UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,

Petitioner,

v.

SINGULAR COMPUTING LLC,

Patent Owner.

Patent No. 11,169,775
Filing Date: May 25, 2020
Issue Date: November 9, 2021

Inventor: Joseph Bates
Title: PROCESSING WITH COMPACT ARITHMETIC PROCESSING ELEMENT

PATENT OWNER'S RESPONSE

Case No. IPR2023-00397

TABLE OF CONTENTS

Page(s)

I. INTRODUCTION 1

II. THE '775 PATENT 3

III. THE ALLEGED PRIOR ART 10

A. Stuttard (PCT Publication WO2005/037326A2) 10

B. Shirazi ("Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines," Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, April 19-21, 1995) 13

IV. LEVEL OF ORDINARY SKILL IN THE ART 14

V. CLAIM CONSTRUCTION 14

VI. LEGAL STANDARD 15

VII. THE CHALLENGED CLAIMS ARE VALID 17

A. Ground 1A: Claims 1-5 Are Not Obvious Over Stuttard in Combination with Shirazi 17

1. A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi 17

(a) A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi to Yield a Reduced Bitwidth Poly Unit 17

(b) A POSA Would Not Have Modified Stuttard to Be a Custom Computing Machine 21

2. Petitioner's Combination Does Not Disclose or Render Obvious "an input-output unit" as Recited in Claim 1 25

B. Ground 1B: Claims 7-14 and 16-23 Are Not Obvious Over Stuttard in View of Shirazi 26

1. A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi 26

2. Petitioner's Combination Does Not Render Obvious "a processing element array comprising a plurality of first processing elements, wherein the plurality of first processing elements is no less than 5000 in number, wherein each of a first subset of the plurality of first processing elements is positioned at a first edge of the processing element array, and wherein each of a second subset of the plurality of first processing elements is positioned in the interior of the processing element array;" [7C] 26

(a) The 5000-LPHDR PE Limitation Carries Patentable Weight and Is Not Obvious 27

(b) A POSA Would Not Have Stacked Large Numbers (*e.g.*, 5000) of Low Precision High Dynamic Range PEs in any Computer Architecture, Including the One Described by Stuttard 30

(c) A POSA Would Not Have Made the Allegedly Obvious Implementation of the Scaled Stuttard-Shirazi Processor with 5000+ LPHDR PEs and a Plurality of 32-Bit Floating Point Multipliers 32

(d) A POSA Would Not Have Had a Reasonable Expectation of Success in Making the Proposed Stuttard-Shirazi Implementation with 5000+ PEs and a Plurality of 32-bit Floating Point Multipliers 37

3. Petitioner's Combination Does Not Render Obvious "an input-output unit connected to each of the first subset of the plurality of first processing elements;" 38

VIII. CONCLUSION 39

TABLE OF AUTHORITIES

Page(s)

Cases

Apple Inc. v. Samsung Elecs. Co.,
839 F.3d 1034 (Fed. Cir. 2016) 15

Chemours Co. FC, LLC v. Daikin Indus., Ltd.,
4 F.4th 1370 (Fed. Cir. 2021) 22

Forest Lab'ys, LLC v. Sigmapharm Lab'ys, LLC,
918 F.3d 928 (Fed. Cir. 2019) 15

Google LLC v. Singular Computing LLC,
IPR2021-00155, Paper No. 62 (P.T.A.B. May 11, 2022) 4

Graham v. John Deere Co. of Kansas City,
383 U.S. 1 (1966).....15

In re Kahn,
441 F.3d 977 (Fed. Cir. 2006) 16

KSR Int'l Co. v. Teleflex Inc.,
550 U.S. 398 (2007).....15, 16

In re Magnum Oil Tools Int'l, Ltd.,
829 F.3d 1364 (Fed. Cir. 2016) 16

In re NuVasive, Inc.,
842 F.3d 1376 (Fed. Cir. 2016) 16

TriVascular, Inc. v. Samuels,
812 F.3d 1056 (Fed. Cir. 2016) 16

Statutes

35 U.S.C. § 103 15

EXHIBIT LIST

Exhibit No.	Description of Document
2001	Definition of "bus" from Microsoft Computer Dictionary, Fifth Edition
2002	Declaration of Sunil P. Khatri, Ph.D.
2003	Curriculum Vitae of Sunil P Khatri, Ph.D.
2004	Deposition Transcript of Paul Franzon, Ph.D.

I. INTRODUCTION

The claims of the ’775 Patent are directed to computer systems that achieve massive parallelism, scalability by sacrificing execution/arithmetic unit-level precision while still achieving useful application-level results for a wide variety of applications. In particular, claim 7 of the ’775 Patent is directed to architecture comprising a processing element array of at least 5000 processing elements, each containing at least one arithmetic unit which operates on floating point values having low precision (i.e., no more than 11 mantissa bits) and high dynamic range (at least 6 exponent bits) (“LPHDR execution units”). By sacrificing precision, individual LPHDR execution units require fewer transistors and occupy less area on an integrated circuit (IC), which in turn allows a computer to have lots of them and specifically in claim 7, at least 5000 of them in a processing element array (PEA). This enables such systems to perform more computations per unit of resource (e.g., transistor, area, volume).

At the time of filing, the prevailing view in the prior art was that many applications require high levels of precision and can only be performed by full-precision floating point execution units. Dr. Joseph Bates, however, realized that the conventional wisdom was wrong: he understood that a computer with a processing array of many LPHDR execution units could provide just as good or better precision at the application level than the systems with full-precision execution units disclosed

in the prior art, even in applications that were thought to require full precision execution units. Ex 1001, '775 Patent, 20:44-51, 20:58-60.

Dr. Bates' counterintuitive insight is based on the fact that LPHDR execution units, by virtue of their lower transistor count, take up much less space than high-precision ones. Therefore, sacrificing full-precision capability (while maintaining high dynamic range) increases the number of arithmetic units that can fit on a single chip as Dr. Bates explains in the patent specification:

> Despite the common belief among those having ordinary skill in the art that modern applications require high precision processing, in fact a variety of useful algorithms function adequately at much lower precision. . . . [S]uch algorithms may be performed by processors or other devices implemented according to embodiments of the present invention, which come closer to achieving the goal of using a few transistors to multiply . . . thus enabling massively parallel arithmetic computation to be performed with relatively small amounts of physical resources.

'775 Patent, 7:13-23.

None of the references cited by Google disclose Dr. Bates' fundamental insight about the tradeoff between precision and parallelism. Instead, Petitioner's principal references disclose either processing arrays or low-precision arithmetic operators, but not processing arrays comprised of low-precision arithmetic operators. This is not surprising, since the conventional wisdom was that a large

number of low-precision units was not a useful approach to perform computations. The Stuttard reference discloses a SIMD unit comprised of an array of processing elements but does not disclose that its processing elements include low-precision arithmetic operators. In fact, it suggests otherwise. The Shirazi reference discloses low-precision arithmetic operators but does not disclose using these arithmetic operators in an array of processing elements. The remaining cited references similarly fail to disclose or suggest an array of low-precision processing elements. It is only Petitioner's expert who conjures up a low-precision processing array through the exercise of hindsight and by disregarding the evidence in the '775 Patent itself that POSAs believed that "massive amounts of arithmetic on a chip or in a massively parallel machine are not useful, and that ***massive amounts of LPHDR arithmetic are even worse***." '775 Patent, 7:1-4 (emphasis added). The '775 Patent erases this misconception. *Id.,* 17:29-23:51.

