

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SINGULAR COMPUTING LLC,
Patent Owner.

Patent No. 10,754,616
Filing Date: May 25, 2020
Issue Date: August 25, 2020

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

DECLARATION OF SUNIL P. KHATRI, Ph.D.

Case No. IPR2023-00395

TABLE OF CONTENTS

	<u>Page(s)</u>
I. INTRODUCTION	1
A. Background and Qualifications	1
1. Research and Professional Experience	2
2. Education	4
3. Publications, Awards, Grants, and Summary	5
II. COMPENSATION	8
III. MATERIALS CONSIDERED	9
IV. LEGAL PRINCIPLES	10
A. Level of Skill in the Art	10
B. The Claimed Invention of the '616 Patent	11
V. THE CITED REFERENCES	17
A. Stuttard	17
B. Shirazi	20
VI. CLAIM CONSTRUCTION	21
VII. THE CHALLENGED CLAIMS ARE VALID	21
A. Ground 1A: Claims 1-5 Are Not Obvious Over Stuttard in Combination with Shirazi	21
1. A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi	21
(a) A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi to Yield a Reduced Bitwidth Poly Unit	22
(b) A POSA Would Not Have Modified Stuttard to Be a Custom Computing Machine	25

2.	Petitioner’s Combination Does Not Disclose or Render Obvious “an input-output unit,” as Recited in Claim 1	29
B.	Ground 1B: Claims 7-14 Are Not Obvious Over Stuttard in View of Shirazi	30
1.	A POSA Would Not Have Been Motivated to Combine Stuttard and Shirazi	30
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].....	31
(a)	The 5000-LPHDR PE Limitation Carries Patentable Weight and Is Not Obvious	32
(b)	A POSA Would Not Have Stacked Large Numbers (<i>e.g.</i> , 5000) of Low Precision High Dynamic Range PEs in any Computer Architecture, Including the One Described by Stuttard	34
(c)	A POSA Would Not Have Made the Allegedly Obvious Implementations of the Scaled Stuttard-Shirazi Processor	36
(d)	A POSA Would Not Have Had a Reasonable Expectation of Success in Making the Proposed Stuttard-Shirazi Implementations with 5000+ PEs.....	41
(e)	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;”	42
C.	Ground 1C: Claims 16-22 and 24 Are Not Obvious Over Stuttard in Combination with Shirazi	42
1.	A POSA Would Not Modify Stuttard as Petitioner Proposes.....	42

2.	Petitioner’s Combination Does Not Render Obvious Limitation [16M]	43
VIII.	CONCLUSION	43

I, Sunil P. Khatri, declare as follows:

1. I have been asked by counsel for Patent Owner Singular Computing LLC (“Singular” or “Patent Owner”) to review U.S. Patent No. 10,754,616 (the “’616 Patent”) entitled PROCESSING WITH COMPACT ARITHMETIC PROCESSING ELEMENT, and to provide my technical review, analysis, insights, and opinions regarding the ’616 Patent in view of the prior art cited by Petitioner Google LLC (“Google” or “Petitioner”). I submit this declaration in support of Patent Owner’s Response in this IPR proceeding. I have personal knowledge of the matters stated herein and would be competent to testify to them if required.

2. I have been retained by Singular for the above-captioned *inter partes* review proceeding. I understand that the ’616 Patent is currently assigned to Singular Computing LLC.

3. I am over 18 years of age. I have personal knowledge of the facts stated in this Declaration and could testify competently if asked to do so.

I. INTRODUCTION

A. Background and Qualifications

4. I have reviewed and am familiar with the specification of the ’616 Patent. I understand that the ’616 Patent has been provided as Exhibit 1001. I will use the column and line numbers of the ’616 Patent to refer to particular portions of the specification (using the format [column no.]:[line no.]).

