

Filed: September 26, 2025

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

BMW of North America, LLC,
Petitioner,

v.

Foras Technologies Ltd.,
Patent Owner.

IPR2024-01346
U.S. Patent No. 7,627,781

PETITIONER'S REPLY TO PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

	Page
I. Introduction.....	1
II. Petitioner’s Statements of Fact Remain Undisputed.....	1
III. Claims 1-30 Are Unpatentable	2
A. Patent Owner Misconstrues Transferring the Role of the Boot Processor to the Spare Processor During System Runtime.....	2
B. Ground 1: Fox Discloses or Renders Obvious Assigning a Role of Boot Processor	8
C. Ground 2 Renders Obvious Assigning a Role of Boot Processor	13
D. Ground 2 Renders Obvious Transferring the Role of Boot Processor to the Spare Processor.....	13
E. Grounds 1 and 2 Render Obvious Dependent Claim 2	14
F. Grounds 1 and 2 Render Obvious Dependent Claim 14.....	16
G. Patent Owner Does Not Separately Address Independent Claims 9, 16, 25, and Other Dependent Claims.....	18
IV. The Board Should Give Mr. Jones’s Declaration Little Weight.	18
V. Conclusion	21

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Anacor Pharms. v. Iancu</i> , 889 F.3d 1372 (Fed. Cir. 2018)	9
<i>Elbit Sys. of Am., LLC v. Thales Visionix, Inc.</i> , 881 F.3d 1354 (Fed. Cir. 2018)	2
<i>In re Fritch</i> , 972 F.2d 1260 (Fed. Cir. 1992)	16
<i>Gemtron Corp. v. Saint-Gobain Corp.</i> , 572 F.3d 1371 (Fed. Cir. 2009)	2
<i>Icon Health & Fitness, Inc. v. Strava, Inc.</i> , 849 F.3d 1034 (Fed. Cir. 2017)	2
<i>Koninklijke Philips N.V. v. Google LLC</i> , 948 F.3d 1330 (Fed. Cir. 2020)	9
<i>KSR Int’l Co. v. Teleflex, Inc.</i> , 550 U.S. 398 (2007).....	9, 19
<i>Merck & Co. v. Biocraft Lab’ys, Inc.</i> , 874 F.2d 804 (Fed. Cir. 1989)	16
<i>In re NuVasive, Inc.</i> , 842 F.3d 1376 (Fed. Cir. 2016)	17
<i>Randall Mfg. v. Rea</i> , 733 F.3d 1355 (Fed. Cir. 2013)	9
Rules	
Fed. R. Evid. 702(b).....	18
Fed. R. Evid. 703	11

Regulations

37 C.F.R. § 42.65(a).....	18
---------------------------	----

I. Introduction

The Petition demonstrates that the prior art renders obvious transferring the role of the boot processor to a spare processor during system runtime, as claimed and described by the '781 patent. Patent Owner misconstrues this limitation as requiring the new boot processor to perform system initialization after system reboot. Paper 11 ("POR"), 9-11. Patent Owner also relies on unsupported attorney argument that cannot overcome the Petition's Statements of Fact or showing of unpatentability of the disputed claims that are supported by Petitioner's expert testimony and prior art in the record. *Id.*, 3-5, 14-16. And Patent Owner does not dispute any of the motivations to combine the prior art references, as explained in the Petition and by Dr. Brogioli. *See generally id.*

II. Petitioner's Statements of Fact Remain Undisputed

Patent Owner admits facts set forth in Nos. 1 and 5 of Petitioner's Statements of Facts, denies Nos. 2, 3, 6, and 8, and asserts that Nos. 4, 7, and 9 cannot be admitted or denied. POR 3-5. In denying or declining to admit these facts, Patent Owner offers no expert testimony. Instead, Patent Owner relies solely on attorney argument characterizing certain background references cited by Petitioner and Dr. Brogioli. *Id.* By contrast, Petitioner's Statements of Facts are supported by Dr. Brogioli's declaration, which explains the teachings of the cited prior art from

the perspective of a person of ordinary skill in the art (“POSITA”). *See* Brogioli Decl. ¶¶39-54.