For these reasons and as discussed in more detail before, the Board should confirm the patentability of all the Challenged Claims.

II. THE '775 PATENT

The '775 Patent is entitled "Processing with compact arithmetic processing element" and issued on November 9, 2021. The '775 Patent claims priority, through a chain of parent and grandparent applications, to U.S. Provisional Patent Application No. 61/218,691, filed on Jun. 19, 2009. Claims from two of the patents

that have issued from this chain have been found valid by this Board in previous *inter-partes* reviews brought by the same Petitioner. *E.g.*, *Google LLC v. Singular Computing LLC*, IPR2021-00155, Paper No. 62 (P.T.A.B. May 11, 2022) (finding claims not unpatentable). As indicated above, the '775 Patent is directed to a computer processor that economically achieves a massive degree of parallelism through an array of multiplier circuits that each perform arithmetic on values having relatively high dynamic range exponents and low precision mantissas. *See* '775 Patent at 2:1-15; 5:66-6:19; 6:49-61; 7:3-9; 7:19-38; 8:38-42; 13:52-60; 16:41-55; 25:49-55.

The '775 Patent's inventor, Dr. Joseph Bates, recognized that even though then-modern processors contained hundreds of millions of transistors, they could perform only a handful of operations per clock cycle:

> Consider a modern silicon microprocessor chip containing about one billion transistors, clocked at roughly 1 GHz. On each cycle the chip delivers approximately one useful arithmetic operation to the software it is running. For instance, a value might be transferred between registers, another value might be incremented, perhaps a multiply is accomplished. This is not terribly different from what chips did 30 years ago, though the clock rates are perhaps a thousand times faster today.

Id., 1:40-48; Declaration of Sunil P. Khatri, Ph.D. (Ex. 2002) ("Khatri"), ¶ 38.

As Dr. Bates explained, a large portion of this inefficiency comes from using transistor-intensive arithmetic units that perform arithmetic on values having relatively high dynamic range exponents and high-precision mantissas:

> As described above, today's CPU chips make inefficient use of their transistors. For example, a conventional CPU chip containing a billion transistors might enable software to perform merely a few operations per clock cycle. Although this is highly inefficient, those having ordinary skill in the art design CPUs in this way for what are widely accepted to be valid reasons. For example, such designs satisfy the (often essential) requirement for software compatibility with earlier designs. Furthermore, they deliver great precision, performing exact arithmetic with integers typically 32 or 64 bits long and performing rather accurate and widely standardized arithmetic with 32 and 64 bit floating point numbers. Many applications need this kind of precision. As a result, conventional CPUs typically are designed to provide such precision, using on the order of a million transistors to implement the arithmetic operations.

'775 Patent, 2:62-3:10; Khatri, ¶ 39.

However, Dr. Bates realized that such high-precision computing was not necessary for all applications, including many valuable ones:

> There are many economically important applications, however, which are not especially sensitive to precision and that would greatly benefit, in the form of application performance per

> transistor, from the ability to draw upon a far greater fraction of the computing power inherent in those million transistors. Current architectures for general purpose computing fail to deliver this power.

'775 Patent, 3:11-17; Khatri, ¶ 40.

The '775 Patent thus is directed away from prior art computers, which are based on high precision execution units that take up space and are wasteful of transistors, and towards computers based on low precision high dynamic range (LPHDR) execution units, such as those based on LPHDR floating-point arithmetic:

> Embodiments of the present invention efficiently provide computing power using a fundamentally different approach than those described above. In particular, embodiments of the present invention are directed to computer processors or other devices which use low precision high dynamic range (LPHDR) processing elements to perform computations (such as arithmetic operations). One variety of LPHDR arithmetic represents values from one millionth up to one million with a precision of about 0.1%. If these values were represented and manipulated using the methods of floating point arithmetic, they would have binary mantissas of no more than 10 bits plus a sign bit and binary exponents of at least 5 bits plus a sign bit.

'775 Patent, 5:56-6:1; Khatri, ¶ 41.

By using an LPHDR architecture as claimed,

> the area of the arithmetic circuits remains relatively small and a greater number of computing elements can be fit into a given area of silicon. This means the machine can perform a greater number of operations per unit of time or per unit power, which gives it an advantage for those computations able to be expressed in the LPHDR framework.

'775 Patent, 6:11-17. Indeed, "[b]ecause LPHDR processing elements are relatively small, a single processor or other device may include a very large number of LPHDR processing elements, adapted to operate in parallel with each other." *Id.*, 6:49-52. As a result, "embodiments of the present invention may be implemented as any kind of machine which uses LPHDR arithmetic processing elements to provide computing using a small amount of resources (*e.g.*, transistors or volume) compared with traditional architectures." *Id.*, 8:1-5. In short, the '775 Patent achieves the economical delivery of more computing power by teaching a computer architecture having large numbers of processing elements that each process numerical values having low precision and high dynamic range (*i.e.*, shorter mantissa) formats. Khatri, ¶¶ 42-43.

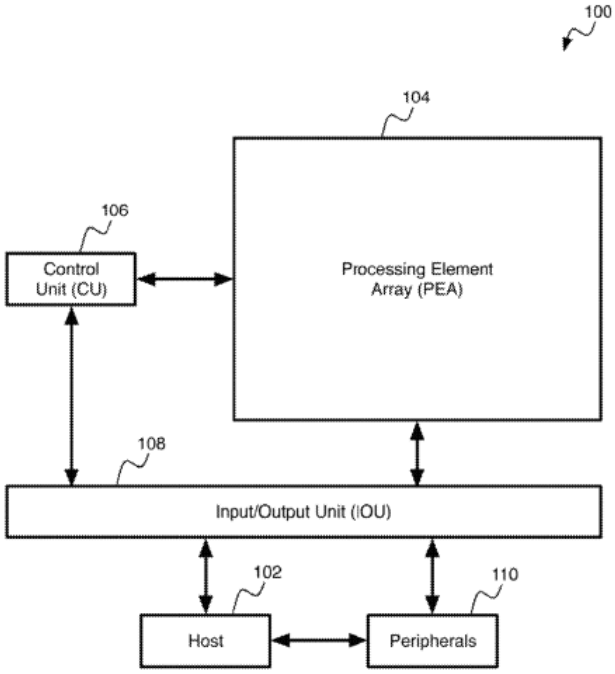
The '775 Patent observes that a POSA at the time of the invention believed that "LPHDR computation, and in particular massive amounts of LPHDR

computation, whether performed in a massively parallel way or not, is not practical as a substrate for moderately general computing." '775 Patent, 6:56-59.