1. Research and Professional Experience

5. I have been serving as Professor in Electrical and Computer Engineering (ECE) at Texas A&M University, College Station, Texas, since September 2015. At Texas A&M, I have held the titles of Professor, Associate Professor (between September 2010 and September 2015), and Assistant Professor (between June 2004 and September 2010). My research is conducted in three main areas – computer systems, including computer architecture from the circuits up, algorithm acceleration using GPUs, FPGAs, and custom ICs and VLSI circuits; logic and its applications; and interdisciplinary extensions of the first two areas. A detailed description of each of these research areas is included in my *curriculum vitae* submitted concurrently herewith as **Exhibit 2003**.

6. From September 2020 to August 2021, I was on sabbatical at Arizona State University (Tempe, AZ) and the Air Force Research Laboratory (Rome, NY), continuing my past work on several topics including neuromorphic circuits, approximate computing, hardware security and hardware machine learning.

7. From September 2011 to August 2012, I was on sabbatical at the University of Texas at Austin, working with Professor Jacob Abraham on several topics, such as genomics, sinusoidal signal-based data transfer and medical electronics.

8. From January 2000 to May 2004, while working as an Assistant Professor in Electrical and Computer Engineering at the University of Colorado at Boulder, I performed research on VLSI logic design automation, VLSI layout design automation, VLSI design methodologies to address Deep Submicron (DSM) issues such as cross-talk and power, along with interdisciplinary extensions. One of the key research areas that I worked on had to do with regularity in Integrated Circuit design using a “layout fabric” which I developed during my Ph.D. research.

9. From August 1993 to December 1999, I worked as a Research Assistant with the CAD group under Professors Robert Brayton and Alberto Sangiovanni-Vincentelli at the University of California at Berkeley. My research topics included CAD and DSM design, Sets of Pairs of Functions to be Distinguished (SPFDs), Binary Decision Diagrams, Engineering Change, Hierarchical Synthesis and Verification, Model Matching and Combinational Verification, Timing Analysis in the Presence of Cross-talk and Multi-valued Logic Synthesis. My Ph.D. thesis was the first work that indicated that a regular layout “fabric” should be used to alleviate DSM problems like cross-talk, manufacturing issues, delay variation, and signal integrity.

10. From August 1989 to July 1993, I worked as a Design Engineer with Motorola’s MC88110 RISC and PowerPC 603 microprocessor groups in Austin, Texas. During that time, I was involved in various design areas from Design for

Testability to Digital and Analog Circuit Design and high-level design. I was also independently responsible for the design of the factory test controller for the MC88110 and became familiar with various ad-hoc and structured test methodologies. I designed digital and analog circuitry, as well as the MC88110's input/output buffers and clock PLL logic. In addition, I performed attendant tasks in a "vertical" VLSI design methodology, including high-level modeling, layout design and verification, full-chip integration, and global and detailed routing.

11. From August 1988 to July 1993, I worked as a Researcher with Professor M. Ray Mercer's group at the University of Texas at Austin, researching topics such as IC testing and Boolean function representation using Canonical XOR-based circuit decompositions, using a multi-level network of symmetric functions.

12. From August 1987 to July 1989, I worked as a Researcher at the University of Texas at Austin. My research included Computer Architecture and Memory Interface design, applied in the context of the METRIC multi-threaded RISC microprocessor which was being developed by Professors Donald Fussell and Roy Jenevein at the time.

2. Education

13. In 1987, I received a Bachelor of Science degree in Electrical Engineering from the Indian Institute of Technology in Kanpur, India. I maintained a GPA of 3.72 and was ranked fourth in my class of sixty students.

14. In 1989, I obtained my Master of Science degree from the Department of Electrical and Computer Engineering at the University of Texas in Austin, Texas. I was awarded the Microelectronics and Computer Development (MCD) Fellowship and maintained a GPA of 3.909.

15. The thesis for my Master's degree was titled "*The Design of the METRIC Memory Interface and Memory System*" and involved research surrounding the design of the memory interface of METRIC, a multi-threaded RISC Microprocessor.

16. From 1993 to 1999, I attended the University of California at Berkeley, where I obtained my Doctorate Degree from the Department of Electrical Engineering and Computer Sciences. From 1993 to 1994, I was awarded the California MICRO Fellowship. I maintained a GPA of 3.963 during my Ph.D. studies.

