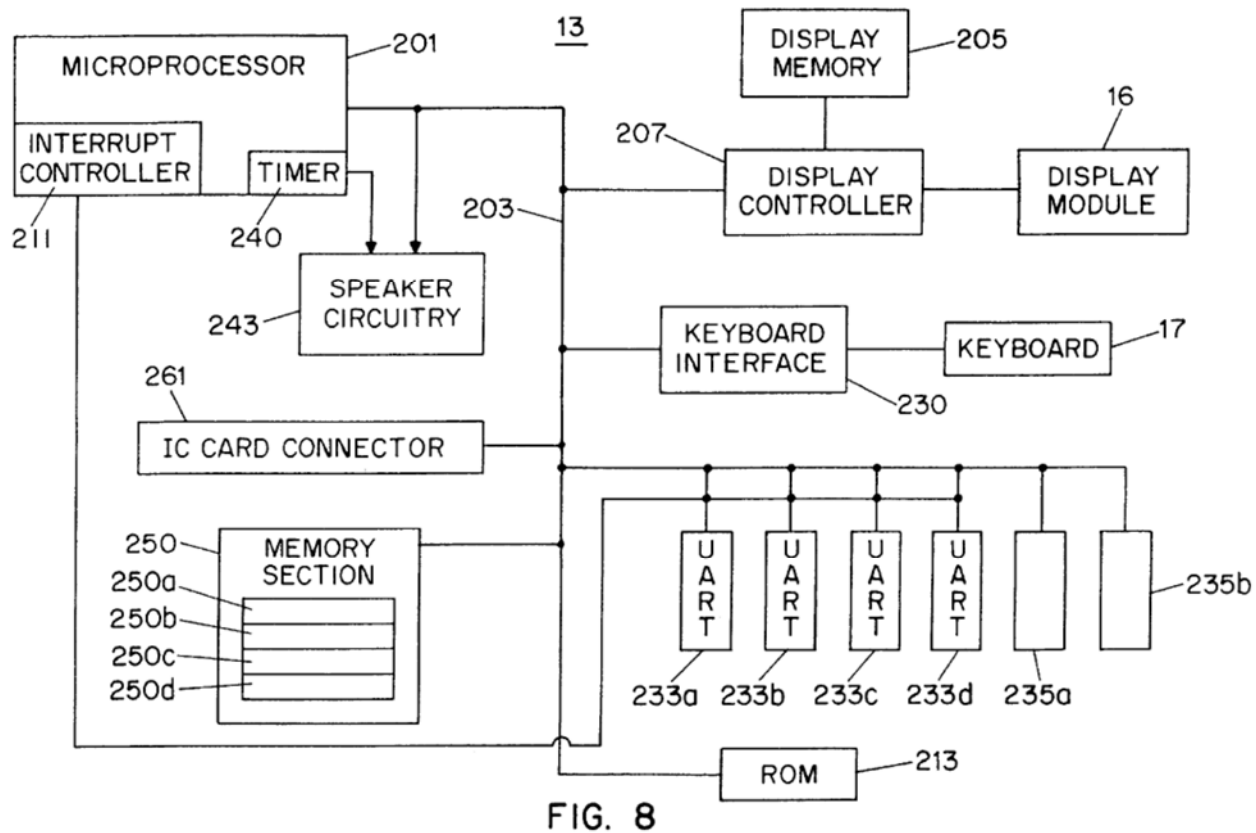
Fig. 2

(*Id.* FIG. 2.) Lewis discloses generating the message authentication code (MAC) shown in Figure 2, using other data from the non-volatile memory. (*Id.* 4:55–5:9.) The MAC is then stored in the non-volatile memory. (*Id.*)

“Prior to using device 16, the system code performs a chip identification and NVM content alteration detection test.” (*Id.* 5:28–30.) In short, the system uses the stored MAC to determine whether the NVM has been altered. (*Id.* 5:31–50.)

(2) Schwartz

Schwartz discloses “an electronic scale system and method which is particularly suitable for mailing or shipping use.” (Ex. 2021 1:25–27.) The disclosed scale system allows updating of postage rates and other shipping charges without necessarily replacing the memory inside the scale system, while also preventing or deterring “unauthorized copying of software.” (*Id.* 2:38–49.) Figure 8 illustrates the components.



(*Id.* FIG. 8.) The disclosed memory section 250 includes a 128 Kbyte flash electrically erasable programmable read-only memory (EEPROM) 250a, a 32 Kbyte EPROM 250b, a 128 Kbyte nonvolatile static random-access-memory (SRAM)

250c, and a 128 Kbyte SRAM 250d. (*Id.* 7:50–57.) Schwartz further illustrates the contents of the respective memories, in Figure 9.

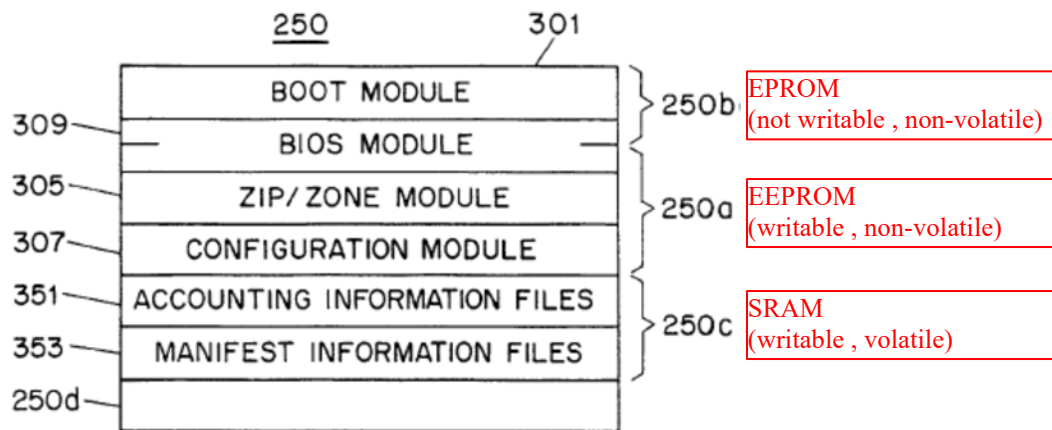


FIG. 9

(*Id.* FIG. 9 (annotations added); Ex. 2018 ¶114.) Schwartz explains that “part of BIOS module 309 [is] located in flash EEPROM 250a” and the “remaining part of BIOS module 309 and boot module 301 are located in EPROM 250b.” (Ex. 2020 8:17–20.)

In one embodiment, Schwartz discloses aspects related to loading rate schedule data from an IC card 401 to the flash EEPROM (*Id.* 9:34–65) and aspects related to loading new application code to the flash EEPROM (10:15–20). The user needs to enter a valid authorization number to enable the new application software, or other new data, or system options selected by the user. (*Id.* 10:21–25.) The authorization number is compared against a generated signature. (*Id.* 10:25–49, 11:58–12:14, FIG. 12.) If the authorization number matches the signature, the authorization number is stored in configuration module 307—separate from, but

within the same EEPROM 250a as the BIOS. (*Id.* 10:49–54.) The system performs a similar verification at each power up, comparing the authorization number stored in configuration module 307 with a generated signature. (*Id.* 11:24–40.)

b) Reexamination Order, August 3, 2009

On August 3, 2009, the examiner ordered reexamination of the '941 patent on the grounds based on Lewis stating that:

Lewis discloses the loading into system memory (volatile) of a program, for which an encryption code (a MAC) is constructed using a driver for an external device in nonvolatile RAM. *It is common in the art to implement such drivers in the BIOS area.* . . . The correlating of specific instantiations of programs to specific computers constitutes a *de facto license* for that computer to use the program. Since *the art cited during prosecution did not show such information being stored in and used from the memory of the BIOS*, it is agreed that a reasonable examiner would have found this reference important in determining the patentability of claims 1-19.

(Ex. 2001 164–65 (emphasis added).) The examiner found that the grounds based on Schwartz did not present a substantial new question of patentability, and did not order reexamination based on Schwartz:

Schwartz discloses a postage scale that may receive new programs and store licensing information in related to

these programs in non-volatile memory. See figure 9. The programs that provide this functionality, however, do not reside in BIOS; rather, they are instantiated as applications running on the operating system. It is therefore the case that the table created cannot be considered to be in BIOS either. *Schwartz is therefore merely cumulative to the art cited by the Examiner during prosecution*, insofar as it teaches to the claim limitations.

(Ex. 2001 at 165.)

**c) Notice of Intent to Issue Reexamination Certificate,
March 9, 2010**

Without issuing any rejections, the examiner concluded that Lewis did not anticipate or render claims 1–19 of the '941 patent obvious. According to the examiner,

Lewis discloses an invention that stores license information in non-volatile memory (which is the BIOS, since it is being setup and used by the system program) related to a system device Although the program is clearly associated with the device, the verification structure that is set up in non-volatile memory by Lewis is derived from a combination of non-functional descriptive material and information on the device itself, rather from the substance of the device driver, and is *only being used to verify the device itself* (or the information for the device written to non-volatile memory) and not the program that

drives the device. Lewis’ invention is not being used to verify the program (as per claim 1) or for verifying the application software program (as per claim 18), but rather *just the device* that the program is being used to access (see Lewis, column 5, lines 27-49).

(*Id.* 249–50 (emphasis added).)

As described above, Lewis discloses a solution to problems stemming from the fact that serial number data is stored in “a programmable memory” that is non-volatile. (*See* Ex. 2020 1:37–41.) Lewis further discloses writing an encryption code (a MAC) into the nonvolatile memory. (*See id.* 4:55–5:9.) The examiner reasoned that the driver writing the MAC to the non-volatile memory was part of the BIOS, because “[i]t is common in the art to implement such drivers in the BIOS area.” (*See* Ex. 2001 164.) Further, the examiner ordered reexamination because “art cited during prosecution did not show such information being *stored in and used from the memory of the BIOS.*” (*Id.* 165.) Nevertheless, the examiner found claims 1–19 over the art cited in the reexamination and in the prosecution history.