> Despite these views—that massive amounts of arithmetic on a chip or in a massively parallel machine are not useful, and that massive amounts of LPHDR arithmetic are even worse—embodiments of the present invention disclosed a small amount of resources (e .g., transistors or volume) herein demonstrate that massively parallel LPHDR designs are in fact useful and provide significant practical benefits in at least several significant applications.

Id., 7:1-7; Khatri, ¶ 44.

In particular, the '775 Patent teaches a new computing architecture based on a massively parallel array of LPHDR processing elements (PEA). '775 Patent, 8:10-26. An embodiment is shown in Figure 1:



'775 Patent, FIG. 1; Khatri, ¶ 45.

In this embodiment, the "Host 102 is responsible for the overall control of the computing system 100." '775 Patent, 8:29-30. The Host seeks "to have the PEA 104 perform massive amounts of computations in a useful way . . . by causing the PE's to perform computations, typically on data stored in each PE, in parallel with one another." *Id.*, 8:37-40. To most efficiently control the PEA, the system may include a specialized control unit (CU) having "the primary task of retrieving and decoding instructions from an instruction memory . . . and issuing partially decoded instructions to all the PEs in the PEA 104." *Id.*, 8:49-52. The Host loads a program for execution by the PEA into the CU instruction memory and then instructs the CU

to interpret the program's instructions and cause the PEA to execute the instruction. *Id.*, 8:61-65; Khatri, ¶ 46.

The massively parallel PEA's ability to process data efficiently requires an I/O unit (IOU) 108 to interface the CU 106 and the PEA 104 with the Host along with the various peripherals in the Host system, with the PEA. '775 Patent, 9:7-15. Getting data into and out of the faster PEA without requiring large amounts of hardware resources is a critical design goal of the architecture covered by the Challenged Claims. *Id.*, 10:1-3. Data may be moved between PEs only to their "nearest neighbors" so that "there are no long distance transfers," as shown in Figures 2 and 3. *Id.*, 9:39-44. In this claimed embodiment of the invention, "data is read and written at the edges of the array and CU instructions are performed to shift data between the edges and interior of the PEA 104." *Id.*, 10:3-5; Khatri, ¶ 47.

III. THE ALLEGED PRIOR ART

A. Stuttard (PCT Publication WO2005/037326A2)

PCT Publication No. WO2005/037326A2 ("Stuttard") (Ex. 1006) was published on April 28, 2005.

Stuttard discloses a processor, called an MTAP processor, comprised of two execution units: a scalar (mono) execution unit and a SIMD (single-instruction multiple-data) parallel processing (poly) unit. Stuttard, 2:29-32. Instructions are fetched and decoded by a control unit which then issues those commands to the

appropriate part of the execution unit: either the scalar (mono) execution unit or the SIMD (poly) execution unit. *Id.*, 2:29-32, 9:18. Figure 7 is a diagram of an MTAP processor:

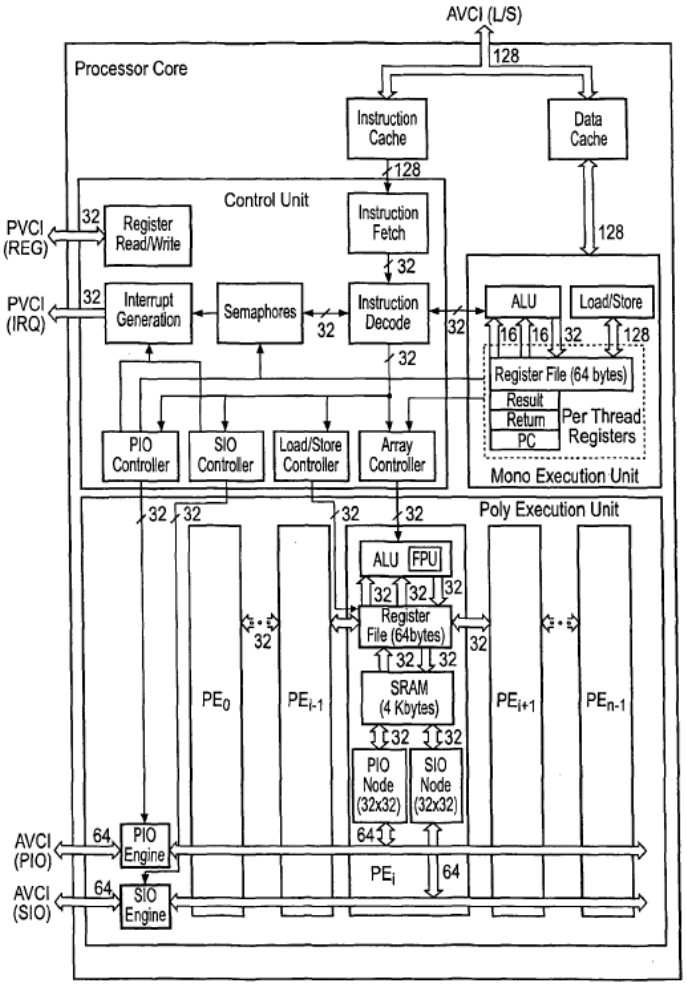


Fig. 7

Stuttard designed the MTAP processor to address two problems in existing parallel processing systems:

> Firstly, the array controller can only be programmed at a very low level: a series of microcode, or possibly assembler,

> instructions to be executed by the array. This is because the array controller is not designed to work with a compiler. Secondly, this programming is completely separated from the main application code running on the host processor.

Stuttard, 1:25-29; Khatri, ¶¶ 48-49.

To solve this problem, the MTAP processor is implemented so that "a single program runs on the processor which does both mono and poly operations as required - typically it will do both at the same time. A single C program can be written which manages program control flow, operates on mono data, and operates on poly data." Stuttard, 3:3-6. As explained by Stuttard:

> The design of this processor architecture was driven by the desire to make it practical and efficient to program in a high level language. In order to make this possible a number of features are required. First, the overall programming model needs to be simple and 'regular' - i.e. ***the same operations and addressing modes can be applied to all data types***.
>
> The execution unit for mono data is based on well known principles used in many RISC processors. This makes it simple for a compiler to target and easy for a user to understand.
>
> ***The functionality supported by each PE is then made as similar as possible to the mono execution unit***: each PE has an ALU, a register file and memory, and supports a range of addressing modes for transferring data between memory and registers.

Id., 7:6-16 (emphasis added); Khatri, ¶ 50.

> The MTAP processor contains two I/O mechanisms:
> The first, *Programmed l/O* (PIO), is the normal mechanism used for accessing memory external to the MTAP core: it supports random accesses to variable sized data by the PEs. The second mechanism is *Streaming l/O* (SIO) which allows chunks of contiguous data to be streamed directly into the memory within PEs.

Stuttard, 10:21-25. Each of these I/O systems consist of three parts: "Controller, Engine, and Node." *Id.*, 19:23. "The PIO and SIO controllers decode I/O instructions . . . ." *Id.*, 19:25. The I/O system includes control lines from the PIO and SIO controllers to their respective I/O engines to allow the controllers to control the I/O engines. Ex. 2004, Franzon Dep., 135:10-18. The I/O engines activate the Nodes in each PE to allow for serial data transfers to the PEs. Stuttard, 20:2-3; Khatri, ¶ 51.