17. My Dissertation was titled "Cross-talk Noise Immune VLSI Design using Regular Layout Fabrics".

3. Publications, Awards, Grants, and Summary

18. I have authored a total of over 280 peer-reviewed publications, including 45 journal papers, 193 conference papers, and 42 workshop papers. Among these papers, five have received a best paper award, while seven other papers have received best paper nominations. Additional journal papers and conference

papers are currently undergoing peer review. In addition, I have co-authored 9 research monographs, 1 edited research monograph, and 4 book chapters.

19. I was invited to serve as a panelist at a conference seven times and have presented three conference tutorials.

20. I am currently a named inventor on six United States patents and one provisional patent application, with two applications currently pending.

21. I have four current research grants totaling \$4.37M (of which my portion is \$592,500). The total amount of the grants in which I have been involved to date is \$17.53 million, of which \$2.85 million is my portion. Some of these grants are with colleagues in the Electrical and Computer Engineering Department, as well as other academic departments at Texas A&M University. My research has been funded both through the government and through industrial sources.

22. I currently serve, or have served, as the following:

- ☐ Associate Editor, ACM Transactions on Design Automation of Electronic Systems
- ☐ Associate Editor, IEEE Transactions on Computers
- ☐ Associated Editor, MDPI Journal of Electronics
- ☐ EDA Track Co-Chair for ICECS (2014)
- ☐ Panel Chair for Texas WISE (2014)

- ☐ Track Co-Chair for ICECS (VLSI Systems, Applications and Computer Aided Design Track) (2013)
- ☐ Poster Session Chair for Texas WISE (2013)
- ☐ Advisory Committee for HotPI (2013)
- ☐ Panel Session Chair for SLiP (2013)
- ☐ Track Chair (Logic Track) for ICCAD (2009-2010 and 2015-2017)
- ☐ Track Chair (Logic Track) for DAC (2016-2017)
- ☐ General Chair for IWLS (2009)
- ☐ Technical Program Chair for IWLS (2008)
- ☐ Track Co-Chair (Computer Aided Network Design (CANDE) Track) for ISCAS (2008-2010)
- ☐ Track Co-Chair (Test and Methodologies Track) for ICCD (2007)
- ☐ Panel Chair for ITSW (2009)
- ☐ Publicity Co-Chair for GLS-VLSI (2009)
- ☐ Member of the TPC (several conferences)
- ☐ Session Chair (several conferences)

23. I have also received several awards, including the “Outstanding Professor Award” from the Electrical and Computer Engineering Department at Texas A&M University in 2007 and 2020, the “Association of Former Students’

Distinguished Achievement Award in Teaching” in 2009, and the “Association of Former Students’ College-level Teaching Award” in 2019.

24. I have over 34 years of research and professional experience in the field of Electrical and Computer Engineering, including in the areas of computer systems, logic synthesis, and algorithms for VLSI Design and interdisciplinary extensions.

25. I have taught numerous undergraduate and graduate level courses at Texas A&M in the area of Electrical and Computer Engineering. I have graduated twelve Ph.D. students, nineteen M.S. students, and twelve B.S. Honors students. I am currently advising six Ph.D. students, four M.S. students, and two B.S. research students. I have also advised 40 undergraduates, four of which received an award for their research. Eight research papers in international conferences (one invited) have resulted from my work with undergraduates, and the dissertation of one of my Ph.D. students was nominated for the ACM Best Dissertation Award in 2014.

26. On the basis of my education and the experience described above, I am qualified to give the opinions set out herein.

II. COMPENSATION

27. My compensation for time worked on this proceeding is not dependent on the outcome of this proceeding, or the substance of my opinions. My compensation for time worked on this proceeding is at my customary rate of

\$700/hour. I have no financial interest in, or affiliation with, the Patent Owner or any of the real parties in interest.

III. MATERIALS CONSIDERED

28. In providing my technical review, analysis, insights, and opinions, I have considered the '616 Patent and its prosecution history.