2. Apple Litigation

Ancora sued Apple for infringement of the ’941 patent on December 29, 2010. On appeal from the claim construction order, the Federal Circuit found the ’941 patent not indefinite. *Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732 (Fed. Cir.

2014) (Ex. 2002). On August 25, 2015 Apple publicly filed its invalidity contentions, including grounds based on Hellman and Chou. (Ex. 2022 at 2, 3, 31.)

3. HTC’s Failed Petition for CBM Review

Ancora sued HTC for infringement of the ’941 patent on December 15, 2016. HTC filed a petition for CBM review asserting invalidity under § 101 for lack of patentable subject matter, under § 112 for indefiniteness and lack of written description, and under § 103 for obviousness. (Ex. 2023 at 24–25.) The Board denied institution. (Ex. 2024 at 2.)

4. HTC’s Failed Patentable Subject Matter Challenge

HTC also challenged the ’941 patent in the district court by moving to dismiss for alleged failure to claim patent eligible subject matter. The Federal Circuit ultimately found the ’941 patent not invalid over these challenges. *Ancora Techs., Inc. v. HTC Am., Inc.*, 908 F.3d 1343 (Fed. Cir. 2018) (holding the claims are not invalid under § 101).

The Federal Circuit relied in part on the prosecution history (Ex. 2011 ANCC000162) to find that the ’941 Patent proceeded against the conventional wisdom in the art to do something “the closest prior art, singly or collectively,” never contemplated: using “programs running at the OS level interacting with a program verification structure stored in the BIOS to verify [a] program using the verification structure”—that is, “using an agent to set up a verification structure in the erasable,

non-volatile memory of the BIOS” as Claim 1 requires. *HTC*, 908 F.3d at 1348–49.

5. TCL’s IPR Petition Asserting Hellman and Chou and the “Me-Too” Challengers

On August 27, 2019, Ancora sued several TCL entities for infringement of the ’941 patent. On September 10, 2020, TCL filed its petition in IPR2020-01609, asserting grounds based on Hellman and Chou—four years after Apple publicly asserted the references against the ’941 patent. (*See* Ex. 2022 at 2, 3, 31 (Apple’s Aug. 25, 2015 invalidity contentions).) The Board’s institution decision in IPR2020-01609 precipitated a series of “copycat” challenges based on the same grounds and substantively identical expert declaration, including the current proceeding.

6. Samsung’s Failed Inter Partes Review Petition

Samsung filed an IPR petition that relied in part on Schwartz (discussed above) combined with an article by Yee.² The Board denied institution for discretionary reasons but noted that “our initial inspection of the merits on this preliminary record suggests Petitioner’s challenges contain certain weaknesses.” (Ex. 2016 at 24.) The Board highlighted differences between the claims relative to Schwartz:

In its Petition, Petitioner takes the position that Schwartz’s

² Yee, “Using Secure Coprocessors,” Carnegie-Mellon University, CMU-CS-94-149 (1994)

EEPROM 250a teaches the claimed “non-volatile memory of the BIOS” under the Federal Circuit’s interpretation in *Ancora v. Apple* “because it stores part of BIOS module 309.” However, other parts of Schwartz’s EEPROM 250a store configuration module 307 and zip/zone module 305. ***Petitioner does not allege that these other modules themselves are associated with the computer’s BIOS.*** Significantly, Petitioner does not explain adequately why the entirety of EEPROM 250a, including the memory space that stores configuration module 307 and zip/zone module 305, is a “non-volatile memory of BIOS.” Moreover, Petitioner admits that claim 1 “requires the ‘verification structure’ to . . . be stored in the ‘erasable, non-volatile memory of the BIOS,’” and that Schwartz’s “authorization number (and hence the electronic signature) is stored in configuration module 307,” not BIOS module 309.

(*Id.* at 26 (emphasis added, citations omitted).)

E. Level of Ordinary Skill in the Art

A PHOSITA, in context of the ’941 patent would be an individual who, as of the 1997/1998 time frame, had earned an accredited Bachelor’s degree in electrical engineering, computer engineering, or computer science, or an equivalent education or level of knowledge, and two years of industrial experience with computer hardware, software, and operating systems. (Ex. 2018 ¶51.) Differences between this

level of ordinary skill and the level proposed by Petitioners would not affect the disputed issues in this matter. (*See id.* ¶52.)

F. Claim Interpretation

Claims in an inter partes review “shall be construed using the same claim construction standard that would be used to construe the claim in a civil action . . . including 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.” 37 C.F.R. 42.100(b); *see Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc). Under *Phillips*, this meaning relies upon the context of the entire patent including the specification. *See Phillips*, 415 F.3d at 1312–13. The claims themselves provide substantial guidance as to the meaning of particular claim terms. *See Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996).

The PHOSITA is deemed to read the words used in the patent documents with an understanding of their meaning in the field, and to have knowledge of any special meaning and usage in the field. *See Phillips*, 415 F.3d at 1313. Dictionaries, encyclopedias, and treatises may be useful sources in understanding a term’s ordinary meaning because they collect the accepted meanings of terms used in various technical fields. *Id.* at 1318.

The prosecution history also plays a role in claim interpretation. The prosecution history helps to demonstrate how the inventor and the PTO understood the patent and whether the inventor limited the invention in the course of prosecution. *Phillips*, 415 F.3d at 1317. Unambiguous statements in the prosecution history will serve to define language in the claims. *Rembrandt Wireless Techs., LP v. Samsung Elecs. Co., Ltd.*, 853 F.3d 1370, 1376 (Fed. Cir. 2017). In *Rembrandt*, for example, the Federal Circuit upheld a narrow claim construction of the phrase “different types” based on the applicants’ remarks, because they argued when adding the “different types” language that it would “refer to two types of modulation methods, i.e., different families of modulation techniques . . .” *Id.* Interpretation of “different types” therefore required “different families of modulation techniques . . .” *Id.*

Further, statements made to convince the examiner of patentability, when followed by an examiner’s concomitant action based on the statement, will bind the applicant. *Kaken Pharma. Co., Ltd. v. Iancu*, 952 F.3d 1346, 1354 (Fed. Cir. 2020) In *Kaken*, the Federal Circuit reversed the Board’s construction of “treating a subject having onychomycosis,” because applicant had argued during prosecution that “onychomycosis specifically affects the nail” and that the “present invention” showed the claimed medication’s unexpected ability to “penetrate the nail.” *Id.* at 1353–54. The examiner subsequently allowed the claims, noting that the claimed

method required applying medication to the nail such that it “penetrates through the nail plate.” *Id.* at 1354. The meaning of “treating a subject having onychomycosis” was therefore bound to applicant’s arguments, requiring “penetrating the nail plate to treat a fungal infection inside the nail plate or in the nail bed under it.” *Id.*

Extrinsic evidence in the form of expert testimony can aid the inquiry in several ways, including by providing background on the technology at issue, explaining how an invention works, ensuring that the Board’s understanding of the technical aspects of a patent is consistent with that of a PHOSITA, and establishing that a particular term has an established meaning in the pertinent field. *Phillips*, 415 F.3d at 1318. The Board should consider such testimony in the context of the intrinsic evidence but may not use it to arrive at a claim construction that contradicts the construction mandated by the intrinsic evidence. *Vitronics*, 90 F.3d at 1583–84.

1. “Agent”

The term “agent” should be understood as an “OS-level software program or routine,” in view of file history that firmly establishes that the claimed “agent” runs separate from the BIOS.

At minimum, the Board should recognize that “agent” must be interpreted as “a software program or routine” as the district courts have recognized. “Any prior claim construction determination concerning a term of the claim in a civil action . . . that is timely made of record in the inter partes review proceeding will be

considered.” 37 C.F.R. § 42.100(b). In prior litigation involving the ’941 patent, the court held that “the plain and ordinary meaning [of] “agent” is “a software program or routine.” (Ex. 1012 3; *see also* Ex. 1013 36.) This plain meaning ascribed to “agent” by the district court is consistent with definitions that would be known to a PHOSITA. (*See* Ex. 2018 ¶¶126, 133; Ex. 2007 at 2–11, Ex. 2008 at 13; Ex. 2009 at 18–19; Ex. 2010 at 1.) In the context of addressing whether the claimed “agent” was indefinite, one district court observed that “the term is clearly limited to software in this case.” (Ex. 1013 at 34.) “A person of ordinary skill in the art would understand that agent is referring to software because of its complete interaction with the OS.” (*Id.* (citing Ex. 1001 6:18–28, Ex. 2011 ANCC000161).) Further, the “history clearly recites an agent as a “licensed program running at the OS level interacting with a program verification structure stored in BIOS.” (*Id.* (citing Ex. 2011 ANCC000161).) But the issues in that case did not prompt either party to request the narrower construction at issue here, and the court consequently did not consider whether a narrower construction would also be appropriate.

Remarks made both by the applicants and the examiner during prosecution of the ’941 patent require a narrower interpretation of “agent.” *See Rembrandt Wireless Techs.*, 853 F.3d at 1376 (adopting applicant’s definitional remarks relating to “different types” of modulation methods); *Kaken Pharma.*, 952 F.3d at 1354 (adopting applicant’s arguments about onychomycosis).

Here, during prosecution of the '941 patent, the applicant and the examiner both filed remarks that limit the “agent” to an “OS-level software program or routine,” by distinguishing references in which ***BIOS programs*** stored information in the BIOS. As discussed above, applicants distinguished the claims from Ewertz and Misra, by arguing that Ewertz disclosed BIOS routines writing to the BIOS memory area while Misra disclosed an OS program. (*See id.* ANCC000153–54.) Applicants further argued that it was not obvious to modify Misra’s application program to store licenses in BIOS and offered several reasons why the ordinarily skilled artisan would not consider using the BIOS “as a storage medium to preserve ***application data.***” (*Id.* (emphasis added).) In contrast, “[s]oftware license management applications, ***such as the one of the present invention,*** are operating system (OS) level programs.” (*Id.* (emphasis added).) These statements unambiguously define the “agent” added in the amendment. *See Rembrandt Wireless*, 853 F.3d at 1376; *Kaken Pharma.*, 952 F.3d at 1354.