B. Shirazi ("Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines," Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, April 19-21, 1995)

Shirazi, *et al.*, "Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines," Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines ("Shirazi") was allegedly published in 1995 (Ex. 1036).

Shirazi proposes specialized floating-point operators appropriate for use in custom computing machines (CCMs) tailored for a single application. Shirazi, 155.

Shirazi discloses floating point units (FPUs) designed for two specific applications: a 2-D fast Fourier transform application and a FIR filter application. *Id.*, 155-56. The floating-point operators for these two specific applications are tailored for a specific computing platform: a specific field programmable gate array (FPGA) chip. Shirazi adopted an 18-bit floating point format for the 2-D FFT application and a 16-bit floating point format for the FIR filter application. *Id.*, 156. Shirazi proposed specific designs for eighteen and sixteen-bit floating point adders/subtracters, multipliers, and dividers implemented in FPGAs. *Id.*, 155-162; Khatri, ¶ 52.

IV. LEVEL OF ORDINARY SKILL IN THE ART

Petitioner challenges claims 1-5, 7-14, and 16-23 (the "Challenged Claims"). Pet., 1.

A person of ordinary skill in the art (POSA) would be a person with a Bachelor's degree in Computer Science, Electrical Engineering, or Applied Mathematics, with 2 years of academic or industry experience in computer architecture. Khatri, ¶¶ 35-36. This is the same as Petitioner's proposed level of skill, except that Petitioner's level included any number of years of experience more than two. Pet., 4.

V. CLAIM CONSTRUCTION

Patent Owner agrees that claim construction is unnecessary.

VI. LEGAL STANDARD

The question of obviousness is resolved on the basis of underlying factual determinations, including: (1) the scope and content of the prior art, (2) any differences between the claimed subject matter and the prior art, (3) the level of skill in the art, and (4) where in evidence, so called secondary considerations. *Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17–18 (1966). A claim is only unpatentable under 35 U.S.C. § 103 if "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007) (quoting 35 U.S.C. § 103).

"An invention is not obvious simply because all of the claimed limitations were known in the prior art at the time of the invention. Instead, we ask 'whether there is a reason, suggestion, or motivation in the prior art that would lead one of ordinary skill in the art to combine the references, and that would also suggest a reasonable likelihood of success.'" *Forest Lab'ys, LLC v. Sigmapharm Lab'ys, LLC*, 918 F.3d 928, 934 (Fed. Cir. 2019). "Of course, concluding that the references are within the scope and content of the prior art to be considered for obviousness does not end the inquiry. *Graham* makes clear that the obviousness inquiry requires a determination whether the claimed invention would have been obvious to a skilled

artisan." *Apple Inc. v. Samsung Elecs. Co.*, 839 F.3d 1034, 1050 n.14 (Fed. Cir. 2016).

"Although the *KSR* test is flexible, the Board 'must still be careful not to allow hindsight reconstruction of references . . . without any explanation as to how or why the references would be combined to produce the claimed invention.'" *TriVascular, Inc. v. Samuels*, 812 F.3d 1056, 1066 (Fed. Cir. 2016) (citation omitted). Further, an assertion of obviousness "cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *KSR*, 550 U.S. at 418 (quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)); *accord In re NuVasive, Inc.*, 842 F.3d 1376, 1383 (Fed. Cir. 2016) (stating that "conclusory statements" amount to an "insufficient articulation[] of motivation to combine"; "instead, the finding must be supported by a 'reasoned explanation'" (citation omitted)); *In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) ("To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.").

VII. THE CHALLENGED CLAIMS ARE VALID

A. Ground 1A: Claims 1-5 Are Not Obvious Over Stuttard in Combination with Shirazi

1. A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi

Petitioner's obviousness arguments are an exercise in hindsight that fail to rebut the inventor's observation that a POSA would not believe that LPHDR computations in an array of processors are useful. Petitioner's motivations to combine lack factual underpinnings and are inconsistent with the inventive concept underlying Stuttard. Khatri, ¶ 54.

(a) A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi to Yield a Reduced Bitwidth Poly Unit

Petitioner's obviousness arguments rely on modifying the processing elements (PEs) in Stuttard's processor array (poly unit) to use the reduced-bitwidth floating-point formats of Shirazi, while citing no evidence that anyone had ever made or even suggested making an array of processors that perform low-bitwidth multiplication as required by the claims. In fact, before the '775 Patent, it was thought that deploying a large number of LPHDR multipliers was a poor design choice. '775 Patent, 6:55-59, 7:1-7. The Petition relies on references disclosing *either* reduced-bitwidth floating point computations *or* processing arrays to support its argument that a POSA would have been motivated to construct a processing array

using reduced-bitwidth floating point values. Pet., 11-14. None of the references cited for the proposition that reduced-bitwidth operators are useful or desirable *also* teach or suggest implementing those operators in a processing array, or vice versa. Khatri, ¶ 55.

The Petition begins by attempting to establish that a POSA knew that low-bitwidth operators are smaller than full-bitwidth operators. Petitioner first cites to Shirazi's teaching that its reduced-bitwidth operators use less chip area as compared to full-precision circuits. Pet., 11. Importantly, Shirazi does not teach that these chip-area savings should be used to implement multiple of his operators in parallel. Khatri, ¶ 56.

The Petition next cites to Tong (Ex. 1034) for further evidence that processors with a reduced number of mantissa bits has "small size." Pet., 11 (citing Tong, 282). Regardless, Tong does not suggest that more than one of these "small size" processors should be used simultaneously to obtain more processing power per unit area as taught by the '775 Patent. In fact, Tong teaches that a reduced-bitwidth processor should be paired with a full-precision floating point unit and a sleep-mode circuit to shut down the unused unit, which would require an even greater area than a full-precision floating point unit. Tong, 282. Thus, Tong not only fails to teach using more than one reduced-bitwidth processor, such as in a processing array, it expressly teaches using only one processor at a time. Khatri, ¶ 57.

The Petition then cites to Ligon (Ex. 1105), Roma (Ex. 1125), Sudha (Ex. 1098) and Aty (1092) to corroborate that reduced-precision floating point units have small size, but, as in the other cited references, none teach taking advantage of that size savings by implementing them in a massively parallel processing array. Khatri, ¶ 58.

The Petition comes to the unsubstantiated conclusion that Shirazi and the other references teach that "custom floating-point formats with lower mantissa bitwidths than the IEEE's full-precision format provided a desirable solution for implementing smaller-sized arithmetic units for various digital-signal and image processing applications, which reduced power consumption and ***allowed for more PEs on a single chip to achieve higher parallelism and speed***." Pet., 12 (citing Ex. 1003, ¶¶ 84-90) (emphasis added). The references teach no such thing. Nowhere do these references suggest putting more PEs on a single chip for achieving higher parallelism or to attain any other benefit. Khatri, ¶ 59.

