IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ARM LIMITED
Petitioner,

v.

ICPILLAR LLC
Patent Owner.

U.S. Patent No. 8,924,899

Inter Partes Review No.: IPR2024-00476

REPLY TO PATENT OWNER'S PRELIMINARY RESPONSE

***Mail Stop* "*PATENT BOARD*"**
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

Page

I. Construing 1[d]: "outputting … software programming commands …." 1

II. Banerjee Discloses 1[d] 3

III. Rompaey Discloses 1[d]. 7

IV. *Sotera* Stipulation 10

CERTIFICATION OF SERVICE ON PATENT OWNER 1

Petitioner's Exhibit List

Exhibit #	***Description***
1001	U.S. Patent No. 8,924,899 ("the '899 patent")
1002	Prosecution History for U.S. Patent No. 8,924,899 (downloaded from PAIR)
1003	Declaration of Dr. Henry Houh.
1004	U.S. Patent No. 5,870,588 ("Rompaey")
1005	U.S. Pat. Appl. No. 2008/0222581 ("Banerjee")
1006	*CoWare-a design environment for heterogeneous hardware/software systems*, K. Van Rompaey, D. Verkest, I. Bolsens and H. De Man, Proceedings EURO-DAC '96, Geneva, Switzerland, pp. 252-257 (1996).
1007	*CoWare—A design environment for heterogeneous hardware/software systems,* Verkest, D., Van Rompaey, K., Bolsens, I. et al. Des Autom Embed Syst 1, 357–386 (1996).
1008	ICPillar Preliminary Infringement Contentions, '899 Patent.
1009	U.S Pat. No. 5,197,016 ("Sugimoto")
1010	U.S. Patent No. 4,922,432 ("Kobayashi")
1011	Fletcher, William I., An Engineering Approach to Digital Design, Prentice-Hall, Inc., pp. 491-505 (1984) (excerpts)
1012	ICPillar Opening Claim Construction Brief
1013	Arm Opening Claim Construction Brief.
1014	Complaint in *ICPillar v. ARM Ltd. et al,* 1:23-cv-282 (WDTX)
1015	Scheduling Order in *ICPillar v. ARM Ltd. et al,* 1:23-cv-282 (WDTX)

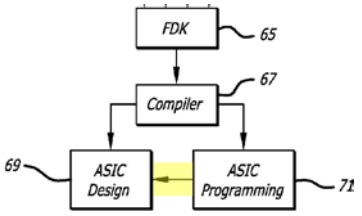
Exhibit #	***Description***
1016	Pacer Docket in *Sable Networks, Inc. et al v. Cloudflare, Inc.*, case 6:21-cv-00261, WDTX.
1017	Pacer Docket in *Demaray LLC v. Samsung Electronics Co., Ltd.*, case 6:20-cv-00636, WDTX.
1018	[this exhibit intentionally blank]
1019	[this exhibit intentionally blank]
1020	*Sotera* Stipulation

ICPillar's creative argument is unsupported by the '899 patent and the plain language of Banerjee and Rompaey. Banerjee's "optimized operating system is compiled into a reduced footprint operating system that is instantiated in the device, preferably in a non-volatile computer readable storage medium." EX-1005, 0083 (emphases herein added). Rompaey's tool "output preferably comprises … machine code for the programmable processors." EX-1004, 11:18-20. As the parties agree that IPR2024-00476 and IPR2024-00476 present the same issues, Petitioner files a common reply using citations to the '899 patent challenged in IPR2024-00476.

I. Construing 1[d]: "outputting … software programming commands …."

The '899 patent has sparse disclosure related to 1[d]. The '899 patent does not recite "outputting" software or "software programming commands." Rather, as cited by ICPillar's expert, the '899 patent recites "providing software." EX-1001, 5:1-4 ("providing software" [twice]); 5:52 ("providing software"); 5:27 ("provide software"); 13:21 ("provide accompanying software"). EX-2001, ¶¶26-28.

ICPillar's expert states that Fig. 4 shows "output[ting]… software program commands" in block ASIC Programming 71. EX-2001, ¶¶26-28. Fig. 4 teaches block 71 as an output of a "compiler 67 of the FDK" that is input into "ASIC Design 69" as shown in the highlighted arrow. ASIC Design 69 is "a circuit design such as a layout or a schematic that can be sent out to a manufacturer for fabrication."



EX-1001, 9:59-61. The '899 specification confirms the software as an input into the layout: "ASIC Programming 71 may be software to be programmed and/or stored in a fabricated ASIC…." EX-1001, 9:61-62; EX-2001, ¶28. Fig. 4 does not describe the "ASIC Programming" as a separate output of the FDK tool – rather, it is just an internal output of the compiler that serves as an input into the "layout or schematic." Fig. 4 is the sole '899 embodiment outputting software.

