

Paper No. ____

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SINGULAR COMPUTING LLC,
Patent Owner.

Case No. IPR2023-00397
Patent No. 11,169,775

**PETITION FOR INTER PARTES REVIEW
UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.1 et seq**

TABLE OF CONTENTS

I.	MANDATORY NOTICES	xvii
A.	Real Party-In-Interest	xvii
B.	Related Matters.....	xvii
1.	United States Patent & Trademark Office	xvii
2.	United States Patent Trial and Appeal Board	xviii
3.	U.S. District Court for the District of Massachusetts	xviii
C.	Counsel and Service Information - § 42.8(b)(3) and (4).....	xviii
II.	FOUNDATIONS AND STANDING	1
III.	'775 PATENT.....	1
A.	Specification	2
B.	Claims	3
C.	Person of Ordinary Skill in the Art (“POSA”)	4
D.	Prosecution History	4
E.	Priority Dates	4
IV.	CLAIM INTERPRETATION	6
V.	<u>GROUND 1A</u> : CLAIMS 1-5 ARE UNPATENTABLE OVER STUTTARD IN COMBINATION WITH SHIRAZI	6
A.	The Stuttard-Shirazi Processor	6
1.	Stuttard (EX1006)	6
2.	Shirazi (EX1036).....	9
3.	Stuttard-Shirazi Combination.....	11
a.	Reduced-Bitwidth Poly PEs.....	11
b.	Two-Dimensional PE Array.....	14
B.	Claim 1	20
1.	[1PRE]: “A computing system, comprising:”	20
2.	[1A]: “a host computer;”	20
3.	[1B]: “a computing chip comprising:”	21

4. [1C]: “a processing element array comprising” “first” and “second” “edge processing element[s] positioned at a first edge of the processing element array” and “first” and “second” “interior processing element[s] positioned at [first and second] location[s] in the interior of the processing element array”	21
5. [1D]: “a first processing element connection connecting the first edge processing element with the first interior processing element; a second processing element connection connecting the second edge processing element with the second interior processing element;”	23
6. [1E]: “an input-output unit connected to the first” and “second edge processing element[s];”	25
7. [1F]: Each Processing Element of Limitation [1C] Has a “memory local to” It.....	30
8. [1H]: Each Processing Element of Limitation [1C] Comprises an “arithmetic unit”	31
9. [1I]: “a host connection at least partially connecting the input-output unit with the host computer;”	32
10. [1J]: “wherein the first, second, third and fourth arithmetic units each comprises a corresponding multiplier circuit adapted to receive as a first input to the corresponding multiplier circuit a first floating point value having a first binary mantissa of width no more than 11 bits and a first binary exponent of width at least 6 bits, and to receive as a second input to the corresponding multiplier circuit a second floating point value having a second binary mantissa of width no more than 11 bits and a second binary exponent of width at least 6 bits;”	34
11. [1G], [1K]: “a fifth arithmetic unit;” “wherein the fifth arithmetic unit comprises a corresponding multiplier circuit adapted to receive as inputs to the corresponding multiplier circuit two floating point values each of width at least 32 bits;”	37
12. [1L]: Each Multiplier Circuit of [1J] “comprises a [first/second/third/fourth] plurality of transistors and has no other transistors”	39
13. [1M]: “the multiplier circuit corresponding to the fifth arithmetic unit comprises a fifth plurality of transistors; and,”	40

14. [1N]: the “fifth plurality of transistors exceeds in number each of the” first through fourth “plurality of transistors”	40
C. Claim 2: “said host computer is programmed to provide instructions to said computing chip that, when executed, cause said processing element array to perform an operation whose output is used to identify at least one image, from a plurality of images to be searched, that is similar to at least one input image.”	41
D. Claim 3	45
1. [3A]: “the computing chip further comprises a control unit connected to the processing element array,”	45
2. [3B]: “the control unit comprising circuitry adapted to decode at least one instruction received from the host computer via the input-output unit, and”	46
3. [3C]: “to send at least one control signal to the processing element array to cause the processing element array to operate according to the at least one instruction.”	47
E. Claim 4: “the computing chip further comprises an instruction memory adapted to store the at least one instruction received from the host computer via the input-output unit, wherein the control unit is further adapted to retrieve the at least one instruction from the instruction memory.”	48
F. Claim 5: “the width of each [first/second] binary mantissa is no more than 10 bits plus a [first/second] sign bit”	50
VI. <u>GROUND 1B: CLAIMS 7-14, 16-23 ARE UNPATENTABLE OVER STUTTARD IN COMBINATION WITH SHIRAZI</u>	51
A. Scaling the Stuttard-Shirazi Processor	51
1. 5,000+ Processing Elements	51
2. Plurality of 32-Bit Floating-Point Multipliers	55
3. Reasonable Expectation of Success	60
B. Claim 7	65
1. [7PRE] “A computing system, comprising:” [7A] “a host computer;” [7B] “a computing chip comprising:”	65
2. [7C]: “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;” 66
3. [7D]: “an input-output unit connected to each of the first subset of the plurality of first processing elements;” 67
 4. [7E]: “a plurality of processing element connections, each processing element connection connecting one of the plurality of first processing elements with another of the plurality of first processing elements, wherein each of the plurality of first processing elements is connected to at least one other of the plurality of first processing elements by at least one of the plurality of processing element connections;” 68
 5. [7F]: “a plurality of memory units, wherein each of the plurality of first processing elements is associated with a corresponding one of the plurality of memory units, and wherein each of the plurality of memory units is local to its associated one of the plurality of first processing elements;” 69
 6. [7G]: “a plurality of first arithmetic units, wherein each of the plurality of first processing elements has positioned therein at least one of the plurality of first arithmetic units;” 70
 7. [7H]: “a plurality of second processing elements; and a plurality of second arithmetic units, wherein each of the plurality of second processing elements has positioned therein at least one of the plurality of second arithmetic units; and” 71
 8. [7I]: “a host connection at least partially connecting the input-output unit with the host computer;” 72
 9. [7J]: “wherein the plurality of first arithmetic units each comprises a first corresponding multiplier circuit adapted to receive” the Same Inputs As Recited in [1J].....72
 10. [7K]: “wherein the first multiplier circuits corresponding to the plurality of first arithmetic units each comprises a first respective plurality of transistors and has no other transistors;” 72

11. [7L]: “wherein the plurality of second arithmetic units each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;”	72
12. [7M]: “wherein the second multiplier circuits corresponding to the plurality of second arithmetic units each comprises a second respective plurality of transistors;”	73
13. [7N]: “wherein each of the second respective pluralities of transistors of the second multiplier circuits corresponding to the plurality of second arithmetic units exceeds in number each of the first respective pluralities of transistors of the first multiplier circuits corresponding to the plurality of first arithmetic units.”	73
C. Claim 8	73
1. [8A]: “other than the plurality of second processing elements, the computing chip has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and”	73
2. [8B]: “wherein the plurality of first processing elements exceeds in number, by at least 100, the plurality of second processing elements.”	74
D. Claim 9: “the plurality of memory units each comprises a corresponding register.”	74
E. Claim 10	75
F. Claim 11: “an aggregate maximum bandwidth of the plurality of processing element connections exceeds a maximum bandwidth of the host connection.”	75
G. Claims 12-14.....	77
H. Claim 16	77
1. [16PRE]: “A circuit board connected to a host computer, the circuit board comprising:”	77
2. [16A]: “a plurality of processing element arrays, each comprising:”	79
3. [16B]-[16H].....	80
I. Claims 17-19.....	80

J. Claims 20-23.....	81
VII. PUBLIC ACCESSIBILITY OF CITED PRINTED PUBLICATIONS	82
A. Prior to June 2008.....	82
B. Prior to June 19, 2009.....	85
C. Prior to June 2010.....	86
VIII. SECONDARY CONSIDERATIONS	87
IX. NO BASIS EXISTS FOR DISCRETIONARY DENIAL	88
X. CONCLUSION.....	88
XI. APPENDIX: CLAIM LISTING	89

TABLE OF AUTHORITIES

CASES

<i>E.I. DuPont de Nemours & Co. v. Synvina C.V.</i> , 904 F.3d 996 (Fed. Cir. 2018)	55
<i>Ericsson v. Intellectual Ventures I</i> , IPR2014-00527, Paper 41 (May 18, 2015)	82
<i>Ex parte Sarkar</i> , Appeal No. 2021-003873, 2022 WL 337233 (Feb. 2, 2022)	65
<i>Genentech, Inc. v. Hospira, Inc.</i> , 946 F.3d 1333 (Fed. Cir. 2020)	55
<i>Hulu v. Sound View Innovations</i> , IPR2018-01039, Paper 29 (Dec. 20, 2019)	82
<i>In re Applied Materials, Inc.</i> , 692 F.3d 1289 (Fed. Cir. 2012)	54, 55
<i>In re Blaser</i> , 556 F.2d 534 (C.C.P.A. 1977)	5
<i>In re Epstein</i> , 32 F.3d 1559 (Fed. Cir. 1994)	65
<i>In re Harza</i> , 274 F.2d 669 (C.C.P.A. 1960)	53, 54
<i>In re Urbanski</i> , 809 F.3d 1237 (Fed. Cir. 2016)	55
<i>In re Wertheim</i> , 646 F.2d 527 (C.C.P.A. 1981)	5
<i>In re Yale</i> , 434 F.2d 666 (C.C.P.A. 1970)	26
<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	18, 52, 56, 79

<i>Lockwood v. Am. Airlines</i> , 107 F.3d 1565 (Fed. Cir. 1997)	5
<i>Microsoft Corp. v. IPA Techs. Inc.</i> , IPR2019-00810, Paper 44 (Oct. 15, 2020)	82, 84
<i>Purdue Pharma L.P. v. Faulding Inc.</i> , 230 F.3d 1320 (Fed. Cir. 2000)	5
<i>Sony Corp. v. Creative Tech.</i> , IPR2016-01407, Paper 13 (Dec. 6, 2016)	87
<i>Supercell Oy v. Gree, Inc.</i> , PGR2018-00008, Paper 42 (Jan. 2, 2019)	54