Because Patent Owner’s denials rest only on unsupported attorney argument, Petitioner’s Statements of Fact remain undisputed. *Gemtron Corp. v. Saint-Gobain Corp.*, 572 F.3d 1371, 1380 (Fed. Cir. 2009) (“[U]nsworn attorney argument ... is not evidence and cannot rebut ... other admitted evidence”); *Icon Health & Fitness, Inc. v. Strava, Inc.*, 849 F.3d 1034, 1043 (Fed. Cir. 2017); *Elbit Sys. of Am., LLC v. Thales Visionix, Inc.*, 881 F.3d 1354, 1359 (Fed. Cir. 2018).

III. Claims 1-30 Are Unpatentable

A. Patent Owner Misconstrues Transferring the Role of the Boot Processor to the Spare Processor During System Runtime

Patent Owner argues Petitioner’s grounds fail to render obvious transferring, during system runtime, the role of boot processor to said spare processor, as required by limitation [1c] because “the ‘*role of boot processor*’ relates to what occurs in the system *during the boot process*.” POR 9-11 (additional emphasis added). Patent Owner and Mr. Jones misconstrue transferring the role of the boot processor to the spare processor as requiring the spare processor—the new boot processor—to perform system initialization after system reboot or reset. *Id.*, 11, 13, 18; *see also* Ex. 2004 (Jones Decl.), ¶ 36 (contending prior art does not disclose switching processor identifier that persists through a reset); *id.*, ¶ 37 (contending the ’781

patent describes that information regarding the changed boot processor is persistently stored “for the next time the system reboots”); Ex. 1037 (Jones Tr.), 122:21-124:21, 138:8-136:3 (boot processor role is performed during boot-up process before operating system is running), 134:1-134:7 (the new boot processor needs to perform boot functions the next time the system reboots and “has no role during runtime”). These assertions are wrong and lack support in the claim language and the specification of the ’781 patent.

First, boot processor responsibilities are not confined to the “*boot process*.” Dr. Brogioli explains that besides performing system initialization and booting the operating system during the boot process, the boot processor “controls various system functions after the initial boot process.” Brogioli Decl. ¶¶ 44-46, 104. For example, the boot processor controls system shutdown, and “operat[es] as the last running processor during system shutdown.” *Id.*, ¶¶ 46, 104. Dr. Brogioli explains that “when a failure is detected in the boot processor, the boot processor should be promptly recovered to ensure continued performance of the boot processor functions during system runtime,” which would reduce system interruption and improve system reliability. *Id.*, ¶ 104.

Second, the ’781 patent claims require transferring the role of the boot processor to a spare processor not “*during the boot process*,” (POR 10-11 (emphasis added)), but “*during system runtime*.” Ex. 1001, 18:54-56 (limitations [1c] (“for

transferring, *during system runtime*, the role of boot processor to said spare processor”) (emphasis added)), 19:24-26 (limitation [9c] (“changing, *during system runtime*, the assignment of the role of boot processor to said spare processor”) (emphasis added)), 20:48-50 (limitation [25f] (“causing said spare processor to *dynamically assume* the role of boot processor”) (emphasis added))).

Nothing in the claims requires boot processor role transfer to “persist through a reset” or that the spare processor (new boot processor) perform initialization “next time the system reboots.” *See* POR 11-14; Jones Decl. ¶ 37. In fact, if the transfer requires the spare processor to perform initialization during the boot process after system reboots, the spare processor would only function as a boot processor after the operating system shuts down and restarts. This is contrary to the claims reciting “during system runtime.” *See* Ex. 1001, 18:31-37 (describing the transfer “occurs *dynamically during system runtime*, rather than requiring that the system be interrupted (e.g., shutdown and restarted, or crashed.)” (emphasis added), 19:27-31 (limitation 9[d]) (reciting the *operating system* recognizes the spare processor as the boot processor) (emphasis added)).