In view of these remarks, the examiner allowed the claims, agreeing with applicants that the cited references did not teach programs running at the OS level interacting with a program verification structure stored in the BIOS. (Ex. 2011 ANCC000161–62.) The examiner’s description of the invention plainly adopts the applicant’s statements and would leave the ordinarily skilled artisan with no

reasonable uncertainty about the scope of the “agent” added through applicant’s amendment. *See Kaken Pharma.*, 952 F.3d at 1354.

The specification of the ’941 patent does not require a different meaning for the claimed “agent.” (Ex. 2018 ¶128.) For example, the specification describes the agent’s prescribed actions of “set[ting] up the verification structure” as “using E2PROM manipulation commands” to store, add and modify items in the “erasable, non-volatile memory of the BIOS.” (Ex. 1001 at 1:65–2:9.)

The Federal Circuit’s decision in *Ancora v. Apple* does not preclude Patent Owner’s proposed construction of “agent.” *See Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732 (Fed. Cir. 2014) (Ex. 2002). There, the court held that the district court erred by narrowly construing “program” to mean “a set of instructions for software *applications* that can be executed by a computer.” *Id.* at 737 (Ex. 2002 at 5). But the Federal Circuit repeatedly emphasized that the “to-be-verified software is different from the verifying software.” *Id.* at 736 (Ex. 2002 at 4). The “program” at issue in that case was the “to-be-verified software.” The court explicitly noted that “[w]e do not have before us a contention that the verification software must be an ‘application’.” *Id.* at 736 n. 1 (Ex. 2002 at 4). Thus, the current case addresses a

different element. The “agent” is clearly part of the verifying software³, and “the applicants were clear that the OS-level language referred to the verifying software.” *Id.* at 737 (Ex. 2002 at 4).

In the institution decision here, the Board considered whether “agent” should be construed as “‘an OS-level software program or routine running on the recited conventional computer to write and/or erase data’ (i.e., performing the task of writing and/or erasing data without the need for additional or specialized hardware).” (I.D. at 9.) The Board declined to adopt Patent Owner’s proposed construction because “it would improperly import extraneous requirements into the claims that are not supported by evidence.” (*Id.* at 10.) The Board, however, has not previously considered the reasoning above, which establishes that “agent” should be interpreted as an “OS-level software program or routine” because of the unambiguous limiting statements in the prosecution history.

III. SUMMARY OF THE ASSERTED REFERENCES

Petitioner asserts that claims 1–2, 11, and 13 would have been obvious over U.S. Patent No. 4,658,093 (hereinafter “Hellman”) in view of U.S. Patent No.

³ Although “using an agent” applies only to setting up the verification structure in claim 1, which is silent about whether the “verifying” or “acting” steps need to occur at the OS level.

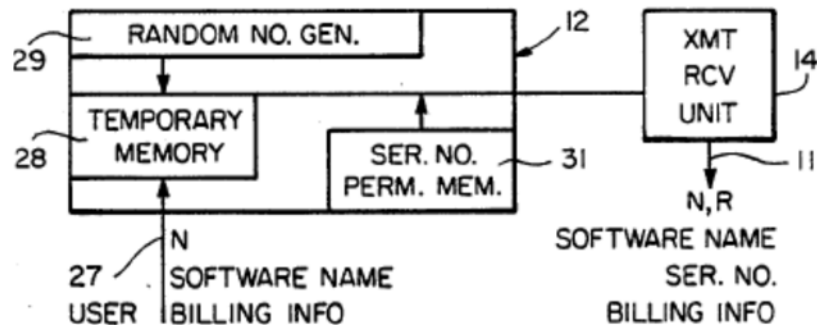
5,892,906 (hereinafter “Chou”) (Ground 1), and that claims 1–3, 6–14, and 16 would have been obvious over Hellman in view of Chou and further in view of U.S. Patent No. 5,933,498 (hereinafter “Schneck”) (Ground 2). (Pet. 6–7.)

A. Hellman

Hellman discloses a “secure software distribution system in which the number of uses of software can be controlled.” (Ex. 1004 1:5–7.) The system is designed to solve a “software piracy” problem and to allow software to be sold on a per use basis. (*Id.* 1:8–23, 4:21–27, 4:34–36.) To accomplish this, Hellman discloses a “base unit” 12 consisting of particular hardware and designed to operate within a pay per use software control system. (*Id.* 5:39–56.)

As shown in Figure 1, Hellman’s base unit 12 is separate from the authorization and billing unit 13. (*Id.* FIG. 1, 5:39–46.) Hellman illustrates features of the base unit in three drawing figures, corresponding to three phases of its operation. (*Id.* 8:52–57.)

Figure 5 illustrates the base unit during generation of a request for software use. (*Id.* 8:61–62.)

**FIG_5**

(*Id.* FIG. 5.) “A user 27 communicates signals representing SOFTWARE NAME, BILLING INFORMATION and N, the number of additional uses desired, to the base unit 12, for example by typing them into a keyboard which is part of base unit.”

(*Id.* 8:62–67.) Hellman discloses that the random number R is generated at the base unit. In the disclosed embodiment, random number generator 29 is “a noisy operational amplifier with hard quantization.” (*Id.* 9:2–7.) The person skilled in the art would recognize such a device as hardware. (Ex. 2018 ¶139.) Importantly, R is stored in temporary memory 28 for later use during verification of the received authorization. (Ex. 1004 9:2–7.) The base unit’s serial number is retrieved from a permanent memory 31, disclosed by Hellman as “a PROM which was burned in during manufacture of the base unit.” (*Id.* 9:8–10.) Such a PROM would be understood not to be writeable, which in the context of Hellman’s disclosure would beneficially keep the serial number “inaccessible to the user.” (*See id.* 9:32–40; Ex. 2018 ¶¶139, 185.) Indeed, Hellman discloses that “[e]ncapsulating the base unit permanent memory 31 *in epoxy or other means* can be used to deter all but a very

small percentage of users from learning their secret keys.” (*Id.* 9:46–49 (emphasis added).)

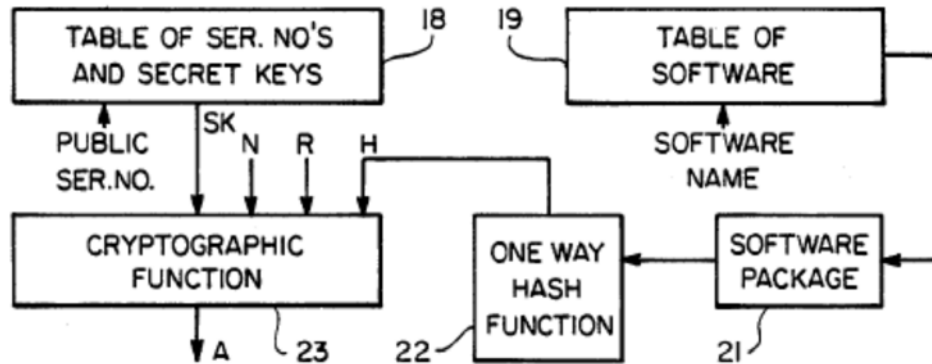


FIG_2

(*Id.* FIG. 5.) At the authorization and billing unit 13, shown in Figure 2, the secret key SK is retrieved from “a table of serial numbers and secret keys” stored in a memory 18, based on the serial number provided by the base unit. (*Id.* 6:17–21.) A “one-way hash generator 22” is used to produce H using a software package 21 retrieved from “another portion of the memory or in an additional memory 19” based on the SOFTWARE NAME provided by the base unit. (*Id.* 6:21–30.) Cryptographic function generator 23 produces “a signal representing authorization A” from “four input signals” that include SK, N, R, and H. (*Id.* 6:62–7:2.) In one embodiment, A is calculated as the output of a one-way hash function using H, R, and N as inputs. (*Id.* 8:13–18.) Alternatively, A is calculated using a modified Data Encryption Standard (DES) using SK as the input to its key port and H, R, and N as inputs to the plaintext port. (*Id.* 8:23–28.)

Figure 6 depicts the base unit during verification of an authorization A to use a software package. (*Id.* 9:16–18.)

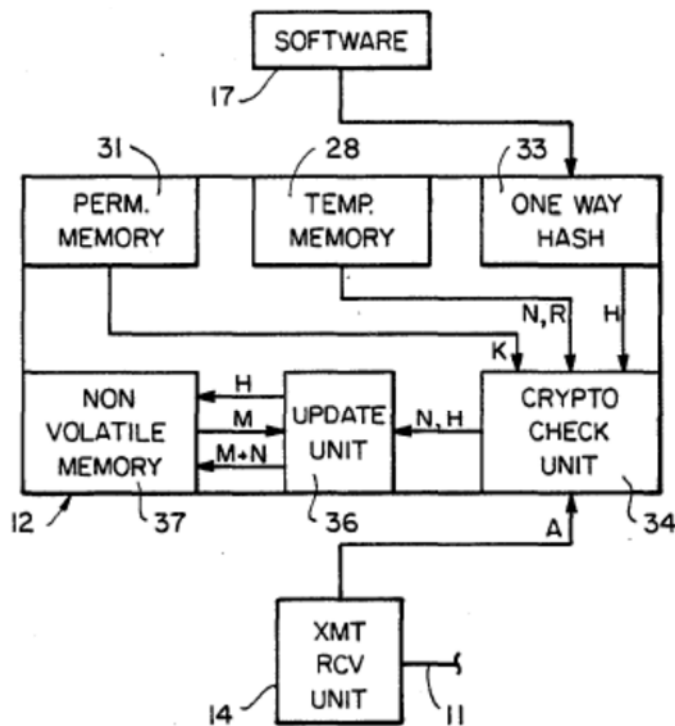


FIG. 6

(*Id.* FIG. 6.) The base unit determines whether authorization A is valid, using a cryptographic check unit 34 described in detail in Figure 7. If A is valid, the cryptographic check unit “applies signals representing N and H to an update unit 36.” (*Id.* 9:64–66.) The update unit 36 then increments the number of uses of software package 17 that are still available, stored in memory 37.