STATUTES

35 U.S.C. § 102(a)	85, 86
35 U.S.C. § 102(b)	1, 9, 82, 85
35 U.S.C. § 103	1
35 U.S.C. § 112	5
35 U.S.C. § 282(b)	6
35 U.S.C. § 314(a)	88
35 U.S.C. § 325(d)	88

REGULATIONS

37 C.F.R. §42.100(b)	6
37 C.F.R. §42.104(a)	1

APPENDIX LISTING OF EXHIBITS

Exhibit	Description
1001	U.S. Patent No. 11,169,775
1002	Prosecution History of U.S. Patent No. 11,169,775
1003	Declaration of Paul Franzon, PhD.
1004	Curriculum Vitae of Paul Franzon, PhD.
1005	Declaration of Dr. Mary K. Bolin
1006	PCT Publication WO2005/037326A2 (“Stuttard”)
1007	Schmidt et al., “KPROC - An Instruction Systolic Architecture for Parallel Prefix Applications,” <i>Parallel and Distributed Computing Practices</i> Vol. 3, No. 2, pp. 163–178 (2000) (“Schmidt-2000”)
1008	Schimmler et al., “A bit-serial floating-point unit for a massively parallel system on a chip,” in <i>Parallel Algorithms and Applications</i> , 19:2-3, 79-95 (2004)
1009	Schmidt et al., “A Parallel Accelerator Architecture for Multimedia Video Compression,” in <i>Euro-Par’99, Lecture Notes in Computer Science</i> 1685, Springer Verlag, 1999, pp. 950–960 (“Schmidt-1999”)
1010	Schimmler et al., “The Instruction Systolic Array – Implementation of a Low-Cost Parallel Architecture as Add-On Board for Personal Computers,” <i>High Performance Computing and Networking, Lecture Notes in Computer Science</i> 797, Springer Verlag, 1994, pp. 487–488 (“Schimmler-1994”)
1011	Schimmler et al., “The Instruction Systolic Array in Image Processing Applications,” <i>Vision Systems: Sensors, Sensor systems, and Components</i> , Europto 96, pp. 136–144 (“Schimmler-1996”)
1012	Yeh et al., “Parallax: An architecture for real-time physics,” the 34th International Symposium on Computer Architecture (ISCA), 2007, pp. 232-243 (“Yeh-ISCA07”)
1013	Yeh et al., “The Art of Deception: Adaptive Precision Reduction for Area Efficient Physics Acceleration,” 40th IEEE/ACM International Symposium on Microarchitecture (MICRO’07), 2007, pp. 394-406 (“Yeh-Micro07”)
1014	Yeh et al., “Fool me twice: Exploring and exploiting error tolerance in physics-based animation” (2007)
1015	U.S. Patent No. 5,355,508 (“Kan”)
1016	G.E. Moore, “Cramming More Components onto Integrated Circuits,” <i>Proceedings of the IEEE</i> , Vol. 86, No. 1 (January 1998)

Exhibit	Description
1017	IEEE SSCS Newsletter 36-37 (reprinting Moore, “Progress in Digital Integrated Electronics,” International Electron Devices Meeting, IEEE, 1975, pp. 11-13).
1018	<i>Microprocessor Report</i> , “Mendocino Improves Celeron,” Volume 12, Number 11, August 24, 1998
1019	Sim et. al, “MIMD-SIMD hybrid system—towards a new low cost parallel system,” <i>Parallel Computing</i> 29 (2003) pp. 21-36 (“Sim”)
1020	Taveniku et. al. “The VEGA Moderately Parallel MIMD, Moderately Parallel SIMD, Architecture for High Performance Array Signal Processing,” <i>Proceedings of IPPS/SPDP’98</i> (IEEE Computer Society), pp 226-232 (“Taveniku”).
1021	The National Technology Roadmap for Semiconductors, 1997 Edition (“NTRS-1997”)
1022	International Technology Roadmap for Semiconductor, 1999 Edition (“ITRS-1999”)
1023	International Technology Roadmap for Semiconductors, 2000 Update (“ITRS-2000”) (from Exhibit B, Folders 1-2 of the Declaration of Nathaniel E Frank-White, EX1029)
1024	International Technology Roadmap for Semiconductor, 2002 Update (“ITRS-2002”) (from Exhibit B, Folders 3-7 of the Declaration of Nathaniel E Frank-White, EX1029)
1025	International Technology Roadmap for Semiconductors, 2004 Update (“ITRS-2004”) (from Folders 8-11 of the Declaration of Nathaniel E Frank-White, EX1029)
1026	International Technology Roadmap for Semiconductors, 2007 Edition (“ITRS-2007”) (from Folders 12-13 of the Declaration of Nathaniel E Frank-White, EX1029)
1027	International Technology Roadmap for Semiconductors, 2009 Edition (“ITRS-2009”) (from Folders 14-15 of the Declaration of Nathaniel E Frank-White, EX1029)
1028	International Technology Roadmap for Semiconductors, 2010 Update (“ITRS-2010”) (from Folder 16 of the Declaration of Nathaniel E Frank-White, EX1029)
1029	Declaration of Nathaniel E Frank-White (November 30, 2022) (ITRS)
1030	International Technology Roadmap for Semiconductor, 2001 Edition (“ITRS-2001”)
1031	Declaration of Nathaniel E Frank-White (November 30, 2022) (Schmidt)

Exhibit	Description
1032	Declaration of Nathaniel E Frank-White (November 30, 2022) (Yeh)
1033	Gargini, “The International Technology Roadmap for Semiconductors (ITRS): Past, present and future,” GaAs IC Symposium. IEEE Gallium Arsenide Integrated Circuits Symposium. 22nd Annual Technical Digest 2000
1034	Tong et al., “Reducing Power by Optimizing the Necessary Precision/Range of Floating-Point Arithmetic,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 8, No. 3, June 2000 (“Tong”) (from pages 6-19 of the Declaration of Gerard P. Grenier, EX1035)
1035	Declaration of Gerard P. Grenier regarding Tong et al., Reducing Power by Optimizing the Necessary Precision/Range of Floating-Point Arithmetic, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 8, No. 3, June 2000
1036	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, April 19-21, 1995 (“Shirazi”) (from pages 6-13 of Declaration of Gordon MacPherson, EX1037)
1037	Declaration of Gordon MacPherson regarding 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, April 19-21, 1995
1038	Altera Product Catalog (2008) (from pages 82-153 of Declaration of Nathaniel E Frank-White, EX1047)
1039	Stackhouse et al., “A 65 nm 2-Billion Transistor Quad-Core Itanium Processor,” in IEEE Journal of Solid-State Circuits, vol. 44, no. 1, pp. 18-31, Jan. 2009 (“Intel-Itanium”)
1040	Dell White Paper, “Performance Analysis of Intel® Pentium® M and Pentium 4 Processors,” July 2005. (“Pentium-4”) (from pages 61-65 of Declaration of Nathaniel E Frank-White, EX1047)
1041	The Guru of 3D Report on Radeon HD Series 5700 (“Radeon-5700”) (from pages 10-11 of Declaration of Nathaniel E Frank-White, EX1047)
1042	“Intel Core 2 Extreme QX9650 – Penryn Ticks Ahead,” Anandtech.com (Oct. 29, 2007) (from pages 30-40 of Declaration of Nathaniel E Frank-White, EX1047)

Exhibit	Description
1043	TSMC Report (2008), “TSMC Ramps Most Advanced Available Process Technology to Volume Production” (pr.tsmc.com/english/news/1526) (“TSMC-40nm”)
1044	“Updated CPU Cheatsheet – Seven Years of Covert CPU Operations,” Anandtech.com (Aug. 28, 2004) (from pages 42-49 of Declaration of Nathaniel E Frank-White, EX1047)
1045	Natarajan et al., “A 32nm logic technology featuring 2nd-generation high-k + metal-gate transistors, enhanced channel strain and 0.171 μ m ² SRAM cell size in a 291Mb array,” 2008 IEEE International Electron Devices Meeting, 2008
1046	RESERVED – NOT USED
1047	Declaration of Nathaniel E Frank-White (December 10, 2022) (742000)
1048	Declaration of Nathaniel E Frank-White (December 12, 2022) (744136)
1049	PCT Publication WO2002/065700A2
1050	UK Patent Appl. Publ. No. GB 2,415,561
1051	UK Patent Appl. Publ. No. GB 2,390,506
1052	“Intel’s 32nm Update: The Follow-on to Core i7 and More,” Anandtech.com (Feb. 11, 2009) (from pages 21-28 of Declaration of Nathaniel E Frank-White, EX1047)
1053	“AMD Evergreen: 40nm DirectX 11 GPU Silicon Arrives,” Anandtech.com (June 3, 2009) (from pages 16-19 of Declaration of Nathaniel E Frank-White, EX1047)
1054	U.S. Patent Appl. Publ. No. 2007/0258641A1
1055	U.S. Provisional Application No. 61/218691
1056	RESERVED – NOT USED
1057	RESERVED – NOT USED
1058	U.S. Patent Appl. Publ. No. 2003/0033509A1
1059	U.S. Patent Appl. Publ. No. 2008/0288823A1
1060	RESERVED – NOT USED
1061	RESERVED – NOT USED
1062	RESERVED – NOT USED
1063	U.S. Patent Appl. Publ. No. 2009/0148068A1
1064	RESERVED – NOT USED
1065	RESERVED – NOT USED
1066	U.S. Patent Appl. Publ. No. 2009/0114952A1
1067	U.S. Patent Appl. Publ. No. 2006/0088958A1
1068	U.S. Patent Appl. Publ. No. 2008/0023700A1
1069	U.S. Patent Appl. Publ. No. 2009/0150112A1