Petitioner then shifts to a discussion of Stuttard which seeks to establish that a POSA would use Shirazi's reduced bitwidth floating-point formats in Stuttard's PEs. Pet., 12-13. Critically, the Petition presents no evidence of any processing array having been implemented with reduced bitwidth floating point formats. Stuttard contains no such embodiment or teaching, nor do the other references which the

Petition contends corroborate that FFT calculations can be done in a multi-processor environment. Pet., 13 (citing Ex. 1080, Ex. 1107, Ex. 1108). Khatri, ¶ 60.

Petitioner argues that a POSA would have been motivated to use Shirazi's reduced bitwidth floating-point formats in Stuttard's PEs to shrink the size of the PEs so that more PEs would fit on the chip, Pet., 13, but Petitioner submits no evidence that using reduced-bitwidth floating-point formats alone would have resulted in size savings that would translate into more PEs in the Stuttard system. In fact, the evidence suggests otherwise. Each PE in Stuttard includes an 8-bit ALU (which includes the FPU), meaning that 8 bits of data are loaded into the ALU in each clock cycle. Stuttard, 17:15, Fig. 7 (showing FPU in the ALU). To accommodate arithmetic on values larger than 8 bits, "instructions exist for multi-byte arithmetic which is handled by iteration." *Id.*, 17:27-28. The reduced-precision formats taught by Shirazi, 16 and 18 bits, are also larger than 8 bits, which suggests that arithmetic on those values would also be done by iteration in Stuttard without altering the ALUs. As such, Petitioner has presented no evidence showing that the 8-bit ALU's would be made smaller in the proposed combination, thereby allowing Stuttard to include more PEs in its poly unit. Petitioner's motivation to combine is therefore unsupported. Khatri, ¶ 61.

In addition, the Petition fails to rebut the evidence in the '775 Patent specification that POSAs did not believe that parallel processing using low-precision

arithmetic was useful. ’775 Patent, 6:55-59, 7:1-7. The fact that Petitioner submitted approximately 100 references with its Petition but not a single one that discloses a parallel processing array with reduced-bitwidth floating-point processing elements further corroborates that POSAs did not view the architecture taught by the ’775 Patent as viable before it was disclosed by Dr. Bates. Khatri, ¶ 62.

(b) A POSA Would Not Have Modified Stuttard to Be a Custom Computing Machine

Petitioner’s argument assumes, without evidence, that a POSA would have found it obvious to modify Stuttard from a general purpose architecture into the type of custom computing machine that Shirazi envisions for his custom reduced-bitwidth operators. To the contrary, a POSA would not have been motivated to modify Stuttard from a general-purpose architecture suitable for many applications into a customer computing machine capable of performing only low precision operations suitable for 2D FFT or FIR filter computations. Khatri, ¶ 63.

The inventive concept behind Stuttard’s architecture and software solution was to “provide a fully integrated architecture where ***the same processor can transparently process both scalar and SIMD operations in a fully integrated fashion***. This means that a single program can be written, in a standard high-level language such as C, which includes all parts of the application – ***both ‘normal’ (scalar) data*** and parallel data processing.” Stuttard, 2:4-8 (emphasis added).

Stuttard envisioned that his architecture would execute a full instruction set which would be common to the poly and mono execution units, *id.*, 2:29-3:14, and would be broadly applicable to many types of "data flow applications," *id.*, 8:29-30, including network processing and bio-informatics (such as molecular simulations of individual molecules and the interactions of molecules, genetic sequence comparisons, and pharmacophore fingerprinting). *Id.*, 23:17-24:21; Khatri, ¶ 64.

Shirazi, on the other hand, proposes creating custom computing machines by implementing custom operators that are appropriate only for one of two narrow applications: either a 2-D FFT or FIR filter application running on a specific platform (the Xilinx 4010 FPGA). Shirazi, 155. Shirazi does not teach or suggest that the custom operators or the reduced-bitwidth formats that they use are appropriate for general applications and/or general platforms. Khatri, ¶ 65.

A POSA would not have been motivated to strip out Stuttard's inventive concept by modifying the poly unit to be capable only of performing a narrow set of computations using reduced-bitwidth operators (such as Shirazi's 2-D FFT and FIR filter operators). *See Chemours Co. FC, LLC v. Daikin Indus., Ltd.*, 4 F.4th 1370, 1377 (Fed. Cir. 2021) (finding insufficient motivation to modify a prior art reference "when doing so would necessarily involve altering the inventive concept of" the reference). While Stuttard teaches that the MTAP architecture can easily be customized for the "target application," Stuttard, 9:13-14, nothing in Stuttard

suggests that a "target application" is so narrow as to create a custom computing machine which performs only a narrow set of computations. Rather, Stuttard's goal was to create an architecture that is applicable to the broad class of "data flow applications," *id.*, 8:29-30, where "the same processor can transparently process both scalar and SIMD operations in a fully integrated fashion" using a single, high-level program. Stuttard, 2:4-9. Stuttard's single program for dispatching a full set of commands to both the mono and poly units would be superfluous if the poly unit executes only one type of instruction as in Shirazi, since Stuttard teaches "[t]he processing elements in the SIMD array and the mono execution unit have a common instruction set." Stuttard, 3:8-9. In this sense, the combining of Shirazi into Stuttard would negate and diminish the inventiveness and flexibility of Stuttard. Khatri, ¶ 66.

The Petition presents no evidence addressing this deficiency in its motivation-to-combine analysis, but the Petition does base its contention that a POSA would have reasonably expected success in the combination because Shirazi's "custom formats . . . are feasible on CCMs" and "Stuttard's MTAP architecture is such a custom computing machine." Pet., 14. According to the Petition, Stuttard discloses a custom computing machine because the PEs use 8-bit ALUs and the mono unit has a 32-bit ALU. However, the fact that the PEs' ALUs accept 8 bits of data per cycle as opposed to the mono unit's 32 bits per cycle is wholly irrelevant to whether the Stuttard is a "custom computing machine." As discussed above, to accommodate

arithmetic in the PEs' ALUs on values larger than 8 bits, "instructions exist for multi-byte arithmetic which is handled by iteration," Stuttard, 17:27-28, allowing both the poly and mono units to have the "same instruction set." *Id.*, Abstract, 3:8-9, 3:26-27, 4:23-31, 5:5-6, Claims 2, 3, 4. The poly unit's PEs are therefore designed to operate on the same types of data as the mono unit:

> [T]he overall programming model needs to be simple and 'regular' – i.e., the same operations and addressing modes can be applied to all data types. . . . ***The functionality supported by each PE is then made as similar as possible to the mono execution unit***: each PE has an ALU, a register file and memory, and supports a range of addressing modes for transferring data between memory and registers.

