

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)

Hearing

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

No. IPR2024-01347 | U.S. Patent No. 7,642,302

Patent Owner's Demonstrative Exhibits

December 5, 2025

Demonstrative Exhibit - Not Evidence

302 Claim 1

[1pre] A method comprising:

[1a] detecting loss of lockstep (LOL) for a processor in a multi-processor system;

[1b] determining that the processor for which said LOL is detected is **assigned a role of boot processor**;

[1c] changing a system identifier, used by the system's operating system to recognize the boot processor, of said boot processor for which said LOL is detected, to a different value, which causes the system's operating system to not recognize the processor for which said LOL is detected as the boot processor; and

[1d] changing a system identifier of the spare processor to that of the boot processor, thereby **switching the role of boot processor to a spare processor** without shutting down the system's operating system.

781 Claim 1

[1pre] A system comprising:

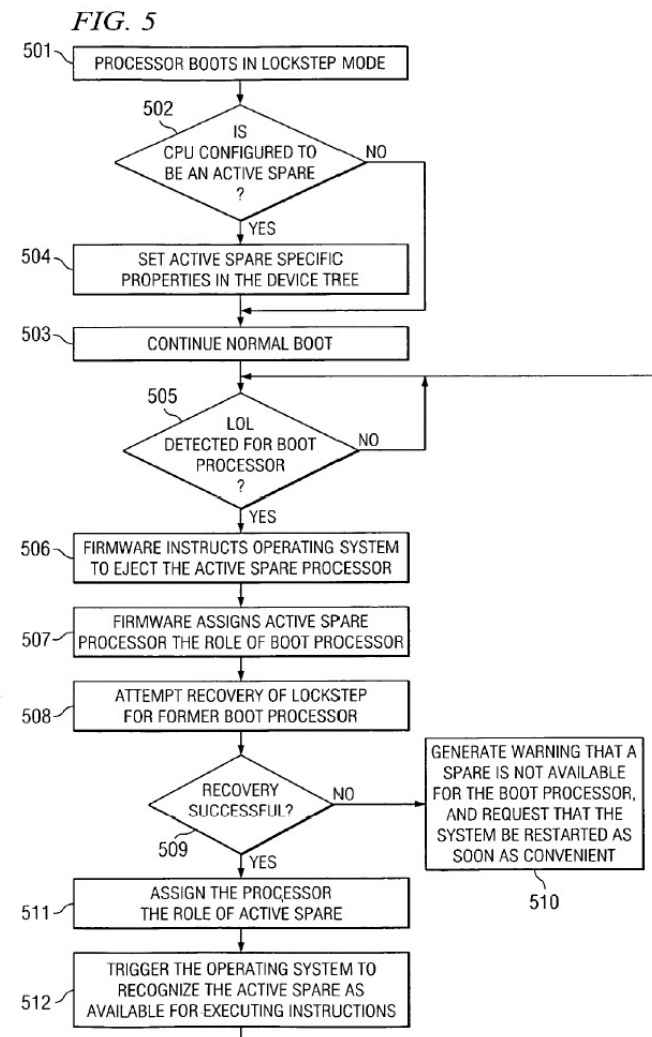
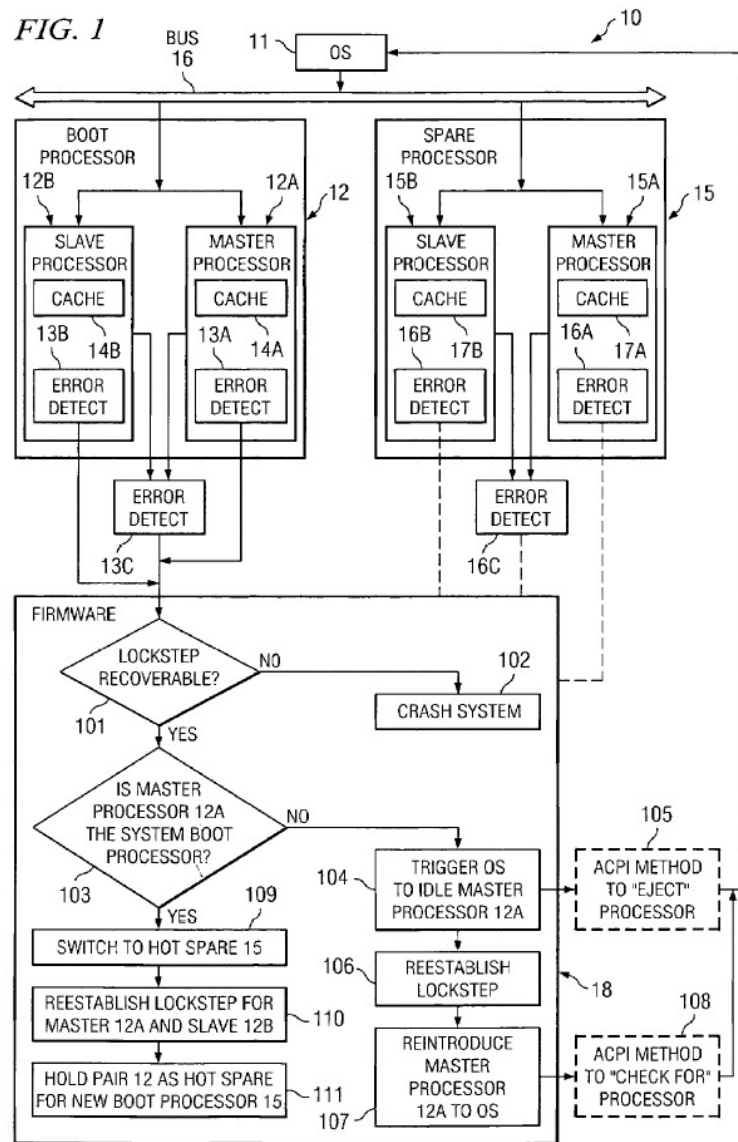
[1a] a plurality of processors;