Update unit 36 applies to interrogatory signal representing H to a non-volatile memory 37, for example an EEPROM or a CMOS memory with battery backup. The non-volatile memory 37 applies a signal to the update unit 36, said

signal representing M , the number of authorized uses of the software package H with hash value which still remain unused prior to this new authorization. The update unit 36 adds M and N and applies a signal representing $M + N$ to the non volatile memory 37, so that $M+N$ replaces the old number M in the non-volatile memory 37 as the number of uses of the software package which have been paid for.

(*Id.* 10:1–13.)

Figure 8 illustrates the base unit during use of a software package. (*Id.* 10:33–34.)

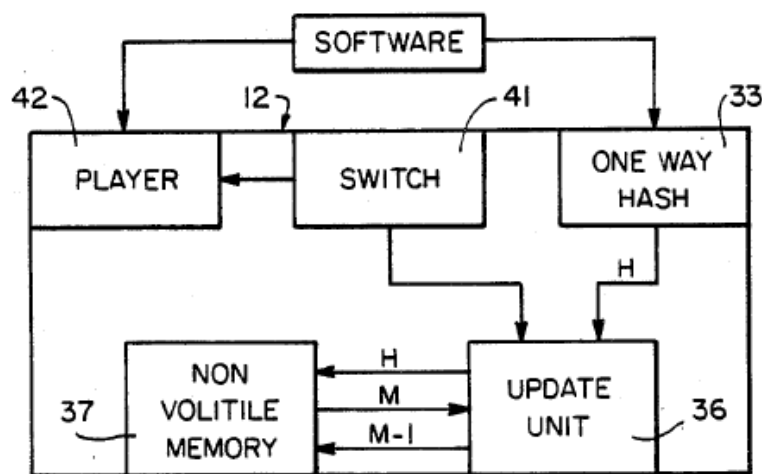


FIG_8

(*Id.* FIG. 8.) In this phase, the base unit controls whether a player 42 can access software based on the value M stored in memory 37. In one embodiment, use of the software package 17 is allowed as long as M is greater than zero. (*Id.* 10:44–54.)

Alternatively, one value of M can represent an infinite number of uses. (*Id.* 10:55–65.) In either case, if M indicates available uses the update unit 36 “sends a control signal to switch 41 which activates software player 42, allowing it to use software package 17.” (*Id.* 10:44–46.) If M does not indicate available uses, the update unit does not send the control signal and the software player 42 therefore would not be activated. (*Id.* 10:50–54.)

The update unit 36 receives signals and generate signals to perform its function.

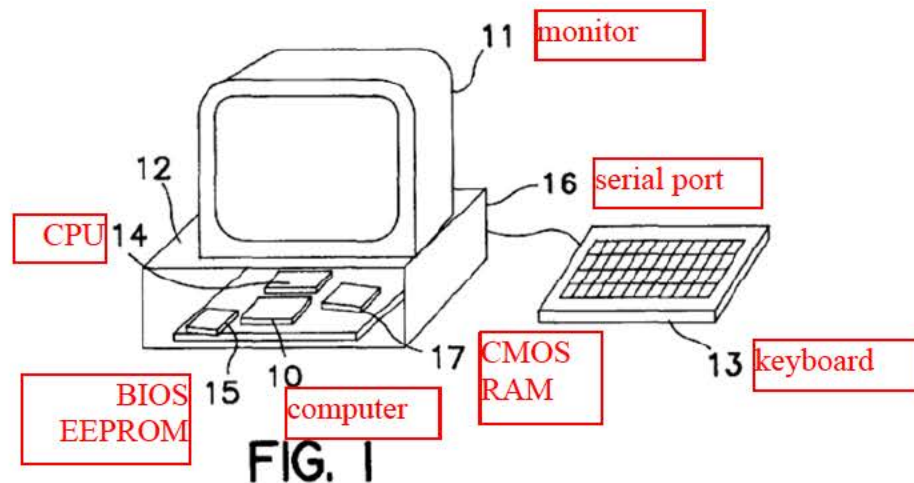
Software package 17 is connected to the base unit 12 and a **signal** representing said software package is operated on by the one-way hash function generator 33 to produce an **output signal** which represents the hash value H. The **signal H** is transmitted to update unit 36 to indicate which software package is being used. Update unit 36 uses H as an **address to non-volatile memory 37**, which responds with a **signal** representing M, the number of uses of software package 17 which are still available.

(Ex. 1004 10:33–43 (emphasis added).) The player 42 could be a variety of hardware devices. “For example, if the software is recorded music then software player 42 would be a record player; if the software is a computer program, then software player 42 would be a **microprocessor or central processing unit (CPU)**.” (*Id.* 10:66–11:3

(emphasis added).) Hellman does not disclose any BIOS, or any operating system (“OS”) or OS-level software. (Ex. 2018 ¶143; Ex. 1003 at ¶105.)

B. Chou

Chou discloses an apparatus and method for discouraging computer theft. (Ex. 1005 Abstract.) This includes use of “a password or other unique information be supplied to the computer *before the computer BIOS routines can be completely executed.*” (*Id.*) The ordinarily skilled artisan would recognize that Chou does not disclose any means of preventing software piracy. (Ex. 2018 ¶144.) Chou’s sole embodiment discloses a computer “which includes a security function stored as a programming routine within the BIOS EEPROM 15,” along with other components illustrated below in an annotated copy of Figure 1.



(Ex. 1005 FIG. 1 (annotations added), 3:30–36; Ex. 2018 ¶144.) Chou discloses that the BIOS memory 15 includes routines for exercising a security function 25. According to Chou, “[u]nless the BIOS routine has completely executed, the

computer operating system can never be accessed rendering the computer inoperative.” (*Id.* 3:60–62; *see also id.* 4:1–5.)

Chou discloses two embodiments: (1) dongle-based locking function (*id.* 5:21–6:19), and (2) password-based locking function (*Id.* 8:42–9:50). In the first (hardware-based) embodiment, the only data stored in BIOS EEPROM is computer serial number and public key—neither of which ever change. In the example that requires a user entered password, the user passwords are stored in a BIOS memory.

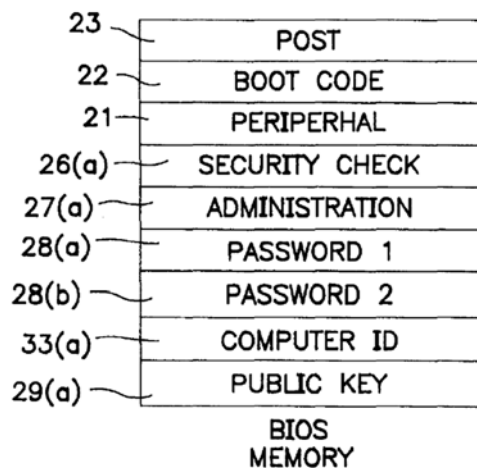
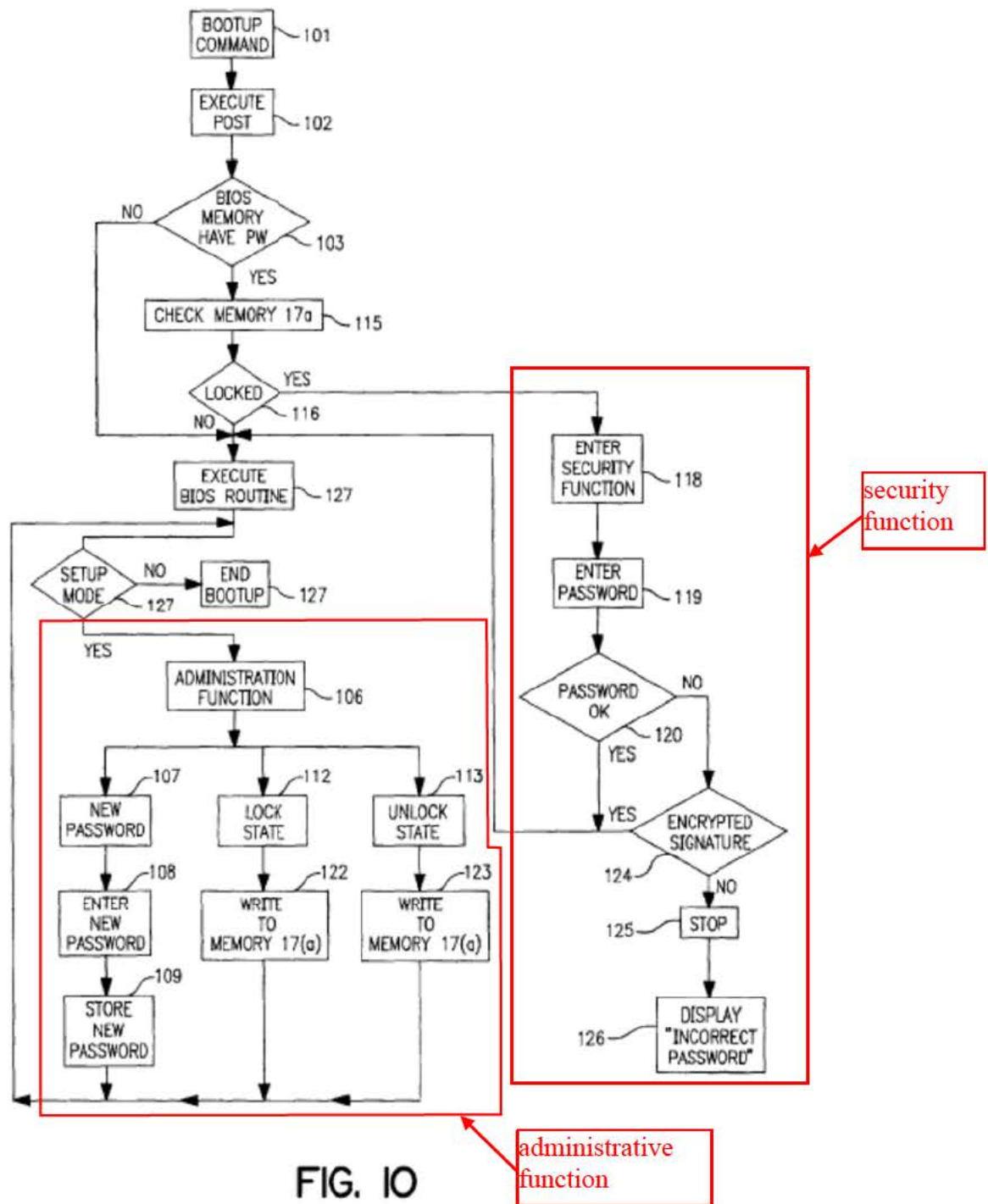


FIG. 7

(*Id.* FIG. 7, 7:14–28.) In both embodiments, the computer’s CMOS RAM includes a location that indicates whether the computer is in a locked state. (*Id.* 7:31–44.)