29. I have also considered the Petition filed by the Petitioner in this proceeding and the relevant exhibits relied on by Petitioner, including the expert declaration submitted by Paul Franzon, Ph.D., and the Board's Decision Granting Institution of *Inter Partes* Review (DI). I have also considered the exhibits cited herein, including the transcript of Dr. Franzon's deposition. To the extent that this declaration does not explicitly reference specific arguments, positions, opinions, or statements made by Petitioner – or by Petitioner's expert, Dr. Franzon – this should not be construed to mean that I agree with those arguments, positions, opinions, or statements.

30. I have also considered my own experience and knowledge, as discussed above and described more fully in my CV, in the areas including VLSI circuit design, computer architecture, approximate computing, parallel computing, parallel software, and hardware-software co-design.

IV. LEGAL PRINCIPLES

31. I understand that a patent claim is unpatentable as “obvious” if the subject matter of the claim as a whole would have been obvious to a person of ordinary skill in the art (POSA) as of the time of the invention at issue.

32. I understand that the use of “the person of ordinary skill” rubric is to prevent one from improperly, in the present day, using hindsight to decide whether a claim is obvious.

33. I understand that the following factors must be evaluated to determine whether the claimed subject matter is obvious: (1) the scope and content of the prior art; (2) the difference or differences, if any, between the scope of the patent claim and the scope of the prior art; and (3) the level of ordinary skill in the art at the time of the invention.

34. I understand that I must construe a claim in accordance with the ordinary and customary meaning of the language of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

A. Level of Skill in the Art

35. I understand that I should perform my analysis from the viewpoint of a person of ordinary skill in the art. I understand that this hypothetical person of ordinary skill in the art is considered to have the normal skills of a person in a certain technical field. I understand that factors that may be considered in determining the

level of ordinary skill in the art include, *e.g.*, the types of problems encountered in the art, prior art solutions to those problems, the sophistication of the technology, and the education level of active workers in the field.

36. I agree with Petitioner's proposed level of skill in the art, except that I disagree that such a person would have more than two years of experience. Therefore, a 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. Pet. at 4.

B. The Claimed Invention of the '616 Patent

37. The '616 Patent is entitled "Processing with compact arithmetic processing element" and issued on August 25, 2020. The '616 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. I have reviewed the '616 Patent and its file history.

38. The '616 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.

39. 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.

Id. at 2:62-3:10.

40. However, Dr. Bates realized that such full-precision, inefficient components were 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.

'616 Patent, 3:11-17.

41. The '616 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.

Id. at 5:56-6:4.

42. By using an LPHDR architecture as claimed,

[T]he 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.

Id. at 6:13-19.

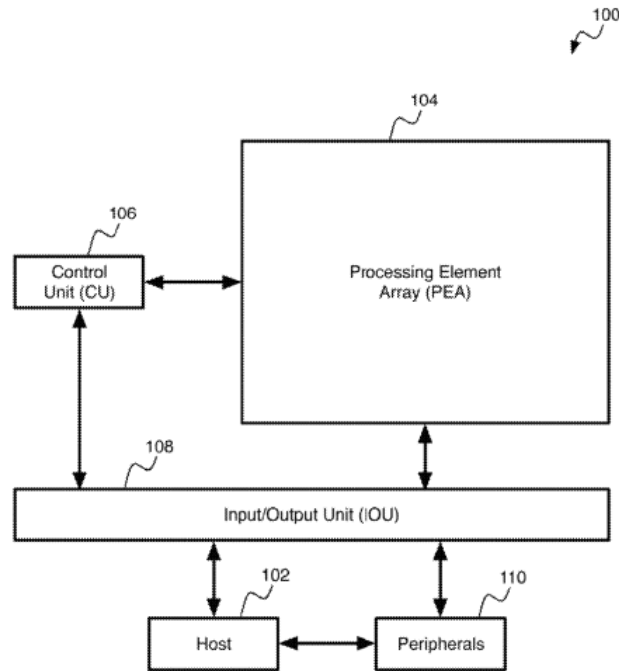
43. Indeed, “[b]ecause LPHDR processing elements are relatively small, a single processor or other device may include a very large number of LPHDR processing elements, adapted to operate in parallel with each other.” *Id.* at 6:49-52. As a result, “embodiments of the present invention may be implemented 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.* at 8:3-7. In short, the ’616 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.