Exhibit	Description
1070	U.S. Patent Appl. Publ. No. 2005/0237327A1
1071	RESERVED – NOT USED
1072	RESERVED – NOT USED
1073	RESERVED – NOT USED
1074	RESERVED – NOT USED
1075	U.S. Patent No. 5,226,142
1076	U.S. Patent No. 6,681,059
1077	U.S. Patent No. 7,243,165 (“Kravec”)
1078	U.S. Patent No. 7,072,872 (“Caid”)
1079	RESERVED – NOT USED
1080	U.S. Patent No. 6,963,891
1081	U.S. Patent No. 6,825,857
1082	U.S. Patent No. 6,353,867
1083	U.S. Patent No. 6,477,610
1084	U.S. Patent No. 5,581,773
1085	RESERVED – NOT USED
1086	RESERVED – NOT USED
1087	RESERVED – NOT USED
1088	VSI Alliance™: Virtual Component Interface Standard Version 2 (OCB 2 2.0) (April 2001)
1089	“Port,” in Microsoft Computer Dictionary, p. 412 (5th. ed. 2002)
1090	RESERVED – NOT USED
1091	RESERVED – NOT USED
1092	Aty et al., “High-Speed, Area-Efficient FPGA-Based Floating-Point Multiplier,” Proceedings of the 12 IEEE International Conference on Fuzzy Systems, December 11, 2003 (“Aty”) (from p. 14-17 of Declaration of Gordon MacPherson, EX1093)
1093	Declaration of Gordon MacPherson regarding Aty et al., High-Speed, Area-Efficient FPGA-Based Floating-Point Multiplier, Proceedings of the 12 IEEE International Conference on Fuzzy Systems, December 11, 2003
1094	Chen et al., “A cost effective 32nm high-K/ metal gate CMOS technology for low power applications with single-metal/gate-first process,” 2008 Symposium on VLSI Technology, 2008, pp. 67-68
1095	Prosecution History of PCT/GB2004004377
1096	U.S. Patent No. 4,450,402
1097	Schimpl et al., “A 0.13 /spl mu/m CMOS platform with Cu/low-k interconnects for system on chip applications,” 2001 Symposium on

Exhibit	Description
	VLSI Technology. Digest of Technical Papers (IEEE Cat. No.01 CH37184), 2001, pp. 101-102
1098	Sudha et al., “An Efficient Digital Architecture for Principal Component Neural Network and its FPGA Implementation,” IETE Journal of Research, Vol. 53, No. 5, September-October 2007, ISSN 0377-2063, pp. 425-431 (“Sudha”)
1099	RESERVED – NOT USED
1100	Oh et al., “A fully pipelined single-precision floating-point unit in the synergistic processor element of a CELL processor,” in IEEE Journal of Solid-State Circuits, vol. 41, no. 4, pp. 759-771, April 2006,
1101	RESERVED – NOT USED
1102	RESERVED – NOT USED
1103	Prabhu, “SOMGPU: An unsupervised pattern classifier on Graphical Processing Unit,” 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), 2008, pp. 1011-1018
1104	Poli et al., “Processing Neocognitron of Face Recognition on High Performance Environment Based on GPU with CUDA Architecture,” 2008 20th International Symposium on Computer Architecture and High Performance Computing, 2008, pp. 81-88
1105	Ligon et al., “A re-evaluation of the practicality of floating-point operations on FPGAs,” Proceedings. IEEE Symposium on FPGAs for Custom Computing Machines (Cat. No.98TB100251), 1998, pp. 206-215
1106	Held, “From a Few Cores to Many: A Tera-scale Computing Research Overview,” white paper, Intel, 2006 (from pages 67-78 of Declaration of Nathaniel E Frank-White, EX1047)
1107	Reddaway et al., “Ultra-high performance, low-power, data parallel radar implementations,” IEEE International Radar Conference, 2005., 2005, pp. 822-826
1108	Lokhmotov et al., “Revisiting SIMD Programming,” Languages and Compilers for Parallel Computing: 20th International Workshop, LCPC 2007, Urbana, IL, USA, October 11-13, 2007, Revised Selected Papers. Springer-Verlag, Berlin, Heidelberg, 32–46
1109	U.S. Patent No. 4,534,010
1110	Shahbahrami et al., “Optimization of Content-Based Image Retrieval Functions,” 2008 Tenth IEEE International Symposium on Multimedia, 2008, pp. 607-612

Exhibit	Description
1111	Noumsi et al., “Acceleration of a content-based image-retrieval application on the RDISK cluster,” Proceedings 20th IEEE International Parallel & Distributed Processing Symposium, 2006
1112	Proceedings 20th IEEE International Parallel & Distributed Processing Symposium, 2006 (TOC)
1113	Pengdong et al., “Performance Comparison between Color and Spatial Segmentation for Image Retrieval and Its Parallel System Implementation,” 2008 International Symposium on Computer Science and Computational Technology, 2008, pp. 539-543
1114	Bradley et al., “Exploiting Loop-Level Parallelism for SIMD Arrays Using OpenMP,” 89-100 (2007) (attachment 13c to Declaration of Dr. Mary K. Bolin, EX1005)
1115	Haran et al., “22 nm technology compatible fully functional 0.1 μm^2 6T-SRAM cell,” 2008 IEEE International Electron Devices Meeting, 2008, pp. 1-4
1116	Tong et al., Sean & Hahn, Scott. (2008). Operating System Support for Shared-ISA Asymmetric Multi-core Architectures.
1117	Borkar, “Thousand Core Chips A Technology Perspective,” Proceedings of the 44th ACM/IEEE Design Automation Conference, pp. 746-749 (2007) (attachment 11b to Declaration of Dr. Mary K. Bolin, EX1005)
1118	“ClearSpeed kicks it up a notch,” Arstechnica.com (Oct. 11, 2004) (from pages 5-6 of Declaration of Nathaniel E Frank-White, EX1123)
1119	U.S. Patent No. 7,447,873
1120	Manocha, “General-purpose computations using graphics processors,” in Computer, vol. 38, no. 8, pp. 85-88, Aug. 2005
1121	RESERVED – NOT USED
1122	U.S. Patent Appl. Publ. No. 2009/0183167A1
1123	Declaration of Nathaniel E Frank-White (December 12, 2022)
1124	Burger et al., “Composable Multicore Chips,” in Multicore Processors and Systems. Integrated Circuits and Systems (Springer 2009)
1125	Roma et al., “In the development and evaluation of specialized processors for computing high-order 2-D image moments in real-time,” in Proceedings Fifth IEEE International Workshop on Computer Architectures for Machine Perception, pp. 170-179 (2000)
1126	“Intel Developer Forum 22nm News Facts,” Intel Corporation (Sept. 22, 2009) (from page 80 of Declaration of Nathaniel E Frank-White, EX1047)
1127	Declaration of Gordon MacPherson regarding EX1013

Exhibit	Description
1128	Declaration of Gordon MacPherson regarding EX1039
1129	Declaration of Gordon MacPherson regarding EX1045
1130	Declaration of Gordon MacPherson regarding EX1094
1131	Declaration of Gordon MacPherson regarding EX1097
1132	RESERVED – NOT USED
1133	Declaration of Gordon MacPherson regarding EX1100
1134	Declaration of Gordon MacPherson regarding EX1110
1135	Declaration of Gordon MacPherson regarding EX1115
1136	Declaration of Gordon MacPherson regarding EX1125
1137	TSMC Press Release, “TSMC Ramps 40nm Volume Production to Promote Innovation as Foundries Assume a Larger Role for \$300 Billion Industry” (Nov. 17, 2008)
1138	Thornton et al., “Tradeoff analysis of integer multiplier circuits implemented in FPGAs,” in 1999 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM 1999). Conference Proceedings (Cat. No.99CH36368), 1999, pp. 301-304
1139	Amended Complaint for Patent Infringement, <i>Singular Computing LLC v. Google LLC</i> , 1:21-cv-12110 (D. Mass. Mar. 10, 2022)

I. MANDATORY NOTICES

A. Real Party-In-Interest

Petitioner is the Real Party-in-Interest.

B. Related Matters

1. United States Patent & Trademark Office

The application from which U.S. Patent No. 11,169,775 (“the ’775 patent”) issued is a Continuation of U.S. Patent Application No. 16/571,871 (U.S. Patent No. 10,664,236), which is a Continuation of U.S. Patent Application No. 16/175,131 (U.S. Patent No. 10,416,961), which is a Continuation of U.S. Patent Application No. 15/784,359 (U.S. Patent No. 10,120,648), which is a Continuation of U.S. Patent Application No. 14/976,852 (U.S. Patent No. 9,792,088), which is a Continuation of U.S. Patent Application No. 13/849,606 (U.S. Patent No. 9,218,156), which is a Continuation of U.S. Patent Application No. 13/399,884 (U.S. Patent No. 8,407,273), which is a Continuation of U.S. Patent Application No. 12/816,201 (U.S. Patent No. 8,150,902), which claims priority to U.S. Provisional Application No. 61/218,691.

The following U.S. patent applications also claim the benefit of the priority of the filing date of U.S. Provisional Application No. 61/218,691:
16/675,693; 16/882,686; 17/029,780; 17/367,051; 17/367,071; 17/367,097.

2. United States Patent Trial and Appeal Board

Concurrently with the present Petition, Petitioner is filing a second petition for *inter partes* review of the '775 patent. Petitioner is also filing petitions for *inter partes* review of related U.S. Patent No. 10,754,616.

3. U.S. District Court for the District of Massachusetts

Patent Owner asserted the '775 patent in *Singular Computing LLC v. Google LLC*, Case No. 1:21-cv-12110 (D. Mass.). Patent Owner has represented to Petitioner that it will soon voluntarily dismiss that case without prejudice.

C. Counsel and Service Information - § 42.8(b)(3) and (4)

Lead Counsel	Elisabeth H. Hunt, Reg. No. 67,336
Backup Counsel	Marc S. Johannes, Reg. No. 64,978 Gregory S. Nieberg, Reg. No. 57,063
Service Information	E-mail: EHunt-PTAB@wolfgreenfield.com MJohannes-PTAB@wolfgreenfield.com GNieberg-PTAB@wolfgreenfield.com <u>Post and hand delivery:</u> Wolf, Greenfield & Sacks, P.C. 600 Atlantic Avenue Boston, MA 02210-2206 <u>Telephone:</u> 617-646-8000 <u>Facsimile:</u> 617-646-8646

A power of attorney is submitted with the Petition. Counsel for Petitioner consents to service of all documents via electronic mail.

II. GROUNDS AND STANDING

Petitioner requests *inter partes* review and cancellation of claims 1-5, 7-14, and 16-23 (the “challenged claims”) of the ’775 patent (EX1001).

The challenged claims would have been obvious under §103 per the following ground. Each reference below is prior art under pre-AIA §102(b).