Chou discloses “execution of the *BIOS routines* including the security function” in Figure 10. (*Id.* 8:42–43 (emphasis added).)



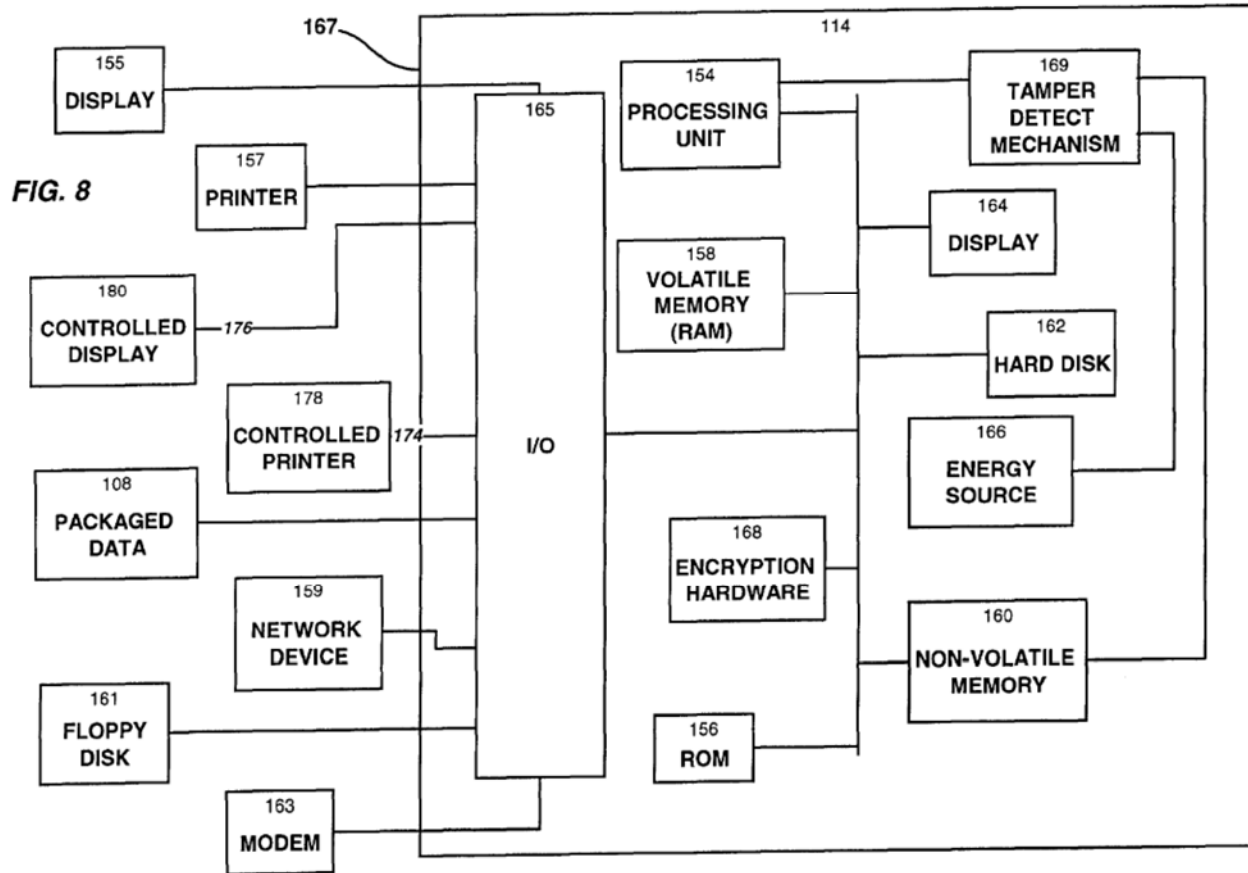
(*Id.* FIG. 10 (annotations added); Ex. 2018 ¶146.) As called out above, Chou's security function 25 includes steps 118–126 requiring the user to enter a password if the computer is in a locked state. (Ex. 1005 9:13–17, 9:26–36.) The system also

includes administrative functions, called out above, that allow the user to enter a new password or to place the computer in the locked or unlocked state. (*Id.* 8:49–61, 9:1–7.) Both the security function and the administrative function are part of BIOS routines that run before the computer can run its operating system. (*Id.* 8:49–55.) The ordinarily skilled artisan would understand that Chou’s administrative function runs outside of the operating system. (Ex. 2018 ¶146.) Further, the only means disclosed by Chou for changing the user-entered password is through the administrative function described above, running outside of the computer’s operating system. (*Id.*; Ex. 1005 FIG. 10.)

C. Shneck

Scheck discloses a system that “relates to control of distribution and access of digital property as well as the payment therefor.” (Ex. 1006 at 1:10–11.) Schneck teaches that known means for protecting digital files were inadequate, including “software-based cryptography” and “the use of external devices or tokens (dongles).” (*Id.* at 3:37–60, 4:65–5:3.)

Schneck discloses a system comprising a distributor 102 (which uses an authoring mechanism 112 to produce packaged data 108) and a user 104 (which uses an access mechanism 114 to access data in controlled ways). (*Id.* at FIGS. 1 & 5, 9:46–59.) Schneck illustrates the access mechanism 114 in Figure 8.



(*Id.* at FIG. 8.) Schneck explains that:

All components of the access mechanism 114 are *packaged in such a way as to exclude any unknown access by a user* and to discover any such attempt at user access to the components or their contents. That is, the access mechanism 114 is packaged in a tamper-detectable manner, and, once tampering is detected, the access mechanism is disabled. The line 167 depicted in FIG. 8 defines a so-called security boundary for the components of the access mechanism 114. Any components required for tamper detection (tamper detect mechanism 169) are also included as part of the access mechanism 114.

(*Id.* at 15:51–63 (emphasis added).) These physical self-protection measures are used as a means for rendering the access mechanism inoperative, among other passive and active mechanisms employed to destroy data. (*Id.* at 15:64–16:19, 16:49–59.)

IV. PETITIONER FAILS TO ESTABLISH OBVIOUSNESS BASED ON EITHER ASSERTED GROUND

The asserted combinations fail to disclose the claimed agent, which requires an OS-level software program or routine. With respect to claim 1, the boiler-plate motivations offered by Petitioners for modifying Hellman with the teachings of Chou fail when scrutinized.

With respect to the asserted combination, both the Petition and the Institution Decision relied on Chou’s teaching that “[r]ecent changes in computer BIOS memory storage devices permit writing data to the BIOS memory, offering the opportunity to provide password protection within the same memory which store the BIOS routines.” (Ex. 1005 1:63–66; *see* I.D. 16, 27; Pet. 34.) But other references cited against the ’941 patent during prosecution and reexamination have had similar disclosures. For example, Ginter teaches the concept of “authentication values hidden in *normally inaccessible locations*” including: “[n]on-volatile, writable, storage in the appliance or its components, such as that used for . . . *standard BIOS software*, etc.” (Ex. 2019 245:55–246:24.) The examiner was aware of this teaching during prosecution of the ’941 patent. (Ex. 2011 ANCC000111.) Ewertz similarly

taught that the flash memory chip storing a BIOS could be used to store “set-up code or data, and other information or logic.” (Ex. 2015 5:20–25.) Likewise, the examiner in the ’560 reexamination recognized that Lewis’s disclosure of writing a MAC to the non-volatile memory was part of the BIOS, because “[i]t is common in the art to implement such drivers in the BIOS area.” (*See* Ex. 2001 164.) But none of the references cited against the ’941 patent, now including Hellman, Chou, and Schneck, disclose using an OS-level software program or routine set up a verification structure in a memory of the BIOS, as claimed.

Petitioners separately fail to establish obviousness of dependent claims 3, 8, 9, and 14 because the alleged combination of Hellman, Chou, and Schneck would not function for its intended purpose.

Additionally, objective considerations emphasize that the claims were not obvious; industry praise and licensing success demonstrate the strength of the invention.

A. Governing Legal Principles

Petitioner must prove that the claimed subject matter would have been obvious to a PHOSITA at the time of the invention. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). A complete analysis requires determining (1) the scope and content of the prior art; (2) differences between the prior art and the claims; (3) the level of ordinary skill in the art; and (4) relevant objective indicia of

nonobviousness. *Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17–18 (1966).