Stuttard. 7:8-16 (emphasis added). *See also id.*, 3:8-9. As such, the already-small size of ALUs in the poly PEs (8-bit) does not suggest that a POSA would be motivated to diminish the functionality of the poly unit by instead using Shirazi's narrow-capability PEs. Further, replacing these 8-bit Stuttard ALUs by Shirazi's units (18-bit for 2-D FFT and 16-bit for FIR filters) would potentially *increase* their size. So, the combination of Stuttard and Shirazi would be a) contrary to the goals of Stuttard (general-purpose applications, 8:29-30, 23:17-24:21, and "unity of the architecture," 3:7-9) and b) possibly result in larger PEs, thus cutting against the motivation to combine proposed by Petitioner. Khatri, ¶ 67.

2. Petitioner's Combination Does Not Disclose or Render Obvious "an input-output unit" as Recited in Claim 1

Petitioner fails to identify the claimed "input-output unit" because "the bus plus the I/O engines and channels" it does identify excludes the key components of Stuttard's I/O system: the I/O controllers (the PIO and SIO controllers). Pet. at 30; Khatri, ¶ 68.

Stuttard discloses that "the I/O system consist of three parts: Controller, Engine and Node." Stuttard, 19:23. "The PIO and SIO controllers decode I/O instructions and coordinate with the rest of the control unit and the mono processor." *Id.*, 19:25-26. The I/O system includes control lines from the PIO and SIO controllers to their respective I/O engines to allow the controllers to control the I/O engines. Franzon, 135:10-18; Stuttard, Fig. 7. The I/O engines activate the Nodes in each PE to allow for serial data transfers to the PEs. Stuttard, 20:1-2; Khatri, ¶ 69.

Petitioner's identification of the claimed input-output unit omits the I/O controllers, the crucial components of Stuttard's I/O system. The "bus plus the I/O engines and channels" are inoperable without I/O controllers to issue control signals to the I/O engines to manage data transfers. Petitioner provides no explanation how Stuttard's "bus plus the I/O engines and channels" is the claimed input-output unit if it does not include the components Stuttard identifies as necessary to drive the I/O engines and the input-output function. Khatri, ¶ 70.

Accordingly, the Petition fails to show that the input-output unit is rendered obvious.

B. Ground 1B: Claims 7-14 and 16-23 Are Not Obvious Over Stuttard in View of Shirazi

1. A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi

Petitioner's obviousness arguments depend entirely on its Stuttard-Shirazi combination discussed above. Pet. at 51-59. For the same reasons set forth with respect to Ground 1A, Petitioner has not shown that a POSA would be motivated to modify Stuttard to adopt Shirazi's bit-width floating point formats. Khatri, ¶ 72.

2. Petitioner's Combination Does Not Render Obvious "a processing element array comprising a plurality of first processing elements, wherein the plurality of first processing elements is no less than 5000 in number, wherein each of a first subset of the plurality of first processing elements is positioned at a first edge of the processing element array, and wherein each of a second subset of the plurality of first processing elements is positioned in the interior of the processing element array;" [7C]

The Board properly "question[ed] the strength of Petitioner's case" that it was obvious to scale the Stuttard-Shirazi to over 5000 reduced bitwidth PEs in one computing chip in the Decision Granting Institution of *Inter Partes* Review (DI). DI, 52. Petitioner relied on the aspirational statement in Stuttard that the "poly execution unit may consist of tens, hundreds ***or even thousands*** of PE's," Stuttard, 10:12 (emphasis added), as a motivation to create a poly unit specifically with at least 5000

PEs. Pet., 51. Petitioner, however, presented no evidence why a POSA would have chosen to include a minimum of 5000 PEs in the array. Khatri, ¶ 73.

The Board also properly found that the Petition lacked any meaningful discussion of "a motivation to scale up to at least 5,000 *low precision* PEs or of a reasonable expectation of success of scaling up to 5,000 low precision PEs." DI, 52-53. None of the evidence in the Petition addresses supports that a POSA would have been motivated to make a massively parallel processor of low-precision PEs, especially in view of the '775 Patent's disclosure that POSAs believed that "massive amounts of LPHDR arithmetic" are not useful. '775 Patent, 6:55-59, 7:1-7; Khatri, ¶ 74.

As discussed below, Claims 7-14 and 16-23 are not obvious over Stuttard and Shirazi.

(a) The 5000-LPHDR PE Limitation Carries Patentable Weight and Is Not Obvious

Lacking evidence that an array of 5000 or more low-precisions PEs was obvious, Petitioner asserts that the limitation carries no patentable weight because an array of that size provides no new and unexpected results. Pet. at 53-55. Petitioner offers no affirmative evidence in support but instead argues that the '775 Patent specification does not disclose that any particular number of PE's provides any particular benefit. Pet. at 53-54 (citing Ex. 1003, ¶ 170). Petitioner's argument

ignores columns of disclosure in the '775 Patent that spells out the new and unexpected result arising from the patent's underlying main discovery, namely that large amounts of LPHDR arithmetic performed by multiple LPHDR processing elements in a massively parallel architecture are, in fact, useful and provide significant practical benefits. '775 Patent, 7:1-36 (noting the prevailing view "that massive amounts of LPHDR arithmetic are" not useful), *id.,* 17:29-23:51 (noting the "surprising" result of performing many low-precision calculations is a result with less error than each individual calculation for several applications), *id.,* 20:46-48 (observing that "[t]o perform many calculations sequentially with 1% error and yet produce a final result with less than 1% error may seem counter-intuitive), *id.,* 20:58-60 (noting that '[t]he surprising results, below, show that even with this extreme level of . . . LPHDR arithmetic, useful results can be achieved"). The specification then goes on to provide various examples of the number of such LPHDR PEs that could be deployed on a chip including 5000. *Id.*, 28:13-29:4. In view of the unrebutted intrinsic evidence that an array of 5000 low-precision PEs is inventive, the Board should consider this limitation, like all limitations, as part of the patented invention. Khatri, ¶ 75.

Petitioner also argues that the claimed number of the claimed PEs is a "result-effective variable," but the Petition merely cites art regarding the number of *full-precision* PEs, not the number of LPHDR PEs as claimed. Pet. at 54-55. As the '775

Patent itself notes, at the time of the invention, a POSA would not expect large amounts of LPHDR arithmetic performed by multiple LPHDR processing elements in a massively parallel architecture to provide useful results, *id.*, 7:1-36, 6:55-59, and in fact the POSA would expect that this would cause results that were simply too imprecise. *Id.*, 17:29-23:51 (noting the "surprising" result of performing many low-precision calculations is a final result with less error than each individual calculation), *id.,* 20:44-51, 20:58-60. Such an unexpected counter-intuitive result arising from increasing the number of LPHDR PEs belies the notion that the number of claimed PEs is a result-effective variable arising from routine experimentation. Khatri, ¶ 76.

Only one reference cited by Petitioner in support of its argument that the claimed number of LPHDR PEs is somehow result-effective, Ex. 1014, mentions lower-precision. That reference addresses the narrow problem space of physics-based animation (PBA) simulations based on human perceptual error tolerance. Ex, 1014, 1. The reference makes specific note that "precision reduction tolerance may vary across different physics engines" and does not suggest including anywhere near 5000 PEs. Ex. 1014, 9; Khatri, ¶ 77.

Accordingly, Petitioner has not shown that this limitation is obvious.