44. The '616 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.” *Id.*, 6:58-61.

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., 7:3-9. The patent shows examples of these benefits (*see* 17:29-23:51), concluding that “[t]he surprising results, below, show that even with this extreme level of imprecise . . . LPHDR arithmetic, useful results can be achieved” *Id.*, 20:58-60.

45. In particular, the '616 Patent teaches a new computing architecture based on a massively parallel array of LPHDR processing elements (PEA). '616 Patent at 8:12-28. An embodiment is shown in Figure 1:



Id., FIG. 1.

46. In this embodiment, the “Host 102 is responsible for the overall control of the computing system 100.” *Id.*, 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.*, 8:51-54. 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.

47. 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. *Id.*, 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.*, 10:4-7.

V. THE CITED REFERENCES

A. Stuttard

48. I have reviewed PCT Publication No. WO2005/037326A2, entitled "Unified SIMD Processor" ("Stuttard"). 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:

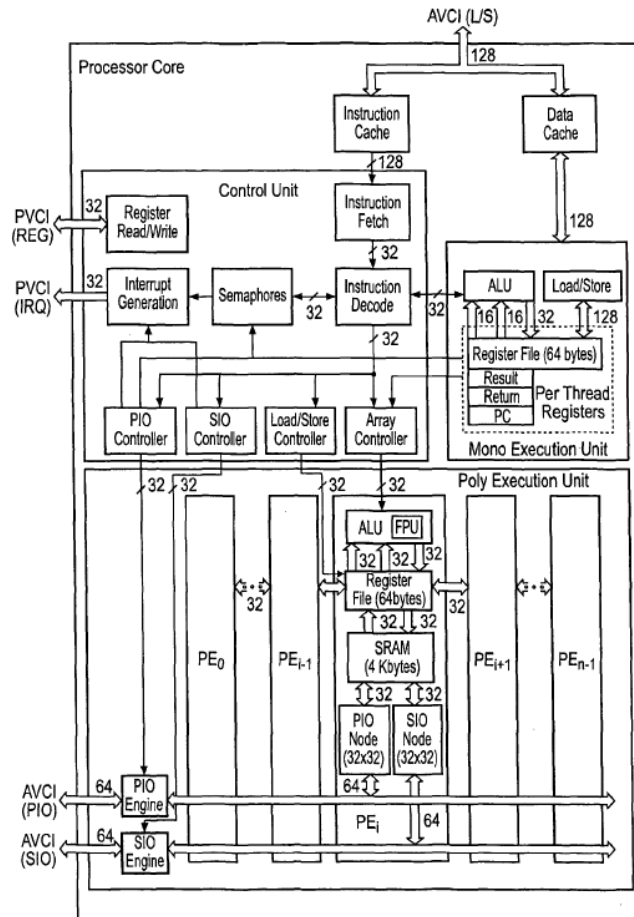


Fig. 7

Stuttard, Fig. 7.

49. 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.

50. 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.” *Id.*, 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).

51. The MTAP processor contains two I/O mechanisms:

The first, *Programmed I/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 I/O* (SIO) which allows chunks of contiguous data to be streamed directly into the memory within PEs.

Id., 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. Franzon, 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.

B. Shirazi

52. I have reviewed “Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines,” by Nabeel Shirazi, *et al.*, published in Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines (“Shirazi”) (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 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/subtractors, multipliers, and dividers implemented in FPGAs. *Id.*, 155-162.

VI. CLAIM CONSTRUCTION

53. In my declaration, I apply the plain and ordinary meaning of the claims as understood by a POSA as of 2009.

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

54. In my opinion, the combination of Stuttard and Shirazi does not render the claims obvious because they fail to rebut Dr. Bates' observation that a POSA would not believe that LPHDR computations in an array of processors are useful. Petitioner's motivations to combine are not supported by the facts and are inconsistent with the inventive concept underlying Stuttard.

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

55. 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 '616 Patent, it was thought that deploying a large number of LPHDR multipliers was a poor design choice. '616 Patent, 6:57-62, 7:3-9. 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.