An invention “composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art.” *KSR*, 550 U.S. at 418. Petitioner must show that a PHOSITA would have been motivated to pick out the asserted multiple references and combine them in the way the claimed invention does. *Personal Web Techs., LLC v. Apple, Inc.*, 848 F.3d 987, 993–94 (Fed. Cir. 2017). This showing must lack any “hint of hindsight,” *Star Scientific, Inc. v. R.J. Reynolds Tobacco Co.*, 655 F.3d 1364, 1375 (Fed. Cir. 2011), so as to avoid “reconstruction by using the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.” *In re NTP, Inc.*, 654 F.3d 1279, 1299 (Fed. Cir. 2011) (internal citation omitted). A challenger cannot pick and choose among individual parts of assorted prior art references as a mosaic to recreate a facsimile of the claimed invention. *Akzo N.V. v. United States ITC*, 808 F.2d 1471, 1481 (Fed. Cir. 1986).

The test for obviousness is what the combined teachings of the references would have suggested to a PHOSITA, *In re Young*, 927 F.2d 588, 591 (Fed. Cir. 1991), and each reference must be evaluated in its entirety. *In re Evanega*, 829 F.2d 1110, 1112 (Fed. Cir. 1987). A reference must be considered for all that it teaches,

including disclosures that teach away from the invention at hand. *In re Dow Chem. Co.*, 837 F.2d 469, 473 (Fed. Cir. 1988). A reference teaches away when a PHOSITA would be discouraged from following the path set out in the reference or would be led in a direction divergent from the path taken by the inventor. *Monarch Knitting Mach. Corp. v. Sulzer Morat GmbH*, 139 F.3d 877, 885 (Fed. Cir. 1998).

If a proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 902 (Fed. Cir. 1984).

Petitioner is required to show that its disparate references would have suggested to a PHOSITA not only that they could make but that they should make the claimed invention and that they would have a reasonable expectation of success in doing so. *In re NuVasive, Inc.*, 842 F.3d 1376, 1381–82 (Fed. Cir. 2016). This requires articulated reasoning based on rational, factual underpinnings; conclusory statements of obviousness do not suffice. *KSR*, 550 U.S. at 418; *In re NuVasive*, 842 F.3d at 1382–83. Any expert testimony must explain in detail how specific references could be combined, which combinations of elements in specific references would yield a predictable result, and how any specific combination would operate or read on the asserted claims. *Activevideo Networks, Inc. v. Verizon Commc'ns, Inc.*, 694 F.3d 1312, 1327 (Fed. Cir. 2012). “Expert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to

little or no weight.” 37 C.F.R. § 42.65(a); *see, e.g., Comcast Cable Commc’ns, LLC v. Rovi Guides, Inc.*, IPR2019-00299, Paper 45 at 103 (PTAB June 29, 2020).

B. Claim 1: The Petition Fails to Establish a Valid Motivation for Combining Hellman With Chou

1. The Petition’s Asserted Motivations to Combine Hellman with Chou Are Logically Flawed

Petitioners assert four motivations that would allegedly result in the use of Chou’s BIOS memory in place of Hellman’s non-volatile memory 37, but each of these motivations is founded on conclusory testimony, incomplete analysis, and/or fails to consider reasons against the proposed combination.

Petitioners first contend that “Hellman’s disclosure would be applied on a computer with BIOS and a memory storing the BIOS.” (Pet. 31–32.) But the mere presence of BIOS among computers does not establish that Chou’s BIOS memory would have been used to store Hellman’s value “M,” in place of Hellman’s separately disclosed non-volatile memory 37. Recognizing this fact, Petitioners contend that storing information in the BIOS would prevent tampering, citing Chou. (Pet. 32.) But Chou’s disclosure does not discuss tamper-prevention. (Ex. 2018 ¶191; Ex. 1005 1:54–2:7.) At most, Chou explains that the password protection cannot be defeated by removing CMOS RAM power, placing the device in a locked state. (Ex. 2018 ¶192; Ex. 1005 1:66–2:1.) None of the other evidence cited by Petitioners establishes that an ordinarily skilled artisan would be motivated to store

information in the BIOS. For example, the mere fact that it was *generally* known to store more than one thing in a single memory would not necessarily motivate using the BIOS to store Hellman’s value “M.” (*See* Pet. 32–33.)

Petitioners secondly contend that the combination would have been motivated because “it was not common to have a large number of [EEPROM] modules in any computer.” (Pet. 33.) The Wolfe declaration does not support this statement, and Dr. Wolfe testified at his deposition that it was common to have multiple EEPROM modules in computers at the time of the invention when non-standard hardware, e.g., graphics cards, was added to a computer. (Ex. 2026 39:16–40:15.) Further, the Petition offers no basis for its implied assumption that adding a single EEPROM (i.e., Hellman’s non-volatile memory 37) would result in a “large number of such modules” in a given computer. (*See id.* 40:16–41:4.) Further, as taught by Ewertz, it was known that BIOS routines and configuration data occupied more memory that was available in standard BIOS memory space. (Ex. 2015 2:6–12.)

Petitioners thirdly contend that storing information in the BIOS would prevent tampering, again citing Chou. The Board gave this argument particular weight when instituting trial. (I.D. 16.) As discussed above, Chou’s disclosure should be interpreted in the context of Chou’s improvement—which discloses disabling “the boot up process” when password information is not provided. (Ex. 2018 ¶191–92; Ex. 1005 1:66–2:1.)

Fourth and finally, Petitioners argue that economic and operational efficiencies would have motivated sharing Chou's EEPROM BIOS in place of Hellman's non-volatile memory 37. (Pet. 34–35.) As discussed above however, Dr. Wolfe testified about “lots of ordinary desktop computers that did have multiple [EEPROM] modules.” (Ex. 2026 39:16–41:15.) Also, according to Dr. Wolfe the switch 41—at least—would have been added to implement Hellman's system. (*See* Ex. 2026 35:9–18 (“There would have to be some hardware involved somewhere”).) Further, the value of the security functions could reasonably outweigh the cost of an EEPROM chip, eliminating the need for cost savings of the type Petitioners rely upon. As taught by Schneck, a separate hardware-protected access mechanism was desirable as a means for protecting information. (Ex. 1006 at 15:51–63.)

Viewed in context of the facts discussed above, Petitioners' alleged motivations for combining Hellman with Chou cannot support obviousness over either Ground 1 or Ground 2. The institution decision credits Dr. Wolfe's analysis, which concludes that storing Hellman's license information in the BIOS EEPROM would discourage users from tampering with the license information and provide extra protection to sensitive information. (I.D. 26–27 (citing Ex. 1003 ¶¶112–16.)) But this is little more than hindsight, and relies on a mischaracterization of Chou, as discussed above. (Ex. 2018 ¶¶191–92.)

2. Petitioners Do Not Offer Any Motivation for Storing Hellman's Value "M" in the Memory Area "Used By the BIOS"

Petitioners' asserted combination does not teach storing the value "M" in the memory area that is "used by the BIOS," as the "memory of the BIOS" term requires. Petitioners assert that the combination of Hellman and Chou would teach an erasable, non-volatile memory of the BIOS, in view of Hellman's non-volatile memory 37 and Chou's BIOS EEPROM. (Pet. 37–38 (citing Ex. 1003 ¶¶104–120F.)) In short, Petitioners contend that Hellman's non-volatile memory 37 would have been modified to include Chou's BIOS. But Petitioners' argument and proposed combination improperly assumes that the entirety of Chou's EEPROM chip constitutes "memory of the BIOS," such that storing data in EEPROM is the same as storing it in BIOS memory.

Petitioner's argument has been considered and rejected twice—once by the Board in Samsung's IPR2020-01184 (Ex. 2016 at 26) and once by the Examiner in Microsoft's '560 reexamination (Ex. 2001 at 163–64). The Hellman/Chou combination here fails for the same reasons as the Schwartz-based grounds in Samsung's IPR Petition, which the Board characterized as "weak" because they did not explain why the "entirety of EEPROM 250a, including the memory space that stores configuration module 307 and zip/zone module 305, is a 'non-volatile memory of BIOS.'" (Ex. 2016 at 26). In the '560 reexamination, the examiner

similarly distinguished programs or tables “in BIOS” from programs or table that merely reside in non-volatile memory that also contains BIOS. (Ex. 2001 at 163–64.) Petitioners here similarly do not explain how Hellman’s non-volatile memory 37 would accommodate Chou’s BIOS such that the claimed verification structure could be said to be “used by” the BIOS. (*See* Ex. 2018 ¶¶194–95.)

C. Claim 1: Petitioners’ Combinations Fail to Identify the Claimed Step of Using an Agent to Set Up a Verification Structure in the Erasable, Non-Volatile Memory of the BIOS

Petitioners rely on Hellman for nearly all elements of the claim, (*see* Pet. 35–42), but assert that Chou it would have made it obvious to implement a BIOS in Hellman’s system on a computer in which the same erasable, non-volatile memory stored both Hellman’s number of authorized values and a BIOS—thus disclosing the claimed “non-volatile memory of the BIOS.” (Pet. 38.)

Petitioners mischaracterize the ’941 patent repeatedly, contending that the invention consists merely of storing a license record for a program in the BIOS memory. (*See, e.g.*, Pet. at 1, 9, 19.) But the mere use of a common erasable, non-volatile memory to *store* a BIOS and a license record is not adequate to invalidate the challenged claims. Claim 1 requires “using an *agent* to set up a verification structure in the erasable, non-volatile memory of the BIOS, the verification structure accommodating data that includes at least one license record.” As discussed above in section II.F, the ’941 patent’s prosecution history establishes that the claimed

“agent” would be understood as an “OS-level software program or routine.” Neither Petitioners nor Dr. Wolfe considered the prosecution history when evaluating whether the art discloses the claimed “agent.” (Pet. 23; Ex. 2026 23:20–24:1.) This is a fatal error. Even Petitioners acknowledge that the “agent” was critical to the Examiner when allowing the claims. (*See* Pet. 19.)