And nowhere does the ’781 specification describe that transferring the role of the boot processor requires storing information of the new boot processor such that the information is persistently stored “for the next time the system reboots.” POR 13-14; Jones Decl. ¶ 37.

Patent Owner recognizes the '781 patent discloses a “device tree” for storing information identifying the boot processor and spare processor, but Patent Owner asserts “assigning the **role of boot processor** is more than merely: (1) copying the **state of the boot processor** into the spare, (2) updating the **device tree** to identify the spare as the boot processor, or (3) changing the **LID** of the spare processor.” POR 11 (emphases added). But Patent Owner provides no explanation or evidence for this assertion. *Id.* Indeed, the '781 patent discloses these steps are performed in transferring the role of the boot processor to the spare processor *during system runtime*.

The '781 patent describes that the “device tree” includes information designating the boot processor and the spare processor, as shown in Table 1 below:

TABLE 1

Device	Status	Lockstep Enabled	Boot Processor?	Spare for Boot Processor?
Processor A	Present, Enabled, and Functioning	Yes	Yes	No
Processor B	N/A	Yes	No	Yes
Processor C	Present, Enabled, and Functioning	Yes	No	No
...	...	...	...	...

Ex. 1001, 13:20-14:3; *see also id.*, 12:59-65 (Firmware builds the “device tree” so the OS can know what is installed in the system).

During operating system runtime, upon detection of a loss of lockstep (LOL) in the system, the firmware accesses this “device tree” to determine if the processor for which LOL is detected is the system’s boot processor. *Id.*, 13:60-66; *see also id.*, 4:54-62, 17:34-41. If the processor having LOL is the boot processor, the firmware determines from this “device tree” the spare processor. *Id.*, 14:1-3; *see also id.*, 4:54-62, 16:31-35, 17:41-46.

To transfer the role of boot processor to the spare processor during system runtime, the firmware PAL copies the state of the “good” one of the boot processor lockstep pair into the spare processor and updates the “device tree” to identify the spare processor as the boot processor (e.g., changing the LID of the spare processor to that of the boot processor) to cause the OS to recognize the spare processor as the boot processor. *Id.*, 16:52-61, 17:46-55; *see also id.*, 15:8-20 (“In performing this switch the LID register of the CPU that lost lockstep is copied into the LID register of the spare CPU and the spare CPU pair will then assume the ID of the failed CPU and start responding to interrupts directed to that ID.”), 16:11-16 (“[T]he LIDs would be swapped, that is the LID register of the boot processor that lost lockstep is copied into the LID register of the now spare processor pair ... , and the LID register of this spare processor pair would be copied into the LID register of the original boot processor.”).

The Petition shows that Fox, Safford, and Arai (Ground 1), or alternatively Fox, Safford, Landry, and Arai (Ground 2), render obvious such transferring the role of the boot processor to the spare processor during system runtime, as explained in the Petition. Paper 1 (“Pet.”), 9-50; *see below* Sections III.B-III.D.

Patent Owner’s contrary arguments rest on conflating a **boot device** with the **boot processor**. Mr. Jones asserts that “the ’781 specification describes a **different approach** wherein, for example, information regarding the **boot device path** can be changed during runtime and that is then persistently stored ‘**for the next time the system reboots.**’” Jones Decl. ¶ 37 (emphases added); Ex. 1001, 12:35-58, 8:4-31; POR 13-14.

The “different approach” emphasized by Patent Owner and Mr. Jones has nothing to do with switching the boot processor. POR 13-14; Jones Decl. ¶ 37. Patent Owner and Mr. Jones cite to column 12, lines 54 to 56 of the ’781 patent specification describing “EFI may be invoked to change the **boot device path** ‘for the next time the system reboots.’” POR 13-14 (emphasis added); Jones Decl. ¶ 37. But this describes changing the **boot device** path and does not describe changing the role of the **boot processor**. And the ’781 patent describes that “**boot devices** ... typically include[] the **disk**. The EFI reads the **disk** to load a program into memory, typically referred to as **an operating system loader.**” Ex. 1001, 12:14-20 (emphases added). As such, Patent Owner and Mr. Jones’s alleged “different approach” for

transferring the role of the boot processor merely describes that firmware EFI can change the path for the boot device (e.g., disk) that stores an OS loader program. It has nothing to do with switching the boot processor.