56. 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.

57. 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). In my opinion, 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 ’616 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 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.

58. 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. I have reviewed these references and, in my opinion, as in the other cited references, none teach taking advantage of that size savings by implementing them in a massively parallel processing array.

59. The Petition comes to the 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, ¶¶ 91-92) (emphasis added). I have reviewed the references and, in my opinion, these references do not suggest putting more PEs on a single chip for achieving higher parallelism or to attain any other benefit.

60. 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. The Petition presents no evidence of any processing array having been implemented with reduced bitwidth floating point formats. In my opinion, 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).

61. 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). Figure 7 also shows that the ALU receives 32-bit inputs. 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, in my opinion, 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. In my opinion, Petitioner’s motivation to combine is therefore unsupported.

62. In addition, the Petition fails to rebut the evidence in the ’616 Patent specification that POSAs did not believe that parallel processing using low-precision arithmetic was useful. ’616 Patent, 6:57-62, 7:3-9. In my opinion, 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 corroborates that POSAs did not view the architecture taught by the ’616 Patent as viable before it was disclosed by Dr. Bates.

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

63. In my opinion, the Petition contains no 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, in my opinion, 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.

64. 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.

65. 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.

66. In my opinion, 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). 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, 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.

67. In my opinion, the Petition presents no evidence addressing this deficiency in its motivation-to-combine analysis. The Petition contends 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 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.” *Id.*, 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. In my opinion, therefore, the 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 narrowing the capabilities of the PEs. Further, replacing these by Shirazi's units (18-bit for 2-D FFT and 16-bit for FIR filters) would potentially *increase* their size. In my opinion, 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.

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

68. In my opinion, 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.

69. 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. Ex. 2004, 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.

70. In my opinion, 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 of 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.

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

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

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

72. My understanding is that Petitioner's obviousness arguments in Ground 1B depend entirely on its Stuttard-Shirazi combination discussed in Ground 1A. Pet. at 46-52. In my opinion, for the same reasons I discuss above in connection with Ground 1A, Google has not shown that a POSA would be motivated to modify Stuttard to adopt Shirazi's bit-width floating point formats.

**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]**

73. In my opinion, the Petition contains no evidence why a POSA would have chosen to include a minimum of 5000 LPHDR PEs in a PEA. I understand that the Board “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. DI, 49. I agree with the Board’s skepticism. In my opinion, Stuttard’s statement in Stuttard that the “poly execution unit may consist of tens, hundreds *or even thousands* of PE’s,” Stuttard, 10:12 (emphasis added), was merely aspirational and would not motivate a POSA to create a poly unit specifically with at least 5000 PEs. Pet., 46. In my opinion, the Petition contains no evidence why a POSA would have chosen to include a minimum of 5000 LPHDR PEs in the array.

74. I agree with the Board’s finding 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, 49. In my opinion, none of the evidence in the Petition supports that a POSA would have been motivated to make a massively parallel processor of LPHDR PEs, especially

in view of the '616 Patent's disclosure that POSAs believed that "massive amounts of LPHDR arithmetic" are not useful. '616 Patent, 7:3-9, 6:57-62.

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

75. I understand that Petitioner asserts that the limitation carries no patentable weight because an array of that size provides no new and unexpected results. Pet. at 48-50. In my opinion, Petitioner offers no affirmative evidence in support of this assertion. Instead, Petitioner argues that the '616 Patent specification does not disclose that any particular number of PE's provides any particular benefit. Pet. at 49 (citing Ex. 1003, ¶ 160). Petitioner's argument ignores columns of disclosure in the '616 Patent that spell 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. '616 Patent, 7:3-38 (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. ’616 Patent, 28:13-29:4. In my opinion, in view of the un rebutted 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.

76. 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 49-50. As the ’616 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 (’616 Patent, 7:3-38, 6:57-62), 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 and 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.

77. 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. I have reviewed that reference and note that it 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.

78. In my opinion, this limitation is not 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

79. I agree with the Board’s finding that Petitioner presented no evidence supporting “a motivation to scale up to at least 5,000 *low precision* PEs.” DI, 49.