[1b] data storage storing information that **assigns a role of boot processor** to one of said plurality of processors and assigns a role of spare processor to another of said plurality of processors; and

[1c] logic operable, responsive to detecting loss of lockstep for said boot processor, for **transferring, during system runtime, the role of boot processor** to said spare processor,

[1d] wherein said another of said plurality of processors that is assigned said role of spare processor is held in reserve as a hot spare processor until said loss of lockstep is detected for said boot processor.

781/302 Overview



'781 Patent: Ground 1

[1b] “data storage storing information that assigns a role of boot processor to one of said plurality of spare processors”

[1b] Fox does not disclose “information that assigns a role of boot processor”

Petition:

“A POSITA would have understood that Fox’s IA-32 system would have set the **BSP flag** of the designated boot processor to 1 (“assigns a role of boot processor”).”

781 Pet. at 22

POR:

Fox does not disclose the “BSP flag,” which appears only in Exhibit 1008.

Exhibit 1008 has not been shown to be prior art.

No evidence that a POSITA would have been aware of Exhibit 1008 at the time of invention.

781 POR at 7-8

[1b] Alleged “data storage” does not “assign a role of boot processor”

Petitioner contends that the registers or tables in Fox that purportedly store flags such as a boot processor flag, are “created by the boot processor.”

Yet claim 1 recites “data storage storing information that assigns a role of boot processor,” and Petitioner has provided no explanation as to **how a role of boot processor can be assigned by a boot processor itself.**

781 POR at 8-9

Jones ¶ 36: “because the table of APIC IDs is created by the BIOS during bootup, the APIC IDs themselves does not assign the role of boot processor”

[1c] Switching APIC IDs does not transfer role of boot processor

Petitioner relies on ARAI for switching APIC IDs:

Arai discloses replacing an active processor detected with an error with a reserved inactive processor in a multiprocessor system. Arai, Abstract, [0002], [0008]-[0009], [0019]-[0024]. Arai discloses that during system initialization, the BIOS creates a table that identifies each processor in the system and its designated operational state, i.e., active or disabled. *Id.*, [0019]-[0022]. If the BIOS determines the error requires replacement of the processor, and if there is a reserved inactive processor, the identification numbers of the active processor in error and the reserved inactive processor are switched. *Id.*, [0022]-[0023], [0025].

781 Pet. at 28

[1c] Switching APIC IDs does not transfer role of boot processor

Petitioner relies on ARAI for switching APIC IDs:

Arai discloses switching the logical APIC IDs of the active processor in error with the reserved processor. *Id.*, [0023]-[0024], FIGS. 4a-4b. A POSITA would have

understood that Arai's logical APIC IDs would be used by Arai's operating system to recognize the processor in error. *Id.*, [0020]-[0022], [0025]-[0026]. A POSITA would have further understood that such switching would have changed the logical APIC ID of the processor in error to the logical APIC ID of the reserved processor.

Id.; Brogioli, ¶134.

781 Pet. at 28-29

[1c] Switching APIC IDs does not transfer role of boot processor

No showing that switching logical APIC IDs in Arai has any impact on which processor performs any boot processor role.

APIC IDs in Arai are stored in RAM.

[1c] Switching APIC IDs does not transfer role of boot processor

“[A] POSITA would readily recognize, based on Arai’s disclosure of how the table of APIC IDs is created by the BIOS during bootup and written to RAM, that changes to logical identifiers made during runtime have no impact on which processor assumes the role of boot processor” Jones ¶ 36.