Patent Owner also asserts that “the ’781 specification further discloses, for example, an embodiment where ‘commands for assigning the boot processor and/or spare processor’ are pushed ‘into NVRAM so they will be honored on the next reset.’” Ex. 1001, 14:4-32; POR 13-14; Jones Decl. ¶ 37. But this does not disclose transferring/changing the role of the boot processor to the spare processor dynamically *during system runtime*. Ex. 1001, 18:31-36, FIG. 6 (block 603) (“chang[ing] in the role of boot processor occurs dynamically during system runtime, rather than requiring that the system be interrupted (e.g., shutdown and restarted, or crashed.”); *see above* Section III.A.

B. Ground 1: Fox Discloses or Renders Obvious Assigning a Role of Boot Processor

Patent Owner and Mr. Jones admit Fox discloses an IA-32 multiprocessor system “provided within basic input output system (BIOS) of the IA-32 architecture.” POR 15; Jones Tr.77:16-18, 79:22-80:11, 83:5-12, 92:14-93:3. Patent Owner contends that “Fox does not disclose a boot processor or the BSP flag,” and the Petition improperly relies on Ex. 1008, IA-32 Intel® Architecture Software

Developer's Manual, Vol. 3: System Programming Guide, ("IA-32") and Ex. 1028, MultiProcessor Specification ("MP Specification"). POR 7-8. Not so.

The Petition and Dr. Brogioli explain how a POSITA, based on background knowledge of the IA-32 multiprocessor system, would have understood that Fox's IA-32 multiprocessor system would have, or it would have been obvious to have, designated a boot processor to perform system initialization. Pet. 15-17, 22; Brogioli Decl., ¶¶44-46, 61-62, 100-106, 118; *Koninklijke Philips N.V. v. Google LLC*, 948 F.3d 1330, 1337-38 (Fed. Cir. 2020) (finding that the Board properly relied on expert evidence, corroborated by prior art, in concluding that a claim limitation was not only in the prior art, but also within the general knowledge of a skilled artisan); *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398, 401, (2007) (the obviousness analysis requires an assessment of the "background knowledge possessed by a person having ordinary skill in the art"); *Randall Mfg. v. Rea*, 733 F.3d 1355, 1362-63 (Fed. Cir. 2013) ("the knowledge of such an artisan is part of the store of public knowledge that must be consulted when considering whether a claimed invention would have been obvious"); *Anacor Pharms. v. Iancu*, 889 F.3d 1372, 1380-81 (Fed. Cir. 2018) (petitioners may rely on expert testimony and supporting documentation to show how a skilled artisan would have understood the prior art).

Fox discloses its multiprocessor system is configured with IA-32 architecture and discloses "system boot" utilizing the BIOS or "BIOS boot." Pet. 22; Fox,

Abstract, FIG. 1, [0009], [0014]-[0015], [0034]-[0035], [0040], claims 4-6, 15, 16; Brogioli Decl. ¶¶ 59-61. Dr. Brogioli explains that it was conventional and well known to a POSITA that one of the processors in a multiprocessor system would be the boot processor (bootstrap processor or BSP) that initializes the system, including initializing the other processors and booting the operating system. Brogioli Decl. ¶ 44. Dr. Brogioli also explains that Intel's multiprocessor system configured with IA-32 architecture, such as Fox's IA-32 system, selects one of the processors as the boot processor, and as part of the selection process, a BSP flag of the selected processor is set to 1 to indicate it is the boot processor. Brogioli Decl. ¶¶ 44-46; IA-32, 238. And after the initialization is completed, each processor is assigned and can be identified by a unique local APIC ID. Brogioli Decl. ¶ 46; IA-32, 244; MP Specification, 33. Based on such background knowledge and Fox's express disclosure of BIOS boot, Dr. Brogioli further explains that a POSITA would have understood that Fox's IA-32 system would have, or it would have been obvious to have, designated a boot processor to perform its system BIOS boot as in an IA-32 multiprocessor system. Brogioli Decl. ¶¶ 61, 100, 118; IA-32, 238; Schelling, [0003]-[0005]; MP Specification, 17.