Ground Number and Reference(s)		Claims
1	Stuttard (EX1006), Shirazi (EX1036)	1-5, 7-14, 16-23

Petitioner certifies that the ’775 patent is available for *inter partes* review and that Petitioner is not barred or estopped from requesting *inter partes* review as to the challenged claims. 37 C.F.R. §42.104(a).

III. ’775 PATENT

The ’775 patent descends from a long line of continuation applications sharing a specification that states, “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).” EX1001, 5:58:62. Patent Owner (“Singular”) sued Petitioner (“Google”) on three of those continuations in 2019; Google filed IPRs; and the Board invalidated those patents’ claims to a device comprising at least one LPHDR execution unit. IPR2021-00155; IPR2021-00165; IPR2021-00179. After Singular had already sued Google, Singular filed the ’775 patent’s application,

cobbling together disparate, cherry-picked claim limitations to support a second concurrent lawsuit against Google. But the '775 patent's claims still recite no more than obvious variations of what had long been known in the art.

A. Specification

The '775 patent's specification has one isolated statement that “*if* [some LPHDR] 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. However, the circuits to multiply and divide these floating point values would be relatively large.”

EX1001, 5:63-6:3.¹ Having thus criticized this digital floating-point option, the specification then leaves it behind and spends the rest of its 30 columns describing logarithmic and analog embodiments. EX1003 (“Franzon”), ¶¶38-43.

Yet, because Google (and much of the world) uses floating-point arithmetic, Singular's litigation-inspired '775 patent claims now recite multiplier circuits using floating-point values of particular bitwidths. These limitations are claimed together with phrases from distant and unrelated portions of the specification, such as a processing element array on a chip (cols. 8-10) with 5000 processing elements

¹ All emphases herein are added unless otherwise indicated.

(col. 28). None of these cherry-picked add-ons make Singular’s purported invention remotely patentable.

B. Claims

Challenged independent claims 1, 7 and 16 each recite a decades-old processing element (PE) array architecture that was well-known for facilitating parallel computations. Franzon, ¶¶50-52. Further limitations recite conventional computer system components including a “*computing chip*” for the PE array, a “*host computer*,” an “*input-output unit*,” and a “*host connection*.”

The recited PEs include respective multiplier circuits adapted to receive floating-point values having a “*binary mantissa of width no more than 11 bits*” and a “*binary exponent of width at least 6 bits*.” Such floating-point formats—in which a binary number can be represented as a fractional “mantissa” (greater than 1 and less than 2) scaled by a power of two (the power being the “exponent”)—were decades old. Franzon, ¶51. And optimizing the bitwidth of floating-point multiplier circuits to reduce size and power consumption was well-known before Singular’s claimed priority date. Franzon, ¶¶53-62.

Claims 7 and 16 recite that there are “*no less than 5000*” PEs in the array. Franzon, ¶¶63-69. Claim 16 further recites that a circuit board, connected to a host computer, comprises a plurality of such PE arrays. Each of the challenged claims would have been obvious over the prior art, as the grounds herein demonstrate.

C. Person of Ordinary Skill in the Art (“POSA”)

A POSA would have had a bachelor’s degree in Electrical Engineering, Computer Engineering, Applied Mathematics, Applied Mathematics or the equivalent, and two years of academic or industry experience in computer architecture. More education could substitute for experience, and vice versa. Franzon, ¶¶33-37.

D. Prosecution History

The examiner allowed the challenged claims, and the claims in all other applications in the priority chain, without substantively discussing any prior art. EX1002, 14.

E. Priority Dates

The ’775 patent claims priority to a 2009 provisional application that lacks written description support at least for claims 7-24 which recite “no less than 5000” processing elements.

The 2009 provisional application disclosed only that in “certain embodiments” the number of “LPHDR arithmetic elements...significantly exceeds (e.g., by at least 20 more than three times) the number of arithmetic elements” of “traditional precision,” which discloses neither a number of LPHDR elements, nor what number of traditional-precision elements the number of LPHDR elements “significantly exceeds.” EX1055, 8.

The provisional application's disclosure does not remotely "constitute a full, clear, concise and exact description" of an array of processing elements numbering "no less than 5000," which would be required to satisfy the written description requirement under pre-AIA 35 U.S.C. §112, ¶1. *In re Wertheim*, 646 F.2d 527, 538-39 (C.C.P.A. 1981); *Purdue Pharma L.P. v. Faulding Inc.*, 230 F.3d 1320, 1323 (Fed. Cir. 2000) ("[A POSA] reading the original disclosure, must immediately discern the limitation at issue in the claims.").

Although it would have been obvious to integrate as many processing elements as semiconductor processing permitted (as discussed §VI.A.1 below), obviousness does not satisfy the written description requirement. *Lockwood v. Am. Airlines*, 107 F.3d 1565, 1571-72 (Fed. Cir. 1997) ("Entitlement to a filing date does not extend to subject matter which is not disclosed, but would be obvious over what is expressly disclosed."); *In re Blaser*, 556 F.2d 534, 538 (C.C.P.A. 1977) ("[E]nablement and obviousness are not the issues; description of the invention is.").

Thus, independent claims 7 and 16 and their respective dependent claims are not entitled to the provisional application's 2009 filing date. The next-earliest application to which the '775 patent claims priority was filed June 15, 2010.

IV. CLAIM INTERPRETATION

Claim terms are construed herein using the standard used in civil actions under 35 U.S.C. §282(b), and have been given their ordinary and customary meaning as understood by a POSA in accordance with the specification and prosecution history. 37 C.F.R. §42.100(b).

V. GROUND 1A: CLAIMS 1-5 ARE UNPATENTABLE OVER STUTTARD IN COMBINATION WITH SHIRAZI

A. The Stuttard-Shirazi Processor

1. Stuttard (EX1006)

Stuttard discloses a “Multi-Threaded Array Processing (MTAP) architecture” with “parallel data processing capability” for “high performance, high data rate processing.” Stuttard, 8:23-9:14.

As shown in Figure 7 (annotated below), each MTAP “Processor Core” is a “data processor compris[ing] a single control unit” (purple) and two execution units: a SIMD “poly” execution unit (blue) and a scalar “mono” execution unit (red). Stuttard, 2:29-34. Franzon, ¶¶70-75.