“During the initialization and boot process, the **BSP executes the BIOS code to boot the operating system, initialize the APICs**, and build the MP configuration table to communicate to the operating system about the APICs and APs.” Brogioli ¶ 46

“**After the initialization is completed**, each processor is assigned and can be identified by a unique local APIC ID.” Brogioli ¶ 46

“Role of boot processor”

The “role of boot processor” relates to what occurs in the system during the boot process.

- “[T]he firmware assigns active spare processor the role of boot processor . . . Further, the firmware updates the device tree to identify the active spare as the boot processor, and performs any other configuration changes (e.g., changing the LID of the active spare, etc.) to cause the OS to recognize the active spare as the boot processor,” (781 patent at 17:45-55)
- Changes to processor roles can change settings in NVRAM that are “honored on the next reset.” (781 patent at 14:4-31)
- A processor identified as a “hot spare” in the device tree does not show up in the static ACPI tables that are built when the system boots. (781 patent at 13:36-55)

781 POR at 10-11

“Role of boot processor”

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

781 Reply at 3



“In an [*inter partes* review], the petitioner has the burden from the onset to show with particularity why the patent it challenges is unpatentable.”
Harmonic Inc. v. Avid Tech., Inc., 815 F.3d 1356, 1363 (Fed. Cir. 2016).

'781 Patent: Ground 1 – Claim 2

“Non-volatile data storage”

[2] The system of claim 1 wherein said data storage is non-volatile data storage

While Fox discloses a system memory for storing the BIOS, Fox does not expressly disclose the processor register or table is stored in the system memory. Fox, [0028], FIG. 1. Arai discloses this. Brogioli, ¶¶67, 121-122.

The '781 Patent similarly discloses a device tree or ACPI table that is stored in a RAM, accessible to the firmware, and designates which processor is the boot processors. Ex. 1001, 8:43-54, 12:59-65, 13:20-44; Brogioli, ¶124.

A POSITA would have understood Fox's system memory would be "non-volatile" because it was common and well known that the BIOS would be embedded in a non-volatile memory (e.g., ROM). *See* Fox, [0028], FIG. 1; MP Specification, 38 (storing MP configuration table within the BIOS read-only memory space); Brogioli, ¶¶197-198.

[2] The system of claim 1 wherein said data storage is non-volatile data storage

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.

781 Reply at 14

[2] The system of claim 1 wherein said data storage is non-volatile data storage

X

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.").

781 Reply at 14

[2] The system of claim 1 wherein said data storage is non-volatile data storage

Dr. Brogioli:

“That’s what’s happening **in the combination** is, just as Arai says in paragraph 21 . . . that the BIOS goes and creates a table in that main memory during the start-up in the RAM. So, the BIOS is coming out of this read only memory, running and creating the table in the RAM here.”
EX2006 at 111:12-112:12

Mr. Jones:

“An identifier, such as Arai’s APIC ID, stored in RAM would not persist through a reset.” EX2004 ¶ 35.

Fox/Arai combination



[0021] FIG. 2 is a block diagram (50) of a computer system and the designation of the processors in the system. The computer system has four physical processors CPU₀ (60), CPU₁ (70), CPU₂ (80), and CPU₃ (90), with each physical processor having two associated logical processors. For example, physical CPU₀ (60) has logical CPU₀ (62) and logical CPU₄ (64), physical CPU₁ (70) has logical CPU₁ (72) and logical CPU₅ (74), physical CPU₂ (80) has logical CPU₂ (82) and logical CPU₆ (84), and physical CPU₃ (90) has logical CPU₃ (92) and logical CPU₇ (94). The BIOS creates a table in the main memory (RAM) during start-up. In one embodiment, the table is identified as an ACPI (Advanced Configuration and Power Interface) MAPIC (Multiple Advanced Programmable Interrupt Controller) table, where this table is a structure used by ACPI to describe

Arai (EX1007) at [0021]

[2] The system of claim 1 wherein said data storage is non-volatile data storage

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”). ’

781 Reply at 15

Mr. Jones:

“An identifier, such as Arai’s APIC ID, stored in RAM would not persist through a reset.” EX2004 ¶ 35.

'781 Patent: Ground 2 – Fox/Safford/Landry/Arai

Boot Processor

No “data storage” that “assigns a role of boot processor”

Petitioner again relies on Fox’s “register or table,” which is stored RAM

Petitioner has not shown that the “identifications and software tags” that are stored in Fox’s register or table (in RAM) have any impact on which processor performs the role of boot processor.

781 POR at 18

No transfer, during runtime, of role of boot processor

Landry only disclose assigning processor roles during system initialization, which occurs **before** system runtime.

Designation of a boot processor in Landry is only retained until the “system is power cycled” or rebooted, at which point Landry’s self-testing process is repeated. EX1011 at 6 (col 2:20-23); EX2006 at 88:16-90:16.