Patent Owner offers no evidence to rebut Dr. Brogioli's explanation and evidence. *See generally* POR; Brogioli Decl. ¶¶ 44-46, 59-62, 100, 118. Patent Owner even acknowledges that selecting a bootstrap processor in multiprocessor

systems was a “common approach.” POR 4-5 (acknowledging Schelling’s disclosure of selecting a boot strap processor as “a common approach” and Datta’s disclosure of “usually” selecting a single bootstrap processor in a multiprocessor system). Nor does Patent Owner dispute that it would have been obvious to a POSITA that Fox’s system would have designated a boot processor to perform the BIOS boot, which was conventional and well known. Brogioli Decl. ¶¶ 44-46, 61-62, 118; Jones Tr., 96:3-98:9 (a POSITA would have understood that one of the approaches to perform BIOS boot would use a processor executing BIOS).

Patent Owner contends that two of the background references cited by Dr. Brogioli—IA-32 and MP Specification—have not been shown as prior art of the ’781 patent. POR 7-8; *see* Brogioli Decl. ¶¶ 44-47, 61 (discussing using a boot processor to initialize a multiprocessor system was known, citing Schelling, Jayakumar, Datta, IA-32, MP Specification, Landry, Jin). But Patent Owner has not filed any objection to or a motion to exclude these references. These documents are Intel’s published technical specifications, routinely relied upon by skilled artisans. *See, e.g.*, Ex. 1032, 968 (In support of the *ex parte* reexamination of the ’781 patent, Dr. Wolfe cited to MP Specification in his declaration); *id.*, (citing *Intel IA-64 Architecture Software Developer’s Manual*, Revision 1.1 (July 2000) and *Intel, Extensible Firmware Interface Specification*, Version 1.10 (Dec. 1, 2002) (“Intel EFI

Specification”). Experts may rely on such documents regardless of admissibility. Fed. R. Evid. 703.

Additionally, Patent Owner argues the Petition engages in circular reasoning by “contend[ing] that the registers or tables in Fox that purportedly store flags such as a boot processor flag, are ‘created by the boot processor.’” POR 8-9. Patent Owner conflates *storing* information with *creating* information.

The Petition explains that a POSITA would have understood that Fox’s IA-32 system—not the boot processor—would have designated one of its processors as the boot processor. Pet. 15-17, 22; Brogioli Decl. ¶¶44-46, 100, 118; Fox, [0015], [0034]; IA-32, 238-39. The Petition further explains that Fox discloses storing processor identifications in a register or table (e.g., BIOS table, ACPI table, MP table), and a POSITA would have understood that Fox’s IA-32 system would store the processor identifications and software flags, including those of the boot processor and the spare processor (“information”), in such register or table accessible to the BIOS and operating system. Pet. 22-23; Fox, [0026], [0033], [0035], [0038], [0041], [0045]-[0046], Claims 2. IA-32, 244, 256, 709-27; MP Specification, 38, 69, 72-73, 76; Brogioli Decl. ¶¶ 119-120; *see* Ex. 1001, 4:54-58, 8:43-48, 13:20-14:3, 17:37-40, 18:17-18 (the ’781 patent discloses storing information identifying the processors in a “device tree” accessible by the system’s firmware and operating system).

C. Ground 2 Renders Obvious Assigning a Role of Boot Processor

Patent Owner argues Ground 2 fails for the same reasons as Ground 1. POR 17; Jones Decl, ¶¶ 41-44.