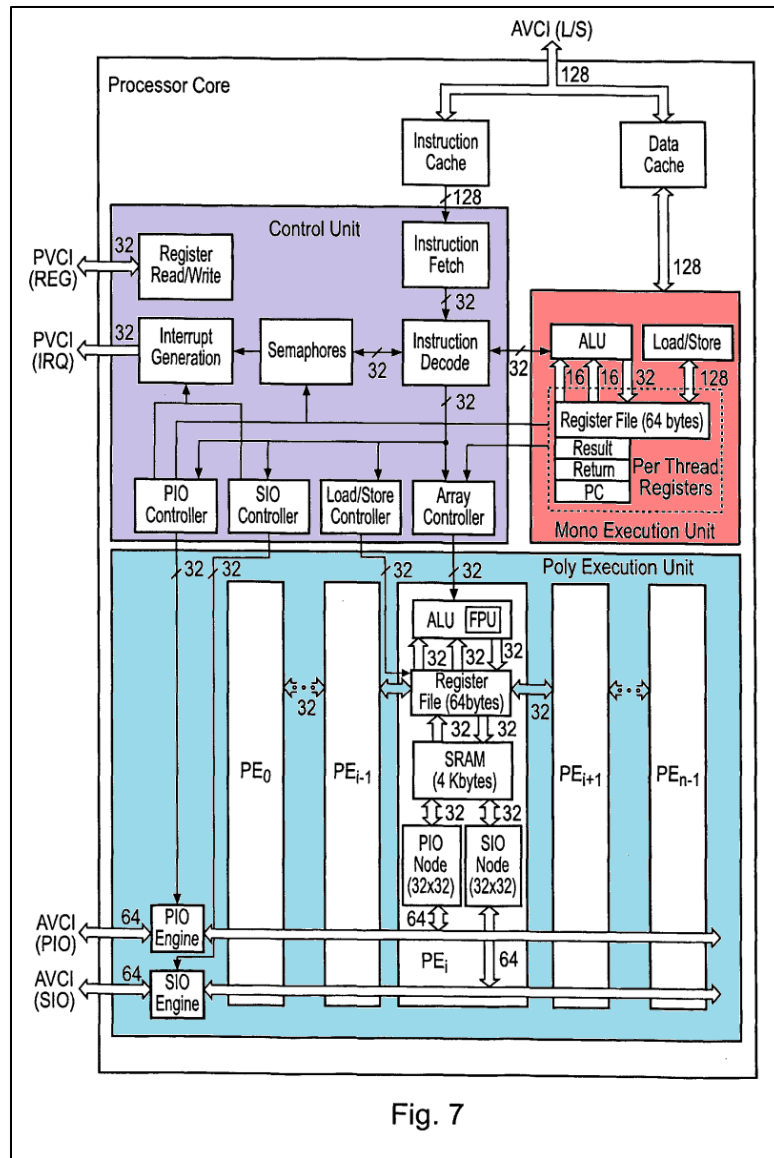


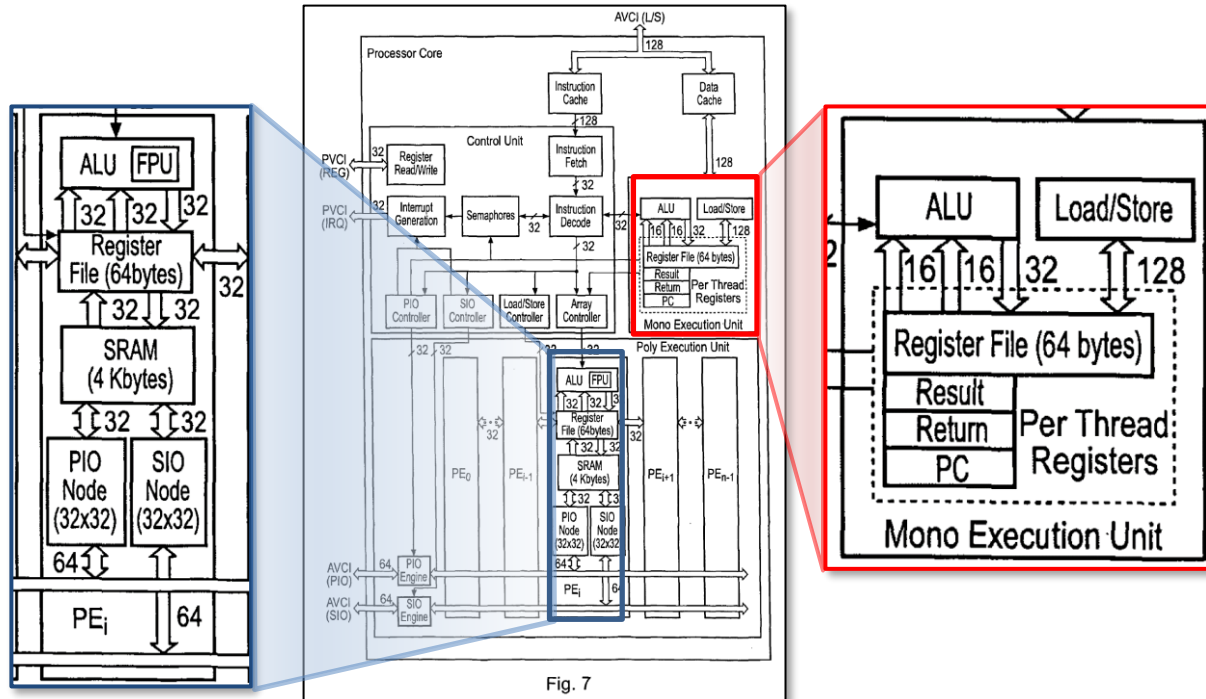
Fig. 7

The poly unit comprises an array of processing elements (PEs) and “gets its performance largely from the number of PEs brought to bear on the problem rather than the processing power of the individual PEs.” Stuttard, 7:19-21. Thus, the number of PEs in the poly unit is designed “to be scaled to meet the needs of the application.” Stuttard, 10:5-6. For example, the poly unit “may consist of tens,

hundreds or even thousands of PEs.” Stuttard, 10:12-14, 12:15-16 (“[T]he [MTAP] core can be scaled...to 256 or even thousands of PEs”). Franzon, ¶76.

The mono unit is for handling “scalar and non-parallel” data (Stuttard, 10:8-10), and has its own circuitry “for accelerating software implementations of floating point operations” (Stuttard, 16:13-20). Franzon, ¶77.

Each poly-unit PE (annotated blue in Figure 4 below) includes an arithmetic logic unit (ALU) with a floating-point unit (FPU) and local memory (e.g., SRAM). Stuttard, 7:14-24, 12:7-11, 13:21-14:4. The mono unit (annotated red below) likewise includes an ALU capable of “floating point operations.” Stuttard, 16:13-20. Franzon, ¶¶78-79.



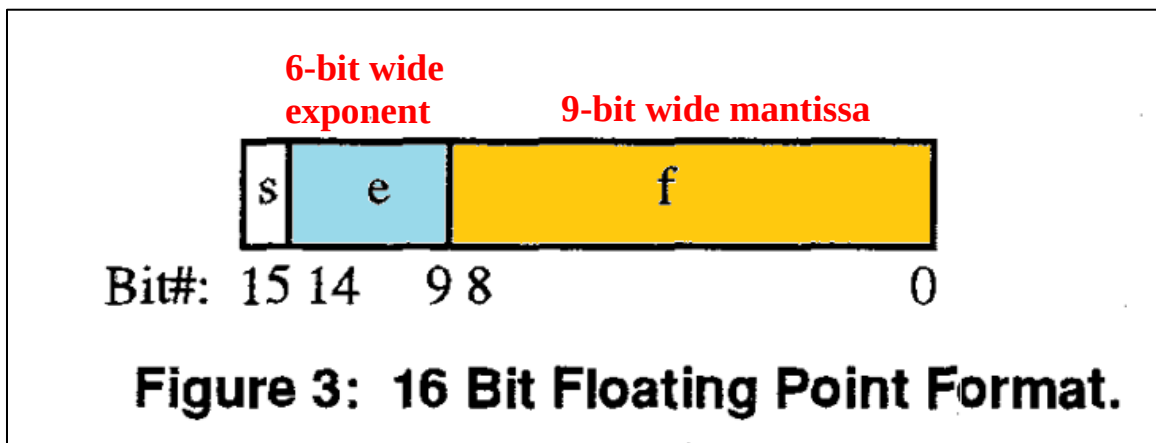
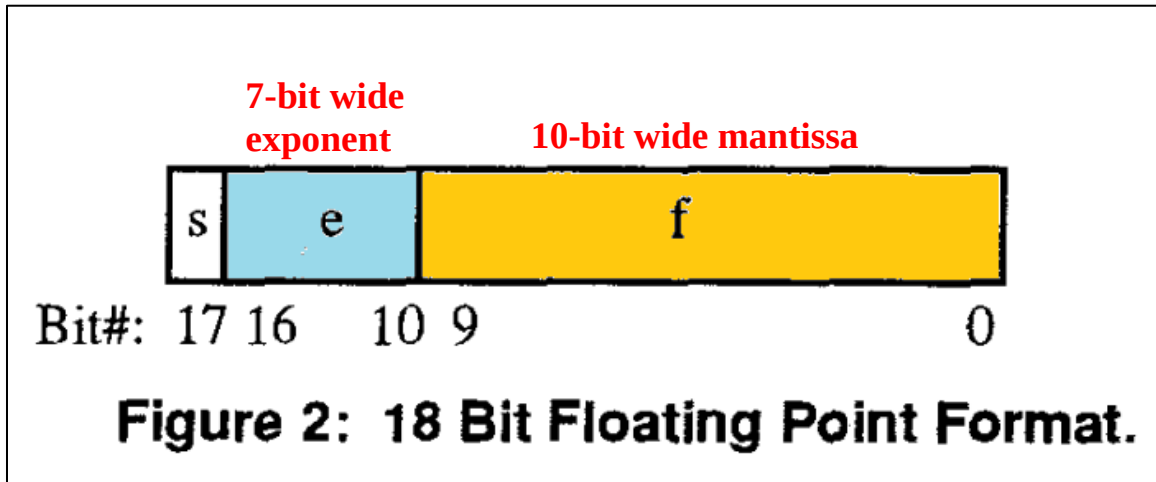
Stuttard teaches that the “mono execution unit is likely to require a wider and more complex ALU” than the poly unit’s ALUs. Stuttard, 7:17-24.

2. Shirazi (EX1036)

Shirazi is §102(b) prior art because it was publicly accessible in 1995. *See infra* §VII.A.

Shirazi teaches that “[m]any algorithms rely on floating point arithmetic for the[ir] dynamic range” and “require millions of calculations per second.” Shirazi, 155. For example, 2-D FFTs (Fast Fourier Transforms) and FIR (finite-impulse-response) filtering computations are commonly used in “[i]mage and digital signal processing applications [that] typically require high calculation throughput.” *Id.* These applications are, therefore, “candidates for acceleration using custom computing machines (CCMs) [] tailored for the application” (*id.*)—i.e., the type of machine Stuttard describes. Franzon, ¶80. Circuitry to operate on IEEE-standard full-precision 32-bit floating-point values (having 8-bit exponents and 23-bit mantissas), however, required “excessive area” on a chip compared to circuitry adapted to “custom formats, derived for individual applications.” Shirazi, 155. Shirazi’s two custom formats are illustrated below (with annotations): (1) an 18-bit format developed for “2-D FFT application,” having one sign bit, 7 exponent bits, and 10 mantissa bits; and (2) a 16-bit format developed for “the FIR filter

application,” having one sign bit, 6 exponent bits, and 9 mantissa bits. Franzon, ¶80.



Shirazi, 156.

These custom formats provided sufficient “dynamic range” (via their exponent bitwidths) for applications like 2D-FFT and FIR, respectively, while also “minimiz[ing] area” on the chip (via their reduced-bitwidth mantissas) compared to circuitry for higher-precision values. Shirazi, 156, 161; Franzon, ¶81.

3. Stuttard-Shirazi Combination

a. Reduced-Bitwidth Poly PEs

POSAs would have had reasons to adapt the PEs in Stuttard's poly unit to perform floating-point math using Shirazi's custom reduced-bitwidth floating-point values. Franzon, ¶82.

As discussed above (§V.A.2), Shirazi taught that its floating-point formats having reduced-bitwidth mantissas provided sufficient precision for common high-data-rate applications, while also allowing the arithmetic circuitry to be implemented with reduced chip “area” compared to full-precision circuits. Shirazi, 155-56. Shirazi was not alone, as this was well-known background knowledge for POSAs. Franzon, ¶¶53-62, 83-84; EX1054. For example, EX1034 (“Tong”) similarly taught that “[i]t has **long been known** that many...applications can get by **with less precision**” (Tong, 273), and thus the IEEE's single-precision (32-bit) format often “is not essential” (Tong, 279). Tong also corroborates that POSAs knew that minimizing the number of mantissa bits reduces each PE's power consumption because a “reduced precision FP unit” has “small size” compared to higher-precision units. Tong, 282. Franzon, ¶83 (citing as further corroborating evidence EX1098, 428 & 425 (using “an 8-bit mantissa and an 8-bit exponent” in “Principal Component Neural Network (PCNN)” applications); EX1092, 274 (“High-Speed, Area-Efficient...Floating-Point Multiplier” “for real-time signal

processing”), 276 Tables 2-4 (16-bit format with 9-bit mantissa, 6-bit exponent); EX1105, 207 (“us[ing] [] reduced precision floating-point formats to conserve chip area” was suitable “[f]or a broad range of applications” and “retain[ed] the benefit of a large dynamic range”); EX1125, 170-174 (for pattern-recognition applications, “[e]xperimental results show...th[e following] floating-point precision...seems to be enough”: “8 bits for the exponent (*e*) and 10 bits used for the fractional part (*f*) of the mantissa”)).

Thus, Shirazi taught (and Tong and others corroborated) 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. Franzon, ¶¶84-90 (citing corroborating Tong, 280, 278-79 (signal processing applications had no “noticeable degradation in accuracy” when using 11 (or fewer) mantissa bits, and six (or more) exponent bits “preserve[d] a reasonable degree of accuracy”); EX1106, 6 (“[P]rocessing elements dedicated to a particular workload...will perform better for that task than a general-purpose execution unit.”)).