This sole '899 embodiment teaches two aspects of construing element 1[d]. The "outputting" cannot exclude: (1) internal outputs from "compilers" that are then used by other portions of the design; (2) that the software is within the "layout … sent out to a manufacturer for fabrication" and "stored in a fabricated ASIC." EX-1001, 9:47-62. ICPillar's position is vaguely defined, but seeks to exclude alleged "internal" outputs and exclude software stored in "layouts." POPR at 19-24, 30, 36.

ICPillar's position is improper for two reasons: (1) a claim construction should not exclude the sole embodiment (detailed above); and (2) the '899 patent does not contain any exclusionary words or teachings supporting ICPillar's positions. ICPillar attempts something even more improper than reading limitations from the specification into claims: reading limitations into claims out of thin air.

Consistent with the plain meaning, the '899 patent defines software broadly as "applications, byte code, … programs, … executable,… instructions, logic, machine code, …." EX-1001, 17:5-10. The '899 patent's sparse disclosure does not

exclude any form of software or mechanics of "providing" software. At a minimum, element [1]d covers the '899 disclosure which "provides" software in any form.

II. Banerjee Discloses 1[d]

Banerjee provides more detail of "outputting" than the '899 patent. Banerjee creates and optimizes operating system software, application software and middleware (EX-1005, [0078-0083]) and then outputs executable software.

> [0083] As indicated at 56, the optimized operating system is compiled into a reduced footprint operating system that **is** instantiated **in the** device, preferably in a non-volatile computer readable storage medium.

EX-1005, ¶0083. ICPillar and its expert rewrite paragraph 0083 in a manner that highlights the lack of merit in their position (green language is ICPillar's rewrite):

> module 24, … compiles the characteristics of the typical or "baseline" "operating system configuration," in order to analyze its expected "footprint" on the IC and how it is expected to be instantiated on the device. *Id*., [0072]-[0074], [0083].

POPR at 11, 23; EX-2001, ¶¶37, 65.

Banerjee recites "is compiled" and "is instantiated in the device" – not "expected to be." Instantiation into the device teaches 1[d]. Also, Banerjee mirrors the '899 disclosure (see above) in which "ASIC Programming 71" is "output" by a "compiler" and then is an input into ASIC Design 69 which is a "circuit design such as a layout or a schematic that can be sent out to a manufacturer for fabrication." EX-1001, 9:61-62; EX-2001, ¶¶26-28. Thus ICPillar seeks to improperly exclude a

disclosure virtually identical to the sole embodiment in the '899 patent.

ICPillar's position is incorrect for other reasons. Banerjee uses the same "provides software" language used in the '899 patent. In designing the overall system/device, "Module 24 [optimizes by] selecting the processor that most closely matches the specified parameters and providing any additional functionality in software either at the application, middleware or operating system level." EX-1005, [0078]. Banerjee's tool "provides … software components, such as application and operating software, as well as the integration of the hardware and software components" to create an SoC. EX-1005, 0008, 0027, 0032. Such "integration" teaches that the "provided software" is output to be executed on the hardware (SoC).

ICPillar's interpretation conflicts with Banerjee's purpose which is to create a tailored system using hardware and software for specific applications. Banerjee extensively describes how a user tailors hardware based on specific applications and the tradeoffs between implementing particular functions in hardware or software. For example, Banerjee's hardware may be designed to have "additional instructions targeted to a specific task…" and "include specific instructions" and thus include "application specific instruction set modifications" with "unique IP block[s] … instantiated in the SoC." EX-1005, [0068-0071].

Based on knowledge of tailored hardware (with support of new instructions), Banerjee's tool tailors the software, for example "to determine whether it is

preferable to implement certain functions in software rather than hardware." EX-1005, [0072-0073]. Banerjee configures software to efficiently use available hardware resources (e.g., memory). EX-1005, 0073.

Banerjee "match[e]s the compiled features of the operating system with the instantiated hardware components." EX-1005, 0074. Banerjee "incorporate[s] corresponding operating system modules that support the functional features of the added IP blocks in the operating system build" and "configures the operating system to support application specific functions by incorporating only selected drivers" (software). EX-1005, [0075]. Banerjee then determines the specific software needed to "support any identified peripherals" in the device. EX-1005, [0076].

Banerjee "make[s] certain tradeoffs" between functions in hardware or software and gives specific examples. EX-1005, 0077-0080. For specific hardware, Banerjee "bring[s] in the optimized driver for the instantiated hardware components" and "optimize[s] the application software" for the hardware. EX-1005, 0081.