80. A central advance of the ’616 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. ’616 Patent, 7:3-38, 20:44-51, 20:58-60. In my opinion, 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 ’616 Patent, respectively, contain no evidence 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. '616 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”). Petitioner simply assumes that a POSA would be aware of this discovery despite providing no evidence to this end.

81. 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 '616 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. The only evidence of record is the '616 Patent's teaching that such an architecture was not obvious to a POSA.

82. In my opinion, none of the evidence of scaling PE's cited in the Petition suggests scaling up to 5000 or more LPHDR PE's as the Board observed. DI, 49. The Petition cites references that purport to address the advantages of scaling PEs, Pet., 46-48, but the references do not address scaling LPHDR PE's and the problems specific to that implementation. The Petition 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 '616 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.

(c) A POSA Would Not Have Made the Allegedly Obvious Implementations of the Scaled Stuttard-Shirazi Processor

83. I have reviewed Petitioner's proposed obvious implementations of a scaled Stuttard-Shirazi combinations to meet the claims. I disagree that any of these implementations would have been obvious to a POSA.

84. Petitioner's first obviousness implementation expands the poly unit to include at least 5000 of Shirazi's PEs because "POSAs would have been motivated to scale the poly unit in a single MTAP core to include as many 'thousands of PEs' as could reasonably be achieved using available process technology." Pet., 51. 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.

85. The '616 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. '616 Patent, 9:57-62, FIGS. 2, 3. In my opinion, 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.

86. 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., 17-18. I have reviewed these references and, in my opinion, 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.

87. In my opinion, 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. In my opinion, 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.

88. In another alternative implementation, Petitioner suggests putting multiple processor cores on the same chip. Pet., 51. The result would be a single processor with several mono units as well as 5000+ PEs in several poly units. In my opinion, 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 by replicating the processor core as the Petition contends.

89. In my opinion, Petitioner's proposed implementations of Stuttard-Shirazi to meet the 5000-PE limitation would not have been obvious.

**(d) A POSA Would Not Have Had a Reasonable
Expectation of Success in Making the Proposed
Stuttard-Shirazi Implementations with 5000+ PEs**

90. In my opinion, for the reasons set forth in the preceding section which would have discouraged a POSA from making the proposed 5000-PE implementations, a POSA would not have reasonably expected to succeed in making those implementations.

91. Petitioner's discussion on likelihood of success is limited to discussing whether existing process technology would have permitted placing 5000 PEs on a single chip. Pet., 52-58. Petitioner presents no evidence regarding the technical challenges associated with the hardware and software redesign needed for a massively parallel system or for a 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. In my opinion, the resulting hardware would entail a significantly different architecture and implementation which would be beyond the skill of a POSA.

92. In the case of a multiple-processor core chip design, 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 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. A POSA with a bachelor's degree and two years' experience in computer architecture would be unlikely to succeed in overcoming these challenges.

**(e) 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;”**

93. I understand that Petitioner relies on the same arguments as with Claim 1's input-output unit limitation. Pet. at 60. For the same reasons as discussed above in Section VII.A.2, therefore, Petitioner's Ground 1B combination does not render this limitation obvious.

**C. Ground 1C: Claims 16-22 and 24 Are Not Obvious Over
Stuttard in Combination with Shirazi**

**1. A POSA Would Not Modify Stuttard as Petitioner
Proposes**

94. I understand that Petitioner's obviousness arguments depend entirely on its Stuttard-Shirazi combination discussed above, although Petitioner does not discuss the motivation to make the combination specifically in connection with Ground 1C. Pet. at 72-74. 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.

**2. Petitioner's Combination Does Not Render Obvious
Limitation [16M]**

95. Petitioner expressly relies on its Ground 1A arguments as to this limitation. Pet. at 74-75. For the same reasons as discussed above, in my opinion, the combination does not render this limitation obvious.

VIII. CONCLUSION

96. In signing this declaration, I recognize that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed this 17th Day of October, 2023

Respectfully submitted,



Sunil P Khatri, Ph.D.