As discussed above (§V.A.1), Stuttard’s MTAP processor architecture is for “high performance, high data rate processing” applications. Stuttard, 8:24-28.

POSAs would have understood this architecture to be well-suited for signal-processing applications like the 2-D FFTs and FIR filtering computations that Shirazi taught were compatible with reduced-precision floating-point formats. Shirazi, 155-56. Franzon, ¶¶91-92 (citing EX1080, 1:55-2:9 (“[t]he FFT is widely used in real time digital signal processing requiring fast execution” and it is “important that the FFT be able to make effective use of these [SIMD] kinds of architectures”); EX1107, 823-25 (describing FFT computations using MTAP-based processors); EX1108, 32-34, 44 (paper co-authored by two of Stuttard’s inventors, corroborating that it was known to program MTAP architectures for “2D and 3D FFTs”)).

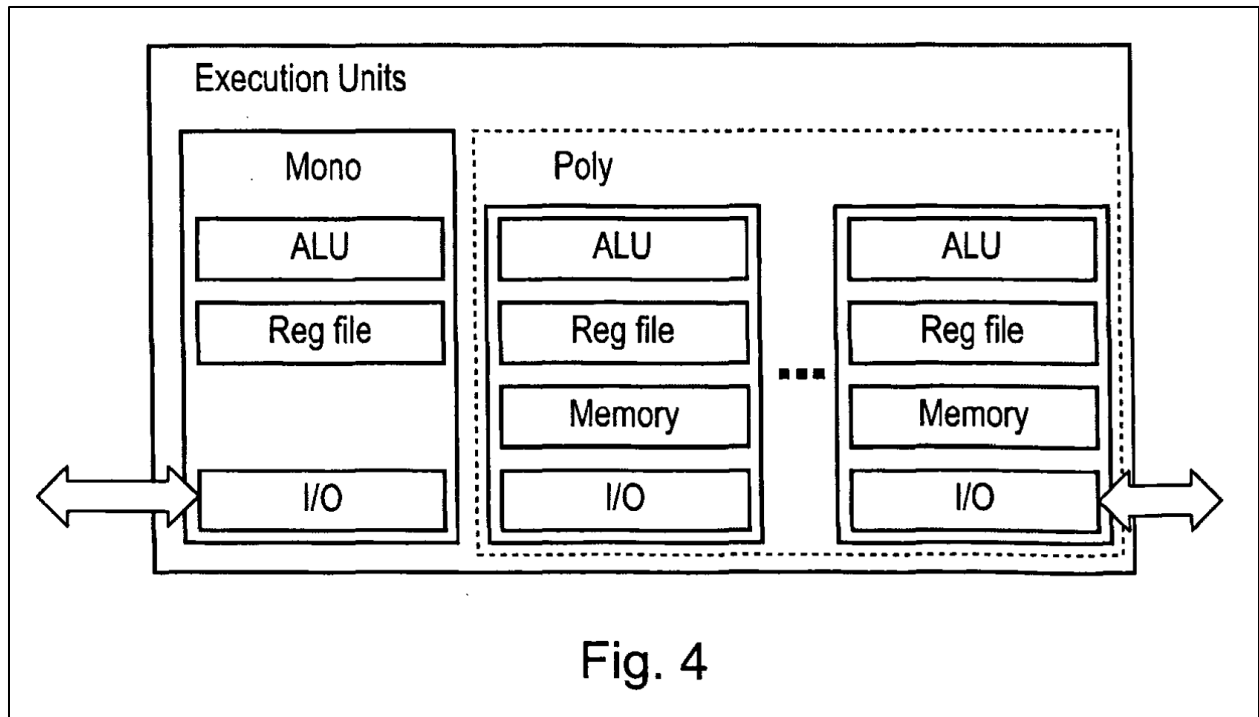
Stuttard’s poly unit “gets its performance largely from the number of PEs brought to bear on the problem rather the processing power of the individual PEs,” which “need to be efficiently implemented.” Stuttard, 7:19-24. This would have further motivated POSAs to pursue techniques like Shirazi’s reduced-bitwidth floating-point formats, which would have allowed for more PEs on the chip. Shirazi, 155-56. Franzon, ¶93.

Thus, POSAs had reasons to adapt the floating-point units in Stuttard’s poly unit to have circuitry that accepts as inputs, and operates on, floating-point values having bitwidths no greater than one of Shirazi’s reduced-precision floating-point formats. Franzon, ¶94.

POSAs would have reasonably expected success in this Stuttard-Shirazi combination because Shirazi’s “custom formats...are feasible on CCMs” (custom computing machines). Shirazi, 155-56. POSAs would have understood that Stuttard’s MTAP architecture is such a custom computing machine: Stuttard teaches that the poly and mono PEs need not have identical “micro-architecture”—e.g., “the poly processing elements could have an 8-bit ALU, while the mono execution unit has a 32-bit ALU.” Stuttard, 3:15-18, 9:13-14, 24:28-29 (describing simulation software for “the core” allowing the “number of PEs, size of memory, and other options” to be “parametrized”). Thus, POSAs would have reasonably expected success in using only ordinary skill to customize the floating-point units in Stuttard’s poly unit for the target applications Shirazi and Stuttard both describe (e.g., FFT computations), by adapting these PEs to receive as inputs floating-point values in one of Shirazi’s reduced-precision formats and no wider. Franzon, ¶95.

b. Two-Dimensional PE Array

Stuttard’s Figure 4 illustrates one “Mono” execution unit and multiple PEs in the “Poly” execution unit. Stuttard, 10:12-19, 12:15-17.



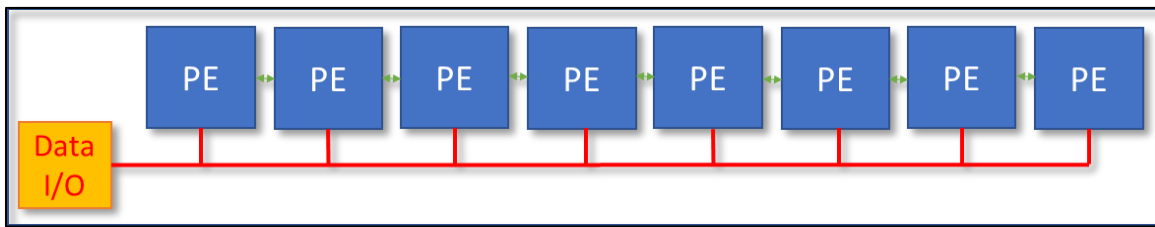
Stuttard's Figures 4 and 7 illustrate poly PEs side-by-side because each PE communicates with its left and right neighbors in what Stuttard calls a "swazzle" path. Stuttard, 20:31-21:2; Franzon, ¶¶96-97. "This is a conceptual view, which reflects the programming model" for an MTAP processor, and POSAs would have understood that it does not limit the physical layout of the processor on a chip. Stuttard, 13:2-5; Franzon, ¶¶97. For example, the mono unit is arbitrarily shown beside the poly unit in Figure 4 but above the poly unit in Figure 7. POSAs would have understood that the physical layout was a matter of design choice and, as discussed below, had reasons to manufacture a physical chip using a two-

dimensional PE layout. Franzon, ¶97 (citing as corroboration EX1081, 8:46-47, 21:5-20, discussed below).

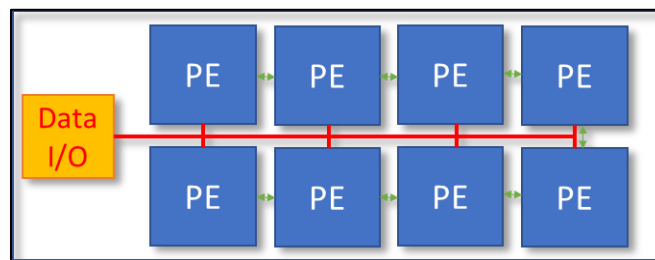
First, POSAs knew that a two-dimensional layout reduces the overall length of wiring in the chip, such as Stuttard’s “channels” that transfer data between PEs and the host. Stuttard, 17:20-23, 23:26-32; §V.B.6 below; Franzon, ¶98. These data channels are connected to every PE in the device. Stuttard, 17:20-23, 23:26-32, FIG. 7. POSAs would have understood that physically arranging PEs in a two-dimensional array reduces the overall length of each channel, beneficially reducing (i) signal delays, (ii) resistances (and thus heat) and (iii) undesirable “parasitic” capacitances. Franzon, ¶98 (citing as corroboration EX1066, [0007]).

To illustrate this, the first demonstrative below shows 8 PEs arranged in a one-dimensional line; the second in a two-dimensional array. The red line in each figure represents Stuttard’s channel connected to each PE; the green lines represent Stuttard’s “swazzle” connections. As demonstrated below, both layouts allow for a swazzle path connecting each PE to its linearly adjacent neighbors, but the channel’s length in the 2D array can beneficially be approximately half as long as in the 1D line. Franzon, ¶99.

Linear Layout



Two-dimensional Layout



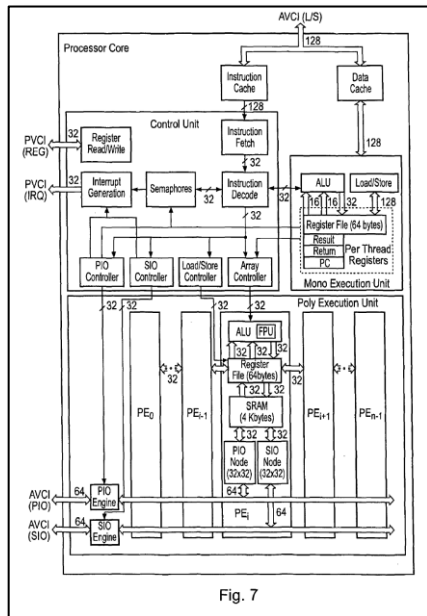
Second, POSAs knew it was desirable and conventional for computing chips to have aspect ratios (width-to-length ratios) closer to 1, because very high or very low aspect ratios cause “mechanical stress[es]” and “adversely change the electrical properties of the IC circuitry.” EX1067, [0007]. Franzon, ¶100.