781 POR at 19

'302 Patent: Grounds 1 & 2 – Dependent Claims 11, 12, 17, 24

Not in Petition

Petition lacks any argument for claims 11, 12, 17, and 24

This Board should reject Petitioner's attempts to assert invalidity challenges to claims 11, 12, 17 and 24 because the Petition is devoid of any arguments applying the prior art to claims 11, 12, 17 or 24.

“[T]he Board should ... not have to decode a petition to locate additional arguments beyond the ones clearly made.” *Netflix, Inc. v. DivX, LLC*, 84 F.4th 1371, 1377 (Fed. Cir. 2023). “It is of the utmost importance that petitioners in the IPR proceedings adhere to the requirement that the initial petition identify ‘with particularity’ the ‘evidence that supports the grounds for the challenge to each claim.’” *Intelligent Bio-Systems, Inc. v. Illumina Cambridge, Ltd.*, 821 F.3d 1359, 1369 (Fed. Cir. 2016). The Federal Circuit has further “held that the Board erred

Petition lacks any argument for claims 11, 12, 17, and 24

X

Claim	Limitation	Ground 1	Ground 2
11	[1a], [13b], [13c]	Pet. 9-25, 38-43; Brogioli Decl. ¶¶ 82-108, 111-114, 153-162	Pet. 30-33, 38-43; Brogioli Decl. ¶¶ 129-135, 138, 153-162
12	[1a], [13b], [13c]	Pet. 9-25, 38-43; Brogioli Decl. ¶¶ 82-108, 111-114, 153-162	Pet. 30-33, 38-43; Brogioli Decl. ¶¶ 129-135, 138, 153-162
17	[1d], [13e], [21e] ¹	Pet. 9-22, 28-30, 43, 51-53;	Pet. 30-33, 36, 43, 51-53;

302 Reply at 21

This table is NOT in the Petition

Petition lacks any argument for claims 11, 12, 17, and 24

Claim 11	
11	The method of claim 1 wherein <u>the processor for which said LOL is detected includes a lockstep pair of processors</u> , and wherein said switching comprises: determining one of the lockstep pair of processors that is not the cause of said LOL; and copying the state of the determined one of said lockstep pair of processors that is not the cause of said LOL to said spare processor.

[1a]	detecting loss of lockstep (LOL) for a processor in a multi-processor system;
[13b]	determining one of said lockstep pair of processors that is not the cause of said LOL;
[13c]	copying the state of the determined one of said lockstep pair of processors that is not the cause of said LOL to a spare processor;

Petition lacks any argument for claims 11, 12, 17, and 24

Claim 12	
12	The method of claim 11 wherein said spare processor includes a lockstep pair of processors, and wherein said copying comprises: copying the state of the determined one of said lockstep pair of processors that is not the cause of said LOL <u>to each of said lockstep pair of processors of said spare processor.</u>

[1a]	detecting loss of lockstep (LOL) for a processor in a multi-processor system;
[13b]	determining one of said lockstep pair of processors that is not the cause of said LOL;
[13c]	copying the state of the determined one of said lockstep pair of processors that is not the cause of said LOL to a spare processor;

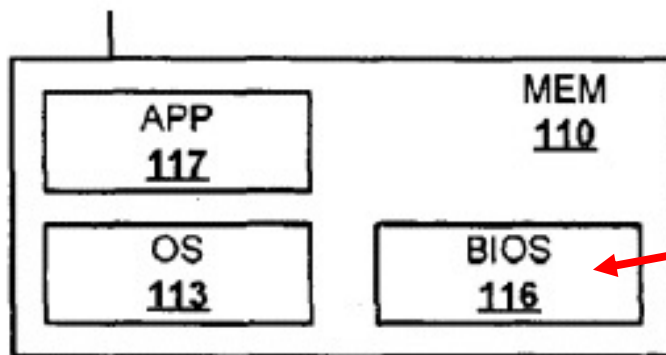
'302 Patent: Grounds 1 & 2 – Claims 6, 8, and 25

“non-volatile data storage”

Fox's "system memory" is not non-volatile

Dr. Brogioli: "A POSITA would have understood that Fox's system memory would be 'non-volatile' because it was common and well known that the BIOS would be embedded in a non-volatile memory (e.g., ROM, EEPROM, etc.)." 302 EX1002 (Brogioli Decl.) at 210.

Read Only Memory
cannot be written to



System memory in
Fox is not just BIOS

Fox, Figure 1 (partial)

302 POR at 46

Fox's "system memory" is not non-volatile

Dr. Brogioli:

"That's what's happening in the combination is, just as Arai says in paragraph 21 . . . that the BIOS goes and creates a table in that main memory during the start-up in the RAM. So, the BIOS is coming out of this read only memory, running and creating the table in the RAM here." EX2006 at 111:12-112:12

Mr. Jones:

"An identifier, such as Arai's APIC ID, stored in RAM would not persist through a reset." EX2004 ¶ 35.