(b) A POSA Would Not Have Stacked Large Numbers (*e.g.*, 5000) of Low Precision High Dynamic Range PEs in any Computer Architecture, Including the One Described by Stuttard

As the Board recognized in the Decision, Petitioner presented no evidence supporting "a motivation to scale up to at least 5,000 *low precision* PEs." DI, 52-53. Petitioner failed to do so because no such evidence exists. Khatri, ¶ 79.

A central advance of the '775 Patent is the discovery that large amounts of LPHDR arithmetic performed by multiple low precision processing elements in a massively parallel architecture are, in fact, useful and provide significant practical benefits which were unexpected and counter-intuitive. '775 Patent, 7:1-36, 20:44-51, 20:58-60. Petitioner has offered no evidence that ***anyone*** had reached this discovery prior to Dr. Bates. Indeed, Shirazi and Stuttard, though published 14 and 6 years before the '775 Patent, respectively, contain no hint that large numbers of low precision and high dynamic range calculations could provide useful results for ***any application***. This discovery, and the related architecture covered by the Challenged Claims, were non-obvious for so long in large part because it was surprising and counter-intuitive that performing so many individual arithmetic operations at low precision can lead to a final result with less error than each such individual arithmetic operation. '775 Patent, 17:29-23:51 (noting the "surprising" result of performing many low-precision calculations is a result with less error than

each individual calculation), *id.,* 20:46-48 ("[t]o perform many calculations sequentially with 1% error and yet produce a final result with less than 1% error may seem counter-intuitive."); *id.* 20:58-60 (noting that "[t]he surprising results, below, show that even with this extreme level of . . . LPHDR arithmetic, useful results can be achieved"). In this IPR, and indeed all of its other IPRs, Petitioner simply assumes that a POSA would be aware of this discovery despite providing ***no evidence*** whatsoever to this end. Khatri, ¶ 80.

Petitioner points to the teachings of Shirazi and other references it has cited in its past IPRs (*i.e.*, Tong, Ex 1034) in an attempt to show that the discovery of the '775 Patent was obvious to a POSA. As discussed in detail above, those references say nothing about the number of processing elements performing imprecise LPHDR arithmetic in parallel, with all processing elements repeatedly performing imprecise LPHDR arithmetic. Indeed, by failing to proffer any contrary evidence, Petitioner tacitly admits that ***no one*** before Dr. Bates conceived of a computer architecture comprising large numbers of LPHDR processing elements in a massively parallel configuration. At a minimum, it was incumbent on Petitioner to provide evidence *why* a POSA would have found it obvious to make this combination despite the '775 Patent's observation that such a combination would have been rejected by a POSA as useless. '775 Patent, 6:55-59, 7:1-7. The failure to offer such evidence is fatal to

the Petition because the only evidence of record is the '775 Patent's teaching that such an architecture was not obvious to a POSA. Khatri, ¶ 81.

To the extent the Petition presents evidence of scaling PE's, none of it suggests scaling up to 5000 or more LPHDR PE's as the Board observed. DI, 52-53. The Petition cites references that purport to address the advantages of scaling PEs, Pet., 51-53, but the references do not address scaling LPHDR PE's and the problems specific to that implementation. The Petition simply assumes, in a hindsight-driven exercise, the existence of the Stuttard-Shirazi combination and applies teachings about scaling conventional PEs. *Id.* Given the error endemic in LPHDR PEs and the '775 Patent's teaching that massively parallel arrays of LPHDR PEs were not considered useful, the mere teaching that conventional PEs can be scaled does not render obvious scaling at least 5000 LPHDR PEs. Khatri, ¶ 82.

(c) A POSA Would Not Have Made the Allegedly Obvious Implementation of the Scaled Stuttard-Shirazi Processor with 5000+ LPHDR PEs and a Plurality of 32-Bit Floating Point Multipliers

Petitioner identifies an allegedly obvious implementation of a scaled Stuttard-Shirazi combination to meet the claims. Lacking any motivation to construct or scale an array of 5000+ LPHDR PEs with a plurality of 32-bit floating point multipliers, the proposed implementation applies hindsight that ignores the POSA's perceptions about the drawbacks to this implementation. Khatri, ¶ 83.

Petitioner's obviousness implementation includes enough mono units and poly units collectively including at least 5000 of Shirazi's PEs because "Stuttard teaches scaling a single MTAP processor core's poly unit to "thousands of PEs." Pet., 59. Even if process technology permitted over 5000 PEs to be included on one chip, Petitioner does not show that a POSA would have been motivated to scale up to 5000 PEs under the I/O structure disclosed in the Stuttard-Shirazi combination. Khatri, ¶ 84.

The '775 Patent teaches that the burden on an I/O unit can be reduced by including data connections between PE's in all four directions: north, south, east, and west. '755 Patent, 9:55-60, FIGS. 2, 3. The Stuttard-Shirazi combination includes "swazzle" connections between the PE's only to the east and west, Stuttard, 20:31-34, which would result in inefficient data movement through the PEs. This would result in increased delays, synchronization issues, higher power consumption, and increased circuit area that would discourage a POSA from scaling the Stuttard-Shirazi combination to 5000 PEs in the poly unit. The increased delays arise from the fact that Stuttard's array is 1D in nature, so traversing 5000 PEs in a linear manner would be delay-intensive. Synchronization issues would result from global signals that need to be transmitted to all PEs in Stuttard's 1D array. A higher power consumption and increased circuit area would result from the fact that the global signals would require larger driving circuits, which consume more power. The

drawbacks of Petitioner's proposed combination and the experimentation required to overcome them would make the combination unappealing to and unattainable by the POSA. Further, if Petitioner suggests using a 2D array of Stuttard's PEs, the generalization of Stuttard's 1D "swazzle" to 2D is not trivial and would be beyond the skill of a POSA, fundamentally changing the hardware design, as well as the software programming model, as well as the compilers and software debugging infrastructure. Khatri, ¶ 85.

Petitioner points to Exs. 1108 and 1081 for evidence that using swazzle connections in two dimensions was a "conventional alternative to a linear topology." Pet., 13, 17-18, 51. Neither reference supports this argument. Ex. 1108, authored by Stuttard, does not mention a 2D arrangement at all. Rather, it states that "other (than linear) interconnect topologies could be introduced." Ex. 1108, 35. Nowhere does the reference refer to 2D topologies. Indeed, Petitioner does not explain why "other than linear" must be a 2D PEA; it could as easily be a ring, star, or other topology. *See* Pet. at 17 (citing Ex. 1108, 35). Finally, all these "other than linear" topologies would entail not just a significant change in the hardware design (which are beyond the skill of a POSA) but would also require a different programming model, new compilers, and new debugging environments, thus changing the way a user would utilize the Petitioner's combination. Khatri, ¶ 86.