1. The Petition Wholly Fails to Identify Any Teaching in the Combined References of the claimed “Agent,” Which Requires an “OS-Level Software Program or Routine”

One cannot conclude that the PHOSITA looking to Hellman and Chou would arrive at a system that results in using an agent, *i.e.* OS-level software program or routine, to set up a verification structure in a memory of the BIOS.⁴ (Ex. 2018 ¶194.)

⁴ The Institution Decision asserts that Patent Owner admitted that Hellman discloses an “agent.” (I.D. 23 (citing IPR2021-00663, Paper 13, at 33). Patent Owner respectfully disagrees. The passage cited by the Board plainly distinguishes Hellman’s update unit 36, using “agent” as shorthand for Petitioner’s contentions. *See also* IPR2021-00663, Paper 13, at 28. The heading of the section at issue explicitly states that Hellman “does not disclose an OS-level software ‘agent’ . . . as Claim 1 requires.” *Id.* The fact that “agent” is not preceded by “alleged” does not change the context or meaning of the arguments and is not an unambiguous admission.

Hellman and Chou have long been known to accused infringers of the '941 patent, along with this plain weakness in the teachings of the references. (*See* Ex. 2022 at 2, 3, 31 (Aug. 25, 2015 invalidity contentions asserting Hellman and Chou).)

Hellman does not teach or suggest a BIOS. (Pet. 37; Ex. 1003 ¶¶105; Ex. 2018 ¶¶143, 182.) By itself, Hellman therefore does not disclose a “memory of the BIOS.” Petitioners contend Chou’s teaching of a BIOS capable of storing authentication information would motivate modifying Hellman to use its non-volatile memory 37 to store a BIOS in addition to the values of M. (Pet. 37–38.) Neither the Petition nor the cited portions of Dr. Wolfe’s declaration analyze whether the proposed combination of Hellman and Chou (or Hellman, Chou, and Schneck) would disclose an OS-level software program or routine that performs the claimed setting up a verification structure in the BIOS. (*See* Pet. 40–41 (citing Ex. 1003 ¶¶133–38.)

Indeed, none of the cited references teach or suggest such an OS-level software program or routine capable of setting up the claimed verification structure in a BIOS memory (exactly what the examiner considered to be the invention). Hellman does not disclose any BIOS, or any operating system (“OS”) or OS-level software. (Ex. 2018 ¶¶143, 182.) This is true even when considering Hellman’s authorization and billing agent 13, which does not write Hellman’s value M to the non-volatile memory 37. (*See* Ex. 1004 10:1–13 (update unit 36 writes M to memory

37).) Hellman consequently does not disclose an OS-level agent capable of manipulating non-volatile memory of the BIOS.

Chou at most discloses a BIOS routine capable of manipulating passwords stored in BIOS EEPROM but fails to disclose any means for storing information in the BIOS during “normal” operation of the computer. (Ex. 1005 FIG. 10, 8:42–43, 8:49–55; Ex. 2018 ¶¶146, 179.)

Because Petitioners’ grounds fail to disclose or suggest OS-level software or program capable of setting up the claimed verification structure in a BIOS memory (*see* Ex. 2018 ¶¶177–180), the claims must be upheld.

2. The Petition Fails to Identify a Legally Sufficient Motivation For Modifying Hellman’s Update Unit 36 or Authorization and Billing Unit 13 to Be Software-Based

At minimum, the “agent” requires a “software program or routine,” as discussed above in section II.F. Petitioners fail to establish obviousness based on Hellman and Chou, even if the Board declines to interpret the agent as requiring OS-level software. The Institution Decision credited Dr. Wolfe’s testimony that a PHOSITA would have been motivated to implement update unit 36 and authorization and billing unit 13 in software. (ID 22 (citing Ex. 1003 ¶¶137B–138B).) Dr. Wolfe’s analysis, however, is legally deficient because it fails to establish what a PHOSITA *would* do—instead relying on a series of statements about what that person might, or could do. *See In re NuVasive*, 842 F.3d 1381–82.

Dr. Wolfe concedes that Hellman does not “specifically disclose how update unit 36 is implemented” (Ex. 1003 ¶137) and “does not explicitly say whether the authorization and billing unit 13 should be implemented in software, hardware, or a combination of the two” (*id.* ¶138A). Dr. Wolfe’s analysis ignores substantial evidence that Hellman’s system is expressly predicated on the use of hardware, as discussed at length by Dr. Martin. (Ex. 2018 ¶¶164–71.) Hellman’s express disclosure of switch 41 (*id.* ¶164), the wire sending the control signal from update unit 36 to software player 42 (*id.*), the use of epoxy to encapsulate switch 41 and the wire (*id.*), special factory seals or a dab of paint (*id.*), and numerous “signals” representing values K, N, R, H, C, A, etc. (*id.* ¶166) confirms that Hellman’s system requires particular hardware. Indeed, every mention of “software” in Hellman refers to the software package being authorized for use. (*Id.* ¶170.) Further, Hellman discloses that switch 41 is needed to activate software player 42. (*Id.* 10:44–46, 10:50–54.) Petitioner never identifies where a software version of Hellman’s update unit 36 would run, i.e. what processor would run it. (*See* Pet. 41; Ex. 1003 ¶¶137–137B.) To the extent Petitioners rely on a processor version of Hellman’s software player 42, this would be inoperative because it could not run before the software player 42 is “activated.” (*See* Ex. 1004 10:44–54.)

Further, Dr. Wolfe’s declaration is littered with equivocal statements that fail to establish whether Hellman’s hardware-based system *would* have been modified

for use as software. For example, Dr. Wolfe contends that implementing Hellman’s update unit 36 or authorization and billing unit 13 would be “*up to the discretion* of the implementer.” (Ex. 1003 ¶¶137A & 138A (emphasis added).) Dr. Wolfe similarly contends that certain activities or tasks are “*of a type* that could be performed in software, hardware, or both.” (*Id.* ¶¶137B & 138B (emphasis added).) None of these statements evidence anything more than what the PHOSITA *could* do, which is inadequate. *See Personal Web Techs.*, 848 F.3d at 993–94 (obviousness requires showing that a skilled artisan would have been motivated to make the combinations or modifications of prior art to arrive at the claimed invention—not merely that they could have done so).

The only motivation for modifying Hellman’s hardware-based system, according to Dr. Wolfe, is to allow “changes in implementation logic” without having to physically modify the systems. (*Id.* ¶¶137B & 138B.) Dr. Wolfe recognizes, however, implementing the update unit in software would come with a set of security risks, for example, requiring encryption of the value M. (Ex. 1003 ¶¶142–43; *see* Ex. 2018 ¶172.) Dr. Wolfe fails to establish why a the ordinarily skilled artisan would proceed despite these downsides. *See In re NTP*, 654 F.3d at 1299 (obviousness analysis must “avoid hindsight reconstruction by using the patent in suit as a guide through the maze of prior art references”). This is a fatal error.

NTP reversed the Board, holding claims were non-obvious because the Board’s “piecemeal analysis” improperly required adding features (there, an additional network interface) to devices that already disclosed the feature. *Id.* Here, Hellman disclosed hardware solutions specifically capable of restricting access to the non-volatile memory and the other components of the base unit. (Ex. 2018 ¶¶169–73.) Further, Dr. Martin explains that implementing Hellman’s devices in software would require the software solution to isolate and protect itself from the very software packages whose authorization is at issue, making obviousness based on the proposed combination “highly implausible.” (*Id.* ¶¶174–75.) Petitioner’s hardware-to-software modifications were unnecessary in view of solutions already identified by Hellman.

In short, Petitioners do not consider both the reasons for *and against* the proposed modifications—and therefore fails to establish the required motivation to modify Hellman’s system. *See In re Dow Chem. Co.*, 837 F.2d at 473 (references must be considered for all that they teach).

3. The Petition Improperly Conflates the Memory of the BIOS and the Verification Structure, Which the Claim Requires to be Separate

Regardless of the Board’s interpretation of “agent,” the Petition improperly cites the same elements of Hellman for the distinct “memory of the BIOS” and “verification structure” limitations of claim 1.

Petitioners contend that the act of storing values “M” in the non-volatile memory 37 is adequate to disclose the claimed step of using the agent to set up a verification structure. (*See* Pet. 41 (citing Ex. 1003 ¶¶133–38.) Dr. Wolfe asserts that Hellman discloses a “verification structure” in the form of the *memory structure* of non-volatile memory 37 storing at least one value M at memory addresses defined by at least one hash value H. (Ex. 1003 ¶135.)

The “verification structure” set up according to the Petition is little more than the ordinary intended use of a computer memory. To read or write data stored in a computer memory, an address is needed. The mere use of such an address does not define a “structure” separate from the memory itself. Neither Petitioners nor Dr. Wolfe indicates that the hash values H are stored or retained in any way. (*See* Pet. 41.) The “memory structure” identified by Dr. Wolfe would be nothing more than Hellman’s memory 37, with values of “M” at locations that would not be ascertainable without recalculating a hash value H from the software 17. (*See* Ex. 1003 ¶135.) The alleged “verification structure” would therefore be nothing more than a nebulous collection of hash values that do not exist—and certainly do not exist in an erasable, non-volatile memory of the BIOS as required by the claim.

D. Dependent Claims 3, 8, 9, 14: The Allegedly Combined System Proposed in the Petition Would Not Function for its Intended Purpose

Claims 3, 8, 9, 14, and 15 of the '941 patent⁵ each require an encrypted license-record. (Ex. 1001 7:8–20 (claim 3), 7:44–50 (claim 8), 7:51–62 (claim 9), 8:7–10 (claim 14), 8:11–21 (claim 15).)

In Ground 1, based on Hellman and Chou, Dr. Wolfe asserts that the license record is the “number of authorized uses value M.” (Ex. 1003 ¶134.) Ground 2 contends that the license record would have been supplied in a modified version of Hellman’s system, in which the authorization A is stored in the non-volatile memory 37. (Ex. 1003 ¶¶139–150.) Dr. Wolfe contends that it would have been obvious to store Hellman’s authorization A in non-volatile memory 37 at memory address H. (Ex. 1003 ¶147.) This modification would render Hellman’s system inoperative. (Ex. 2018 ¶¶197–205.)