Patent Owner does not dispute that Landry discloses identifying logical CPU0 as the boot processor and Fox combined with Landry renders obvious assigning a role of boot processor—logical CPU0—to one of said plurality of processors. POR 17; Pet. 33-36; Brogioli Decl. ¶¶ 47, 70-71, 147-149, 155.

Patent Owner asserts that the identification of the boot processor, logical CPU0, does not have any impact on which processor performs the role of the boot processor. POR 17. But this is inconsistent with its admission that “[o]n or before October 25, 2004, it was known that a boot processor of a multiprocessor system could be identified as logical CPU0.” POR 5.

D. Ground 2 Renders Obvious Transferring the Role of Boot Processor to the Spare Processor

With respect to Ground 2, Patent Owner asserts that switching the identification of the boot processor, APIC ID or logical CPU0, to the spare processor does not transfer the role of boot processor to a spare processor during system runtime because such identifications do not persist on reboot. POR 18-19; Jones Decl. ¶ 43. As discussed above, Patent Owner’s interpretation of transferring the role of the boot processor as requiring the new boot processor identification to persist

and perform initialization after the reboot lacks support in the claim language and specification of the '781 patent. *See above* Section III.A.

Patent Owner also argues Landry does not support Petitioner's argument regarding transferring "*during system runtime*" the role of boot processor to a spare processor. POR 19. But this fails to address Petitioner's Fox-Landry-Safford-Arai (Ground 2) combination where Fox discloses switching the failing processor to the spare processor during system runtime, Fox-Landry renders obvious assigning boot processor role (logical CPU0) to one of its processors, and Fox-Landry-Safford-Arai renders obvious transferring the role of the boot processor (logical CPU0) to the spare processor during system runtime. Pet. 33-37; Brogioli Decl. ¶¶ 145-158.

E. Grounds 1 and 2 Render Obvious Dependent Claim 2

The Petition explains that Arai discloses storing information identifying processors (e.g., ACPI table or MP table) in the system in a main memory, Fox discloses a non-volatile system memory for storing BIOS, and a POSITA would have found it obvious to store Fox's information identifying the boot and spare processors (e.g., register or table) in its non-volatile system memory. Arai, [0019]-[0022], [0025]-[0026]; Fox, [0026], [0028], [0033], [0035], [0038], [0041], [0045]-[0046], Claims 2, 13, FIG. 1; IA-32, 242, 244, 256, 709-727; MP Specification, 38, 69, 72-73, 76; Pet. 22-24, 50; Brogioli Decl. ¶¶ 52, 120-124, 198.

Patent Owner sets up a straw man and argues that in Petitioner's combinations, "the register or table are created by the BIOS and stored in volatile RAM." POR 12; *id.*, 13 ("the table or register ... is expressly stored in volatile memory."). This does not address Petitioner's combinations that store processor information in Fox's non-volatile system memory. Indeed, Patent Owner concedes that Fox's BIOS would be stored in a non-volatile memory. POR 13; Jones Decl. ¶ 55 (admitting Fox includes a BIOS stored in non-volatile system memory). Dr. Brogioli's opinion that it would be obvious to use Fox's non-volatile system memory to store information identifying the processors remains undisputed. Brogioli Decl. ¶¶ 52, 120-124, 198.¹

Patent Owner assumes RAM is volatile and thus cannot be the claimed "non-volatile data storage." POR 13 ("there is no suggestion that RAM is non-volatile data storage"). This assumption is inconsistent with the '781 patent, which discloses the information identifying the processors—the "device tree"—is stored in a SRAM, "which is RAM that is reinitialized." *See* Ex. 1001, 8:44-48, 11:37-42, 12:59-61.

¹ Patent Owner does not dispute claim 10, which also recites "non-volatile data storage." POR 11-14, 19; Jones Decl. ¶ 55.

Brogioli Decl. ¶ 124; Ex. 1001, 14:25-29 (showing some RAM can be non-volatile, such as an NVRAM that saves changes to the next reset). In fact, Patent Owner contends NVRAM, which is a RAM, is non-volatile because information pushed “into NVRAM ... will be honored on the next reset.” POR 11, 14; Ex. 1001, 14:4-32.