Ex. 1081 ("Harasimiuk") does not support Petitioner's argument that swazzle connections in two dimensions were conventional. Describing an array of PEs, Harasimiuk states, "PE interconnections may allow data to be moved along the PE array (hereinafter swazzled). The PE interconnections may result in a 1D ring/line array or a 2D grid array. ***Preferably, a 1D line array is provided***." Ex. 1081, 5:62-65 (emphasis added). Harasimiuk's implementation "assumes that the PEs are connected in a linear array with nearest-neighbour ***bi-directional*** communication capacity." *Id.*, 21:7-9 (emphasis added). If a 2D PEA is used, Harasimiuk teaches that to use his algorithm, one should "treat the mesh as a set of ***linearised and independent arrays***, ignoring the Y connection, and to partition the image recursively among the many ***independent 1D sub-arrays***." *Id.*, 21:14-17 (emphasis added). Harasimiuk states that another common configuration is a "2D array with a two dimensional bi-directional (up, down, left, right) nearest-neighbour configuration," *id.*, 21:11-12, but Harasimiuk proposes no embodiment that successfully implemented such an array. Harasimiuk does not suggest a 2D 5000 LPHDR PEA, and even if it did, this would require a different programming model, new compilers, and new debugging environments. Khatri, ¶ 87.

Petitioner proposes to "replicate an existing processor core design (with its corresponding ratio of poly to mono execution units) multiple times on the chip." Pet., 56. The result would be a single processor with several mono units as well as

5000+ PEs in several poly units. This combination would not be obvious to a POSA because Stuttard's programming solution presupposes a system with a single mono unit and one poly unit. The challenge of programming a processor comprised of multiple mono units and multiple poly units would result in a completely different and significantly more complex parallel computing programming model, which would require new software and hardware implementation which is beyond the skill of a POSA. In particular, this would result in a machine in which instructions would have to be scheduled on the multiple mono and multiple SIMD poly PEs. The resulting hardware would entail a significantly different architecture and implementation. Further, the multiple mono processors would need to be programmed using a MIMD (multiple instruction, multiple data) paradigm, fundamentally changing the programming model of Stuttard's disclosure to a significantly more complex one. With multiple mono and multiple SIMD poly PEs, the system would change into a hybrid MIMD-SIMD architecture. Not only would this require significant hardware experimentation and exploration which would be beyond the skill of a POSA, but also, it would require a fundamental change in the programming model, compiler software and debugger software, a task that a POSA would not be able to undertake. Thus, the overall enterprise would be beyond the skill of a POSA and would have discouraged a POSA from attempting to scale PEs

to 5000 and providing for a plurality of 32-bit floating point multipliers by replicating the processor core as the Petition contends. Khatri, ¶ 88.

For these reasons, Petitioner's proposed implementation of Stuttard-Shirazi to meet the 5000-PE limitation as well as the limitation of a plurality of 32-bit floating point multipliers would not have been obvious.

(d) A POSA Would Not Have Had a Reasonable Expectation of Success in Making the Proposed Stuttard-Shirazi Implementation with 5000+ PEs and a Plurality of 32-bit Floating Point Multipliers

For the reasons set forth in the preceding section which would have discouraged a POSA from making the proposed implementation, a POSA would not have reasonably expected to succeed in making that implementation. Khatri, ¶ 90.

Petitioner's discussion on likelihood of success is limited to discussing whether existing process technology would have permitted placing 5000 PEs and multiple mono units on a single chip. Pet., 60-65. Petitioner ignores and presents no evidence regarding the technical challenges associated with the hardware and software redesign needed for a massively parallel system with multiple mono and poly units. In particular, this would result in a machine in which instructions would have to be scheduled on the multiple mono and multiple SIMD poly PEs. The resulting hardware would entail a significantly different architecture and implementation which would be beyond the skill of a POSA. Khatri, ¶ 91.

Further, the multiple mono processors would need to be programmed using a MIMD (multiple instruction, multiple data) paradigm, fundamentally changing the programming model of Stuttard's disclosure. With multiple mono and multiple SIMD poly PEs, the system would change into a hybrid MIMD-SIMD architecture. Not only would this require significant hardware experimentation and exploration which would be impossible for a POSA to conduct, but also, it would require a fundamental change in the programming model, compiler software and debugger software, a task that a POSA would not be able to undertake. A POSA with a bachelor's degree and two years' experience in computer architecture would be unlikely to succeed in overcoming these challenges. Khatri, ¶ 92.

3. Petitioner's Combination Does Not Render Obvious "an input-output unit connected to each of the first subset of the plurality of first processing elements;"

Petitioner relies on the same arguments as with Claim 1's input-output unit limitation. Pet. at 67. For the same reasons as discussed above in Section VII.A.2, therefore, Petitioner's Ground 1B combination does not render this limitation obvious. Khatri, ¶ 93.

VIII. CONCLUSION

For the foregoing reasons, Patent Owner respectfully requests that the Board issue a final written decision confirming the patentability of all of the Challenged Claims.

Respectfully submitted,

Dated: October 17, 2023 By: /*Peter Lambrianakos* /

Peter Lambrianakos (Reg. No. 58,279)
Lead Counsel for Patent Owner
Vincent J. Rubino, III (Reg. No. 68,594)
Back-up Counsel for Patent Owner
Enrique W. Iturralde (Reg. No. 72,883)
Back-up Counsel for Patent Owner
FABRICANT LLP
411 Theodore Fremd Avenue
Suite 206 South
Rye, New York 10580
Tel. 212-257-5797
Fax. 212-257-5796
Email: plambrianakos@fabricantllp.com
Email: vrubino@fabricantllp.com
Email: eiturralde@fabricantllpp.com

CERTIFICATE OF WORD COUNT

The undersigned hereby certifies that the portions of the above-captioned PATENT OWNER'S RESPONSE TO PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 11,169,775 specified in 37 C.F.R. § 42.24 has 8,326 words in compliance with the 14,000 word limit set forth in 37 C.F.R. § 42.24. This word count was prepared using Microsoft Word for Office 365.

Respectfully Submitted,

Dated: October 17, 2023

By: /*Peter Lambrianakos* /

Peter Lambrianakos (Reg. No. 58,279)
Lead Counsel for Patent Owner
FABRICANT LLP
411 Theodore Fremd Avenue,
Suite 206 South
Rye, New York 10580
Tel. 212-257-5797
Fax. 212-257-5796
Email: plambrianakos@fabricantllp.com

CERTIFICATE OF SERVICE

A copy of PATENT OWNER'S RESPONSE TO PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 11,169,775 and EXHIBITS 2002, 2003 and 2004 have been served on Petitioner's counsel of record as follows:

Elisabeth H. Hunt
Email: EHunt-PTAB@WolfGreenfield.com
Marc S. Johannes
Email: MJohannes-PTAB@WolfGreenfield.com
Gregory S. Nieberg
Email: GNieberg-PTAB@WolfGreenfield.com
WOLF, GREENFIELD & SACKS, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210-2206

Attorneys for Google LLC

October 17, 2023

By: /*Peter Lambrianakos* /
Peter Lambrianakos (Reg. No. 58,279)
FABRICANT LLP
411 Theodore Fremd Avenue
Suite 206 South
Rye, New York 10580
Tel. 212-257-5797
Fax. 212-257-5796
Email: plambriananakos@fabricantllp.com