Third, other related prior art further corroborates that a two-dimensional interconnect topology (and corresponding two-dimensional physical topology) was a conventional alternative to a linear topology. For example, EX1108—co-authored by two of Stuttard’s inventors—described “a simple extension to the C [programming] language” for use in “[m]assively parallel SIMD array architectures” including “MTAP” architectures like Stuttard’s, and corroborates the POSA’s knowledge that “as the number of PE cores increases *other (than linear)* interconnect topologies could be introduced.” EX1108, 35; Franzon, ¶101.

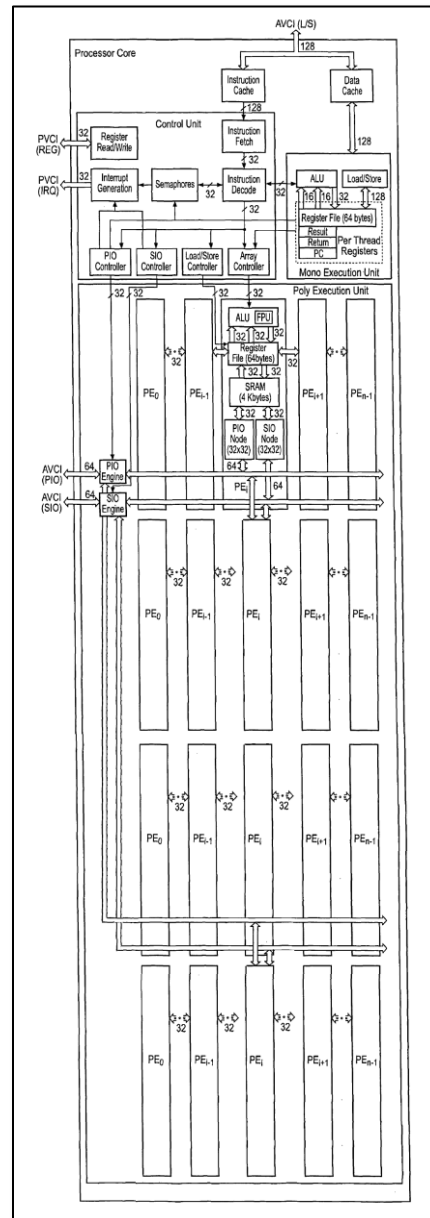
Similarly, EX1081—assigned to Stuttard’s Applicant “ClearSpeed Technology”—teaches a SIMD device similar to Stuttard’s SIMD Processor for use with image processing having “a processing element (PE) array” with “PE interconnections” that move data “along the PE array (hereinafter swazzled),” and explains the array’s “PE interconnection” can be arranged in “a 1D ring/line array **or a 2D grid array.**” EX1081, 5:51-65, 8:40-47 (“Two interconnection networks are generally used: a 1D array/ring and **a 2D grid.**”). EX1081 further taught that “**a 2D array** with a two dimensional bi-directional (up, down, left, right) nearest-neighbor configuration, often known as a mesh” was a “common configuration” and sometimes “preferable to exploit the 2D connectivity.” EX1081, 21:10-22. Thus, using a 2D-array in Stuttard-Shirazi would have been using a known technique to improve similar devices (parallel processor having multiple PEs) in the same way (by facilitating applications that benefited from full-2D connectivity between PEs). *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 (2007). Franzon, ¶102.

Applying the above-discussed principles to Stuttard-Shirazi would have resulted in a two-dimensional layout of poly PEs, an exemplary design of which is depicted below (right, adapted from Stuttard's Figure 7, left). Franzon, ¶103.

Stuttard, FIG. 7



Stuttard-Shirazi



B. Claim 1

1. [1PRE]: “A computing system, comprising:”

The Stuttard-Shirazi processor (§V.A.3 above) is for “use[] in a number of *computing* applications” (Stuttard, 1:3-5) by “integration” into “a variety of *systems*” (Stuttard, 9:15-16). Data and instructions are provided by a host computer. Stuttard, 1:21-24; §V.B.2 below. Thus, the Stuttard-Shirazi processor and host computer are integrated in a computing system, as claimed. Stuttard, 5:12-14, 13:21-25. Franzon, ¶104.

2. [1A]: “a host computer;”

The Stuttard-Shirazi processor communicates with a “host system” including a “host processor” that runs an “application or operating system.” Stuttard, 21:13-31, 1:11-2:3, 13:19-20, 20:5-8 (describing interface for “transfer[ring] data to and from” “external memory...or a host processor”), 22:4-17; Franzon, ¶105 (citing corroborating EX1089, 2-3). Stuttard’s “host” is a host computer, as claimed. Stuttard, 1:11-24 (in “SIMD processor architectures” like Stuttard’s, the “*host processor*...will send data to be processed to the SIMD array with an indication of the function to be performed”); EX1001, 3:18-20 (“computer[]” can be a “typical microprocessor”), 8:10-9:16 (host “is responsible for overall control”), 29:27-36 (“computer includ[es] a processor and other components (such as memory coupled to the processor by a data path)”); Franzon, ¶106.

3. [1B]: “a computing chip comprising:”

The Stuttard-Shirazi processor would have been fabricated as a computing chip, following Stuttard’s teachings that “illustrate the use of an MTAP processor in a real device” manufactured as a “**chip**.” Stuttard, 22:3-7, 7:3-4, 22:12-19 (describing chip’s “bus” connecting “from one **chip** to another” and “on-**chip** SRAM”); Franzon, ¶107.

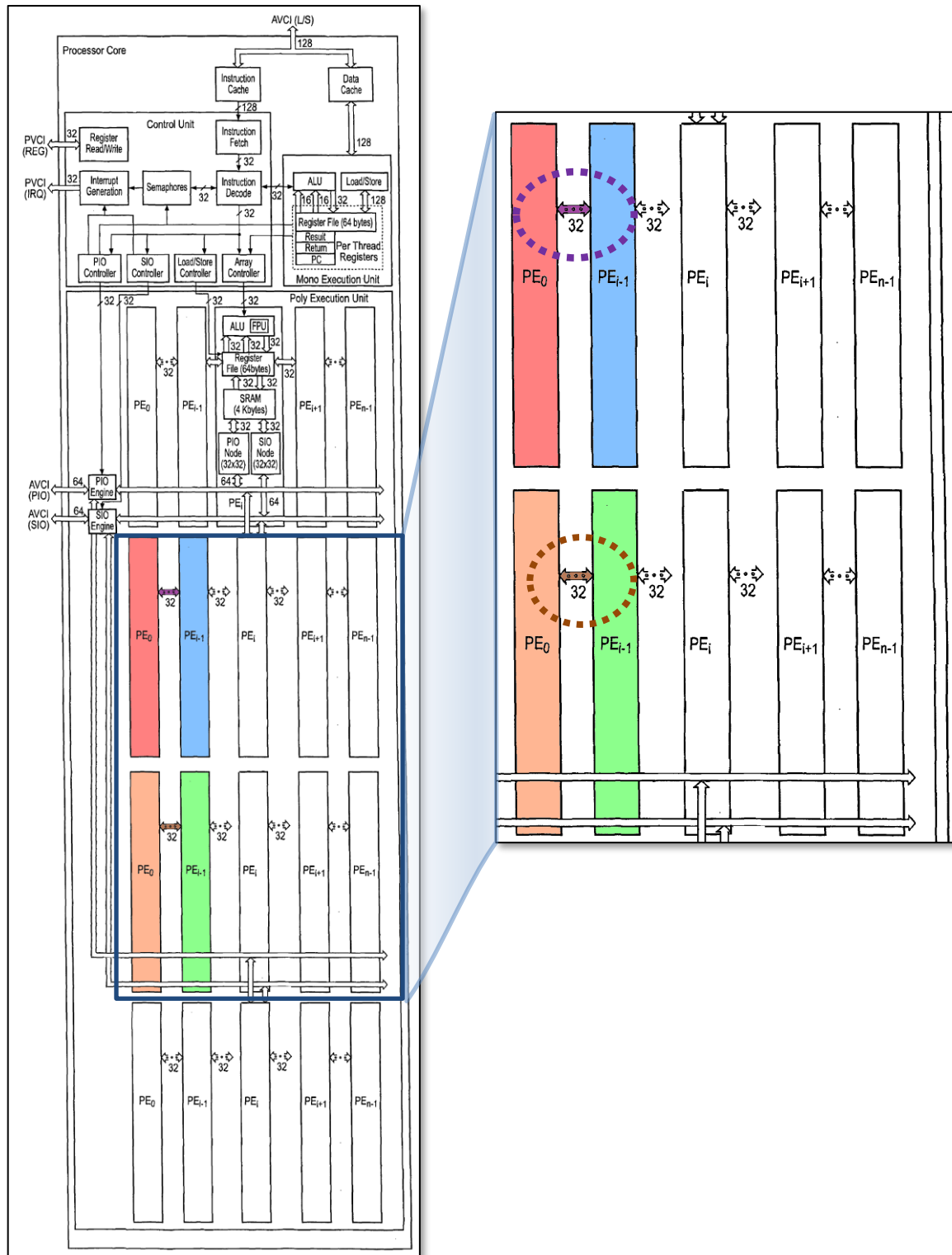
4. [1C]: “a processing element array comprising” “first” and “second” “edge processing element[s] positioned at a first edge of the processing element array” and “first” and “second” “interior processing element[s] positioned at [first and second] location[s] in the interior of the processing element array”

Stuttard’s “processing elements” (PEs) are in a SIMD “array.” Stuttard, 3:7-9, 3:30-34, 5:29-30, 10:12-19, claim 6; Franzon, ¶108; *see* §V.B.5 below. As discussed above (§V.A.3.b above) and illustrated below right, the Stuttard-Shirazi processor’s PE array is two-dimensional:

(green) at a second location in the interior of the PE array, as claimed. Franzon, ¶109.