First, Hellman discloses two embodiments of the authorization A. (Ex. 1004 8:13–22, 8:23–30.) In the first embodiment, authorization A is generated using a “one-way hash function” that Dr. Wolfe himself recognizes is practically impossible to reverse. (Ex. 2026 28:6–29:20; Ex. 1004 3:20–56.) Storing a hash value of this

⁵ Several additional claims, not at issue here, also require an encrypted license-record or similar terms.

sort would render any information used to create the value irretrievable and entirely defeat the purpose of storing a number of authorized uses in the non-volatile memory.

Second, in the alternate embodiment where authorization A is an encrypted value, only a system where SK and K are the same value would allow the base unit to update the value of authorization A to increment or decrement the number of remaining uses. (Ex. 2018 ¶¶202–05.) In the alternative public-key variation, only the authorization and billing unit can generate new values of A, making it impossible for the update unit to change the number of remaining uses. (Ex. 2018 ¶¶199–201.)

Finally, even assuming Dr. Wolfe intended to rely upon the alternate embodiment where authorization A is an encrypted value, Hellman discloses that the values of H, R, N, and SK are used to generate authorization A. (Ex. 1004 6:62–7:2; Ex. 2018 ¶203.) Absent is the value M, which is the number of uses still available. (Ex. 1004 10:42–43; Ex. 2018 ¶203.) In short, storing authorization A would prevent Hellman’s system from operating to verify the number of remaining uses before allowing the software player 42 to operate. (Ex. 2018 ¶203.) Neither Petitioners nor Dr. Wolfe suggests any reason for modifying authorization A to include this information, nor do they consider the effects of doing so.

E. Dependent Claims 2, 6–7, 10–13 Are Not Obvious For All of the Same Reasons as Claim 1

Claims 2, 6–7, 10–13 depend from claim 1 and are therefore not obvious for all of the reasons discussed above.

F. Objective Evidence Establishes Non-Obviousness of the Claimed Invention

Objective indicia of non-obviousness are part of the overall obviousness analysis, not merely an inquiry after a finding of obviousness is made. *Apple Inc. v. Samsung Elec. Co., Ltd.*, 839 F.3d 1034, 1048 (Fed. Cir. 2016) (en banc). Indeed, objective indicia “can be the most probative evidence of non-obviousness in the record, and enables the . . . court to avert the trap of hindsight.” *Custom Accessories, Inc. v. Jeffrey-Allan Indus., Inc.*, 807 F.2d 955, 960 (Fed. Cir. 1986).

1. A Leading BIOS Company Praised the Claimed Invention and Agreed to Commercialize

Professional approval and praise by others can indicate non-obviousness of an invention. *Power Integrations, Inc. v. Fairchild Semiconductor Int’l, Inc.*, 711 F.3d 1348, 1368 (Fed. Cir. 2013); *Apple*, 839 F.3d at 1053.

Here, a significant player in the BIOS market directly acknowledged the value in the ’941 patented technology. In 2005, American Megatrends Inc. (AMI) issued a joint press release directly praising the technology of the ’941 patent. (Ex. 2027.) At the time, AMI was “best known for its BIOS, but also supplie[d] state-of-the-art

hardware, software and utilities to over 55 percent of today's high-tech manufacturers.” (*Id.* at 2; *see also* Ex. 2026 at 15:7–11, 17:17–19:18.)

From this position of strength in the industry, AMI chose to enter a strategic alliance with Ancora “to offer BIOS based security products utilizing Ancora’s Platform Security Anchor™ (PSA) technology (as described in US Patent 6,411,941.” (Ex. 2027 at 1.) AMI’s president stated “We see great value for consumers in BIOS based security products.” (*Id.*) [REDACTED]

[REDACTED]

[REDACTED]

The Board must find nexus “where the evidence of secondary considerations is the ‘direct result of the unique characteristics of the claimed invention.’” *Fox Factory Inc. v. SRAM LLC*, 944 F.3d 1366, 1374–75 (Fed. Cir. 2019), citing *In re Huang*, 100 F.3d 135, 140 (Fed. Cir. 1996). Nexus is found when the claims specify features that the objective evidence demonstrates are significant in some way. *Polaris Indus., Inc. v. Arctic Cat, Inc.*, 882 F.3d 1056, 1073 (Fed. Cir. 2018) (the transmission of a claimed all terrain vehicle was a significant feature, nexus was found); *Apple*, 839 F.3d at 1053–57 (nexus found despite mobile phones being complex devices having have hundreds of unclaimed features because the claims specified one feature that the objective evidence demonstrated was significant); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1333–34 (Fed. Cir. 2019)

(nexus found where the claims were drawn to a fry pan with an oil sensor and the industry had praised that feature).

AMI's praise, entry into the strategic alliance, and [REDACTED] directly evidence the non-obviousness of the '941 patent. First, the announcement specifically references the '941 patent. Second, the strategic alliance was specific to Ancora's PSA technology, embodying the '941 patent. (Ex. 2030 ¶4.) Accordingly, the required nexus plainly exists, linking the praise and partnership to the '941 patent. Looked at another way, the '941 patent claims using BIOS to store at least one license record used to verify a program. Like the cases cited above, AMI's statement in the press release specifically credits this claimed feature of the '941 patent's claim 1: BIOS-based security.

Additionally, [REDACTED]

[REDACTED] *See Bosch Automotive Service Solutions, LLC v. Matal*, 878 F.3d 1027, 1037-38 (Fed. Cir. 2017) (requiring proof that licenses were entered based on the strength of the patents and not solely to avoid the costs of litigation). [REDACTED]

[REDACTED] AMI's press release corroborates that [REDACTED] based on the patented technology's strength [REDACTED].

2. Additional Licenses Establish the Significance of the Claimed Invention

Several very significant companies have taken a license to the ‘941 patent. Licensing activity evidences the non-obviousness of a patent where the licensing holds a nexus to the patent at issue and was entered into based on the strength of the patent. *Impax Laboratories Inc. v. Lannett Holdings Inc.*, 893 F.3d 1372, 1381-82 (Fed. Cir. 2018); *Transocean Offshore Deepwater Drilling, Inc. v. Maersk Drilling USA, Inc.*, 699 F.3d 1340, 1353 (Fed. Cir. 2012).

[REDACTED] (among others) have licensed the ‘941 patent. [REDACTED]
[REDACTED] Each license was for [REDACTED]
[REDACTED]
[REDACTED] and the licensing cannot be questioned. [REDACTED]
[REDACTED]

Although Ancora sued each of these companies, [REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED] Licenses in which the licensee

paid above the cost of litigation demonstrate a nexus between the license and patent's strength. *Transocean*, 699 F.3d at 1353.

Moreover, [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

Of particular note, Hellman and Chou were known references before each of these settlements. Apple publicly filed invalidity contentions on August 25, 2015 that identified asserted references including Hellman and Chou for all litigants to see. (Ex. 2022 2, 3, 31.) [REDACTED]

[REDACTED]

Importantly, at least [REDACTED] asserted the Hellman/Chou invalidity combination [REDACTED]

[REDACTED]

[REDACTED] If these sophisticated companies truly believed in the strength of the asserted invalidity grounds, [REDACTED]

[REDACTED].

In short, [REDACTED] and BIOS leader AMI (among others) each have confirmed the value of the '941 patent.

V. CONCLUSION

For the foregoing reasons, the Board should find the challenged claims of the '941 patent not invalid.

Respectfully submitted,
FITCH EVEN TABIN & FLANNERY LLP

Dated: May 3, 2022

By: /David A. Gosse/
David A. Gosse
Registration No. 61,511
dgosse@fitcheven.com

120 South LaSalle Street
Suite 2100
Chicago, Illinois 60603
(312) 577-7000
(312) 577-7007 (fax)

CERTIFICATE OF WORD COUNT

The undersigned hereby confirms that the foregoing Preliminary Response complies with the type-volume limitation of 37 C.F.R. § 42.24(a). Excluding the items listed in 37 C.F.R. § 42.24(a) as not included in the word count, this preliminary response contains less than 14,000 words, calculated by the word-processing system used to prepare it.

Dated: May 3, 2022

By: /David A. Gosse/
David A. Gosse
Registration No. 61,511
dgosse@fitcheven.com

120 South LaSalle Street
Suite 2100
Chicago, Illinois 60603
(312) 577-7000
(312) 577-7007 (fax)

CERTIFICATE OF SERVICE

The undersigned hereby certifies that on May 3, 2022, a complete and entire copy of the PATENT OWNER RESPONSE, was served via electronic mail to the attorneys listed below at PerkinsService-Nintendo-Ancora-IPR@perkinscoie.com:

Lead Counsel	Back-up Counsel
Jerry A. Riedinger Reg. No. 30,582 riedinger-ptab@perkinscoie.com Perkins Coie LLP	Jose Villarreal, Reg. No. 43,969 villarreal-ptab@perkinscoie.com Perkins Coie LLP Kyle Canavera, Reg. No. 72,167 canavera-ptab@perkinscoie.com Perkins Coie LLP Theresa H. Nguyen (<i>pro hac vice</i>) nguyen-ptab@perkinscoie.com Perkins Coie LLP Tara Kurtis, Reg. No. 74,846 kurtis-ptab@perkinscoie.com Perkins Coie LLP

Dated: May 3, 2022

By: /David A. Gosse/
David A. Gosse
Reg. No. 61,511
dgosse@fitcheven.com
Nicholas T. Peters
Reg. No. 53,456
ntpete@fitcheven.com
Karen J. Wang
Reg. No. 62,503
kwang@fitcheven.com
FITCH, EVEN, TABIN & FLANNERY LLP
120 South LaSalle Street, Suite 2100
Chicago, IL 60603

(312) 577-7000

(312) 577-7007 (fax)

Attorneys for Patent Owner