Banerjee "optimizes" its software such as the OS and device drivers. EX-1005, 0030, 0038, 0081, 0083, claim 6 ("optimizing operating system code based on the optimized representation of an integrated circuit"). After this tailoring, Banerjee compiles software which "is instantiated in the device…." EX-1005, 0083.

ICPillar's position is that, after extensively tailoring specific hardware and specific software and making tradeoffs in functionality implementations, Banerjee's

tool just discards the tailored software. That is nonsensical. Banerjee does not describe any ability of any other tool to take advantage of the tailored and specialized hardware – such as the application-specific instructions. Under ICPillar's theory, Banerjee teaches very tailored, specific, optimized hardware – but with no ability for any device to use the optimizations with the software that will "match… the instantiated hardware components." EX-1005, 0074. ICPillar's theory renders Banerjee's entire hardware / software optimization a nullity. Optimizing the software serves no purpose if that software is to be discarded as ICPillar posits.

Banerjee teaches that "selected" and "optimized" software drivers are used for the "selected" and "instantiated hardware components." EX-1005, 0075, 0081. Device drivers are software that is specifically tailored to particular hardware. Absent output of the tailored device drivers, the hardware is not operational.

Banerjee solves "needs" by providing the "complete configuration" of an SoC with a method that "provides for … the integration of the software and hardware components." EX-1005, 0007-0008. There is no "integration" of the hardware and software components if the software is discarded as ICPillar theorizes.

There are two choices for interpreting Banerjee. Petitioner's position gives meaning to Banerjee's: (1) extensive tailoring and tradeoffs in developing hardware and software specifically designed to work together; and (2) statements that the software "is compiled" and "is instantiated in the device…" which meets Banerjee's

goal of integrating hardware and software components for an SoC. PO's position: (1) excludes the sole '899 embodiment related to 1[d]; (2) renders large portions of Banerjee to be an irrelevant nullity; (3) results in tailored hardware for which optimizations cannot be used; (4) does not result in the "integration" of hardware and software; and (5) ignores and attempts to rewrite Banerjee's express language.

III. Rompaey Discloses 1[d].

Rompaey describes a "CoWare" system for "hardware/software co-design." EX-1004, Title. Rompaey recites the claimed "outputting" succinctly:

> FIG. 2 shows the architecture of the CoWare system. … The input is a heterogeneous specification of an electronic system, the output 12 may include a net list for prior-art commercial tools for the generation of the implementation layout. **<u>This output preferably comprises</u>** structural VHDL or Verilog and **<u>machine code for the programmable processors</u>**.

EX-1004, 11:12-20. Thus, this one Rompaey passage contains far more explicit disclosure of the "outputting" of 1[d] than any '899 patent passage.

ICPillar and its expert ignore this express statement. ICPillar's expert agrees that "machine code" are instructions executed on a processor – which meets the claimed "software programming commands to be executed by the hardware." EX-2001, ¶74; POPR at 28. But then ICPillar and its expert makes the creative and surprising argument that: "Rompaey outputs <u>only</u> 'a net list comprising the layout information of the implementation'…." EX-2001, ¶76. ICPillar and its expert just ignore the portions of the Rompaey that directly conflict with their creative

argument. Moreover, ICPillar apparently seeks to exclude the '899 patent's sole embodiment describing "ASIC Design 69" as a "layout … sent out to a manufacturer for fabrication" that includes the alleged software output (ASIC Programming 71).

Rompaey generates and outputs hardware and matching software drivers:

> [A]utomated synthesis of hardware/hardware and hardware/software interfaces, including the generation of the software drivers in programmable processors is possible. This is an essential part of hardware/software co-design.

EX-1004, 12:39-43; 4:60-63 ("most essential step is to generate the necessary software and hardware…"). CoWare's "outputs" include these "essential" drivers:

> OUTPUT: CoWare encapsulation containing the original process definition, the I/O device drivers, the hardware interfaces, the hardware and software model of the processor.

EX-1004, 27:61-64.

Rompaey describes a fully implemented pager. EX-1004, 21:34-26:49. In that pager, drivers are "merged with the original software application code." *Id.*, 26:38-49. Such a soup-to-nuts teaching of a functional pager designed by CoWare teaches the software output for the device – in greater detail than the '899 patent.

Rompaey's "essential" step generates tailored hardware with associated software. Rompaey generates a "heterogenous" output with "hardware subsystems" that match "software subsystems" executed on the hardware. EX-1004, 6:45-53, 9:16-25. Outputting only the hardware without the corresponding software renders

the device unusable because a future user of the hardware would not have information on the specific software (e.g., drivers) needed to operate that specific hardware. Rompaey's goal was using models including the hardware and software. EX-1004, 1:60-2:48 (describing the problem of separate hardware and software that Rompaey sought to solve); 6:6-18 (summarizing the solution which is to synthesize the hardware with the software together). ICPillar's position ignores Rompaey's entire goal to co-design the hardware and software for efficiency purposes.