Relying on attorney argument, Patent Owner argues the ’781 patent’s disclosure of storing the “device tree” in a RAM is irrelevant for the “*non-volatile data storage*” limitation in claim 2. POR 13. Patent Owner again rests on the erroneous understanding that changing information regarding the boot device (e.g., a disk) path changes the boot processor that persists through system reboot. Ex. 1001, 12:35-58; *see above* Section III.A.

F. Grounds 1 and 2 Render Obvious Dependent Claim 14

Patent Owner’s arguments regarding claim 14 rely solely on attorney argument (POR 14-16)², which cannot overcome the Petition’s showing—supported

² Patent Owner does not dispute claims 6 and 23, which also recite “an active spare.” POR 18-19.

by expert testimony—that Grounds 1 and 2 each render obvious claim 14. Pet. 50-52, 56-57; Brogioli Decl. ¶¶ 201-203, 217-218, 227; Fox, [0015], [0034]-[0035].

Patent Owner contends that Fox does not disclose “an active spare” or “idling an active spare processor,” because “Fox only teaches a hot-spare processor” that “remains hidden until one of the primary processors fails.” POR 14-15; Fox, [0037]. This is incorrect.

The prior art is relevant for all that it teaches. *See In re Fritch*, 972 F.2d 1260, 1264 (Fed. Cir. 1992); *Merck & Co. v. Biocraft Lab ’ys, Inc.*, 874 F.2d 804, 807 (Fed. Cir. 1989). Fox discloses an alternative embodiment that enables the hot spare processor “on-the-fly by the OS” with a driver-OS-SMI handshake. Pet. 50-51; Brogioli Decl. ¶ 201; Fox, [0035].

The Petition explains that this alternative embodiment discloses or renders obvious configuring one of the active processors to be the spare processor dynamically as needed while the operating system is running, and a POSITA would have understood such active processor as “an active spare.” Pet. 51-52; Brogioli Decl. ¶¶ 201-203; Fox, [0035]; *see also* Brogioli Decl. ¶ 202 (“A POSITA would have been motivated to use ‘an active spare’ in Fox because allowing the spare processor to be available to the operating system as an active spare improves the processing capacity of the system when no processor failure occurs.”).

Fox discloses during system BIOS boot (“during system boot-up”), the BIOS (“system firmware”) identifies the spare processor by setting a software tag. Fox, [0015], [0034]; Brogioli Decl. ¶¶ 201-203. The Petition also explains that a POSITA would have understood that, for this alternative embodiment, Fox’s “active spare” would have been initialized as an active processor with a software tag indicating it is a spare processor during system BIOS boot. Pet. 51, 56; Fox, [0034]; Brogioli Decl. ¶¶ 201-202, 217-218.

G. Patent Owner Does Not Separately Address Independent Claims 9, 16, 25, and Other Dependent Claims

Patent Owner argues Grounds 1 and 2 do not render independent claims 9, 16, and 25 obvious for the same reasons as claim 1. POR 19-20. Patent Owner does not separately dispute Petitioner’s showing as to these independent claims or any of the dependent claims 3-8, 10-13, 15, 17-24, and 26-30. POR 19-21; *In re NuVasive, Inc.*, 842 F.3d 1376, 1379-82 (Fed. Cir. 2016). These claims are unpatentable. Pet. 38-66; Brogioli Decl. ¶¶ 159-254.

IV. The Board Should Give Mr. Jones’s Declaration Little Weight.

The Board should give the declaration of Patent Owner’s expert Mr. Jones little weight because (1) Mr. Jones’s testimony lacks sufficient factual basis; and (2) Mr. Jones’s testimony does not reflect the knowledge and skill of a POSITA. Pet. 5; POR 3; Brogioli Decl. ¶¶ 28-30; Jones Decl. ¶¶ 24-26.