Rompaey claims the "database" generated and output by CoWare. EX-1004, Claim 6 ("implementation is a <u>heterogeneous implementation comprising</u> hardware subsystems and software subsystems, said <u>software subsystems being executed on … said hardware subsystems</u>"). In Claim 8, the software is "machine instructions." *See also* EX-1004, claim 36 ("making an implementation of … digital system" with same implementation as claim 6). The '899 patent does not exclude CoWare's output of such "heterogenous implementation" including software from being within element 1[d]. Moreover claim 38 adds a "layout" of the implementation (that includes the executable software from claim 36) and thus describes what the sole embodiment of the '899 patent describes in Fig. 4 as detailed in Section II.

Also, Rompaey outputs a "software model" for the tailored hardware. EX-1004, 27:61-64. This "software model" includes the software "for compilation … into machine code" to be executed on the hardware. EX-1004, 18:25-30.

ICPillar wrongly cites Rompaey's simulation teachings. POPR at 29, 31, 34, 38-39. Rompaey has separate "simulation" and "implementation" teachings. *Compare* EX-1004, claim 2 ("simulation") with claim 6 ("heterogenous implementation comprising… software subsystems"); 6:36-44 (simulation); 6:45-53 ("implementation [with] machine instructions…"). Also, the '899 patent does not exclude that element 1[d] encompasses output that is also for simulation.

Similarly, ICPillar disparages that Rompaey "adds RAM and ROM blocks to store the program code" as "hardware." POPR at 46 (citing EX-1004, 20:22-25). But "***the*** program code" references ***the*** code in the preceding passages such as the "software I/O driver." EX-1004, 19:10-20:17. Even if RAM/ROM blocks are hardware, stating that they "store the program code" teaches outputting "the program code" to be stored. Also, "the program code" stored in a ROM block falls within the '899 Fig. 4 embodiment with software input to the layout. *See* Section II.

ICPillar's position requires ignoring: (1) Rompaey's express disclosure; (2) Rompaey's identified problem and solution; and (3) the '899 patent's disclosure.

IV. *Sotera* Stipulation

Petitioner stipulates "not to pursue in a parallel proceeding the same grounds or any grounds that could have reasonably been raised before the PTAB." EX-2010, at 3; EX-1020. Such stipulations at the POPR stage uniformly moot *Fintiv* issues. EX-2010, at 3 ("PTAB will not discretionarily deny institution).

Respectfully submitted,

DUANE MORRIS LLP

BY: /Kevin P. Anderson/
Kevin P. Anderson, Reg. No. 43,471
kpanderson@duanemorris.com
DUANE MORRIS LLP
901 New York Ave NW 700 E
Washington, D.C. 20001
T:202-776-5213
F:202-776-7801

Dated: June 10, 2024

CERTIFICATION OF SERVICE ON PATENT OWNER

Pursuant to 37 C.F.R. §§ 42.6(e), 42.8(b)(4) and 42.105, the undersigned certifies that on June 10, 2024, a complete and entire copy of this Reply to Patent Owner's Preliminary Response was served, as per agreement between the parties, via email and/or electronic download to ICPillar LLC by serving the following counsel for ICPillar, LLC:

Jaime W. Marquart (Cal. Bar No. 200344)

WAYMAKER LLP

515 S. Flower Street, Suite 3500

Los Angeles, California 90071

Telephone: (424) 652-7800

jmarquart@waymakerlaw.com

John Jeffrey Eichmann (Cal. Bar No. 227472)

EICHMANN, a professional corporation

662 N. Sepulveda Blvd., Suite 300

Los Angeles, California 90049

310-237-9190 (tel.)

jeichmann@eichmann.com

Nate Lowenstein

Kenneth Weatherwax

Colette Woo

Robert Pistone

Lowenstein & Weatherwax LLP

1016 Pico Boulevard

Santa Monica, California 90405

weatherwax@lowensteinweatherwax.com

lowenstein@lowensteinweatherwax.com

woo@lowensteinweatherwax.com

pistone@lowensteinweatherwax.com

/Kevin P. Anderson/

Kevin P. Anderson, Reg. No. 43,471
kpanderson@duanemorris.com
DUANE MORRIS LLP
901 New York Ave. NW 700 E
Washington, D.C. 20001
T:202-776-5213
F:202-776-7801

ATTORNEY FOR PETITIONER