First, Mr. Jones interprets transferring the role of the boot processor as requiring the new boot processor to initialize the system after system reboot. *See above* Section III.A. This interpretation lacks support in the specification and contradicts the claims, which require the transfer to be performed during system runtime. *See above* Section III.A. His opinions also rest on the incorrect assumption that a “boot device” for storing an OS loader program is equivalent to a “boot processor.” *See above* Section III.A; Jones Decl. ¶ 37; Fed. R. Evid. 702(b); 37 C.F.R. § 42.65(a) (“Expert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little or no weight.”).

Second, Mr. Jones’s testimony does not reflect the knowledge and skill of a POSITA. He admitted his current work is litigation-based consulting, and in providing his declaration, he did not provide a CV that “lists his technical experiences in detail,” but instead provided his “expert witness CV” focusing on the cases he worked on. Jones Tr., 30:22-31:15; Ex. 2005. While he claimed that he had experience using Intel processors (not disclosed in Ex. 2005) and considered materials for those processors (Jones Tr., 29:20-32:16), and acknowledged that Fox discloses an Intel IA-32 multiprocessor system (*id.*, 77:16-18, 79:22-80:11, 83:5-12), he admitted that he “have not given any thought to” Intel multiprocessor systems prior to 2004 (*id.*, 34:4-37:5, 41:17-43:8). *But see id.*, 102:14-105:19 (Mr. Jones admitting “Intel publishes technical documentation about its processors,” and some

people working on an IA-32 processor “would have consulted Intel documentation”).

Mr. Jones also emphasized that, in forming his opinions, he considered only the documents mentioned in his declaration and the Petition itself. *Id.*, 36:4-10, 41:9-42:18; *see also id.*, 107:1-108:13 (asserting the MP Specification is irrelevant for this matter because he “can’t believe [it] was published.”).

And although Mr. Jones acknowledged Fox discloses an IA-32 multiprocessor system that performs BIOS booting (Jones Tr., 92:14-93:3), and that a POSITA would have known various approaches for BIOS booting, including using a processor to execute the BIOS (*id.*, 94:22-98:9), he nevertheless asserted that a POSITA would have considered how Fox’s system boots as irrelevant (*id.*, 81:12-83:4). He even testified that a POSITA “wouldn’t have to know what a boot processor is” (*id.*, 69:7-22), treating a POSITA’s background knowledge of boot processors as irrelevant (*id.*, 74:2-75:1).

Accordingly, Mr. Jones’s testimony reflects that he considered only a small subset of the record and offered opinions without the factual basis or technical perspective required of a POSITA. *KSR*, 550 U.S. at 421 (“A person of ordinary skill is also a person of ordinary creativity, not an automaton.”).

V. Conclusion

For the foregoing reasons, and as explained in the Petition, the Board should find all challenged claims of the '781 patent unpatentable.

Dated: September 26, 2025

Respectfully submitted,

/Lionel M. Lavenue/

Lionel M. Lavenue, Lead Counsel
Reg. No. 46,859

CERTIFICATION UNDER 37 C.F.R. § 42.24(d)

Pursuant to 37 C.F.R. § 42.24(a)(1)(i), the undersigned hereby certifies that the foregoing **Petitioner's Reply to Patent Owner's Response** contains 4,446 words, excluding the parts exempted under 37 C.F.R. § 42.24(a), as measured by the word-processing system used to prepare this paper.

/Lionel M. Lavenue/

Lionel M. Lavenue

Reg. No. 46,859

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the foregoing **Petitioner's Reply to Patent Owner's Response** was served on September 26, 2025, via email directed to counsel of record for the Patent Owner at the following:

Brett Cooper
bcooper@bc-lawgroup.com

Robert A. Auchter
robert@auchterlaw.com

Seth Lindner
slindner@bc-lawgroup.com

Foras_Counsel@bc-lawgroup.com

Dated: September 26, 2025

By: /Daniel E. Doku/
Daniel E. Doku
Senior Litigation Paralegal
FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, LLP