5. [1D]: “a first processing element connection connecting the first edge processing element with the first interior processing element; a second processing element connection connecting the second edge processing element with the second interior processing element;”

Stuttard’s PEs in the Stuttard-Shirazi processor “communicate with one another via what is known as the *swazzle* path that **connects** the register file of **each PE with** the register files of **its left and right neighbours.**” Stuttard, 20:31-33. These swazzle connections include a first PE connection (circled purple below) connecting the first edge PE (red) with the first interior PE (blue) and a second PE connection (circled brown) connecting the second edge PE (orange) with the second interior PE (green), as claimed. Franzon, ¶110.



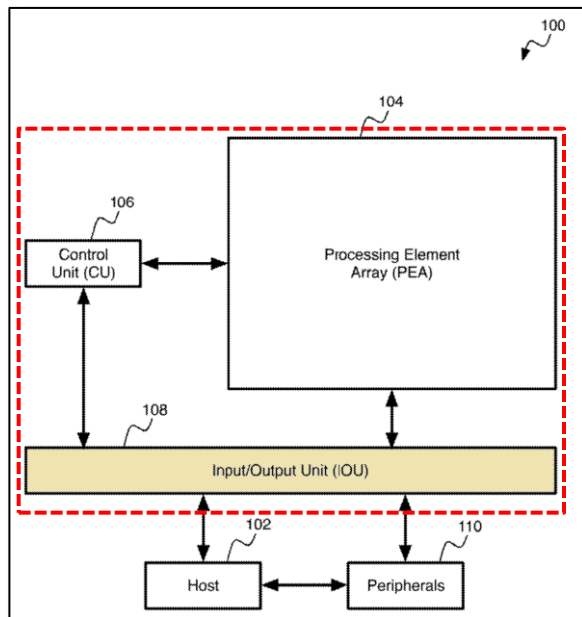
6. [1E]: “an input-output unit connected to the first” and “second edge processing element[s];”

The '775 patent's specification mentions an “I/O unit (IOU) 108” but does not limit it to being composed of any particular components. EX1001, 8:12-13. The specification merely states that “I/O could be routed...through the separate I/O unit,” which “interfac[es] the CU [control unit] 106 and PEA [processing element array] 104 with the Host 102” to “get data into and out of the CU 106 and PEA 104.” EX1001, 9:5-11, 9:24-28. Franzon, ¶111.

The '775 patent's Figure 1 (annotated below left) illustrates the IOU connecting an on-chip Control Unit (CU) and Processing Element Array (PEA) to off-chip Host and Peripherals. EX1001, 8:10-14. Stuttard's similar Figure 8 (annotated below right), illustrates “a block diagram...of the MTAP architecture as applied to an evaluation chip” (Stuttard, 7:3-4) which includes an on-chip bus (“ClearConnect Bus”) with “two external ports at each end” that, like the '775 patent's IOU, connect the chip to “peripherals...or a host interface.” Stuttard, 22:3-17.² Franzon, ¶112.

² Stuttard at 22:9 references “Figure 9,” but Stuttard has only eight figures (during prosecution, other “Figure 9” references were corrected to “Figure 8”), and POSAs would recognize the “Figure 9” is a typographical error meant to reference Figure

'775 Patent, FIG. 1



Stuttard, FIG. 8

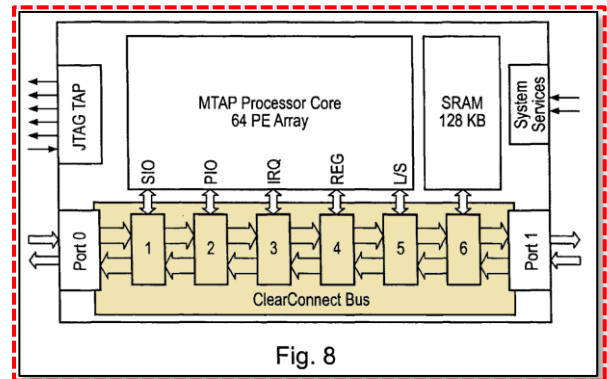


Figure 8's MTAP Processor Core contains a control unit and a PE Array (composed of Figure 7's PEs). Thus, Stuttard teaches that in a real-world chip the bus connects a control unit and PEs to off-chip peripherals and a host system, like the '775 patent's IOU. Franzon, ¶114.

The exemplary bus in Stuttard's Figure 8 (above right) includes nodes 1-5 that connect to "five ports of the MTAP processor [core]." Stuttard, 22:9-12. In

8, as the "evaluation chip EVI" description at 22:3-19 corresponds with the illustrations in Figure 8 (illustrating an "evaluation chip"—Stuttard, 7:3-4). Franzon, ¶113. *In re Yale*, 434 F.2d 666, 668-69 (C.C.P.A. 1970) ("Since it is an obvious error," POSAs would "mentally disregard [it] as a misprint or mentally substitute [the correct word] in its place.").

Figure 7 (above right), the bus nodes have corresponding virtual component interfaces (VCIs) to the PE array. Stuttard, 13:6-20, 21:20-22. Though different VCIs are shown in various locations in Figure 7’s “conceptual view” (Stuttard, 13:1-5), POSAs understood that they all physically connect to the bus. Franzon, ¶115 (citing corroborating EX1082, 1:13-21 (“On-chip buses are used to interconnect ‘virtual’ components” which “perform distinct functions but are not implemented as physically separate devices.”)).

Stuttard explains that VCIs are “defined by the VSI Alliance.” Stuttard, 13:6-9. The VSI Alliance defined VCIs as providing “a point-to-point connection between **two units**.” EX1088, 5. Franzon, ¶116. As Figure 8 illustrates, the two units connected by Stuttard’s VCIs are the MTAP processor core and ClearConnect bus. Franzon, ¶117.

Stuttard’s Figure 7 illustrates three AVCIs (Advanced VCIs) for the processor core: “Programmed I/O (PIO),” “Streaming I/O (SIO),” and L/S (“load/store”). Stuttard, 19:12-20, 13:6-16; Franzon, ¶117. The PIO and SIO AVCIs are for PEs to exchange data with off-chip devices (e.g., the host). For example, the PIO is “for accessing memory external to the MTAP core,” like accessing “variable sized data **by the PEs**.” Stuttard, 10:20-23. Similarly, the SIO

“allows chunks of contiguous data to be streamed *directly into the memory within PEs.*” Stuttard, 10:23-25. Franzon, ¶117.

Stuttard’s Figure 7 shows that these interfaces connect the bus to respective PIO/SIO “engines,” which connect to every PE in the poly-unit array via the “channels.” Stuttard, 19:22-31, FIG. 7. Exemplary connections from the channels to a PE in the Stuttard-Shirazi chip are circled below. Franzon, ¶118. Similar connections exist to every PE in the processor core. Stuttard, 20:1-4 (“There is an I/O node in each PE” that is “activated” by an “I/O Engine”). Thus, the PIO/SIO AVCIIs are input-output interfaces that connect to every poly-unit PE (via their engines and channels). Franzon, ¶118.

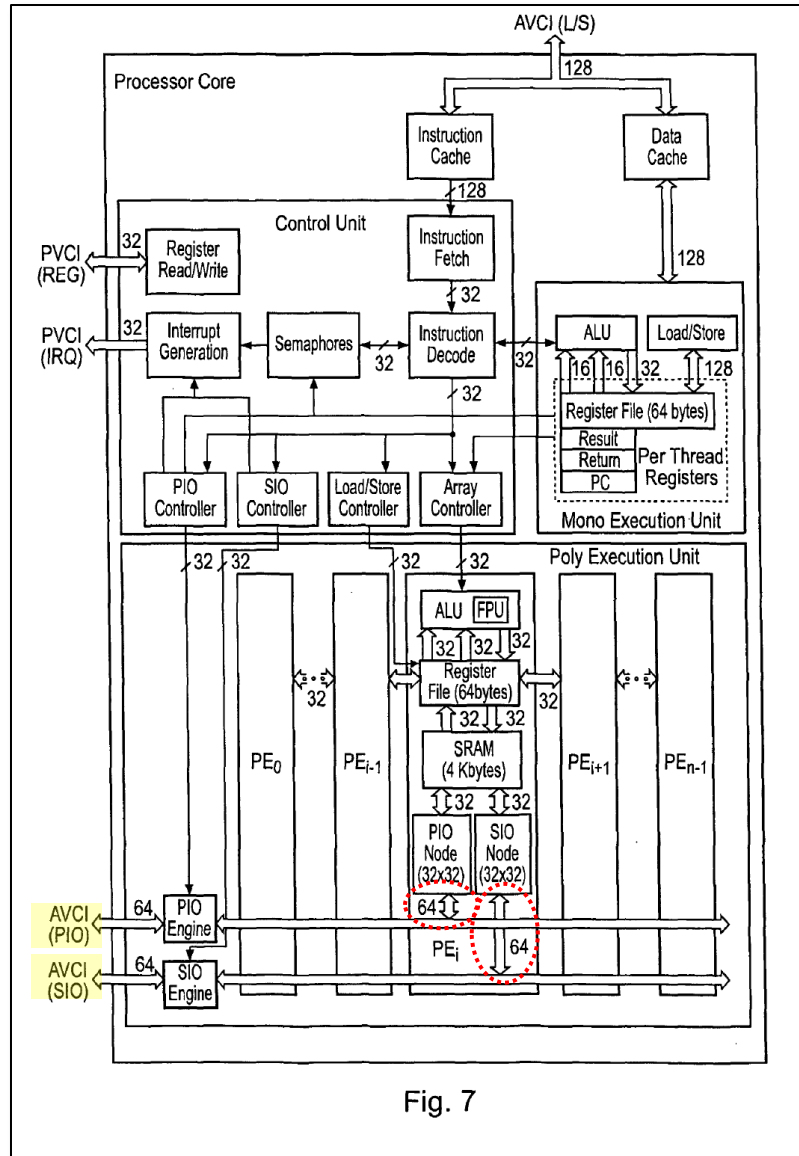


Fig. 7

The AVCI (L/S) is another one of the bus's input-output interfaces to the MTAP processor core: inputs include instructions fetched from the host; outputs include data from the mono unit. Stuttard, 13:6-13, 15:17-25, 9:17-21, FIG. 5. Franzon, ¶119. As Figure 7 illustrates, the AVCI (L/S) connects every poly-unit PE to the host: e.g., instructions from the host pass through the L/S port into the control unit, then to the PIO/SIO engines, which send the instructions via the

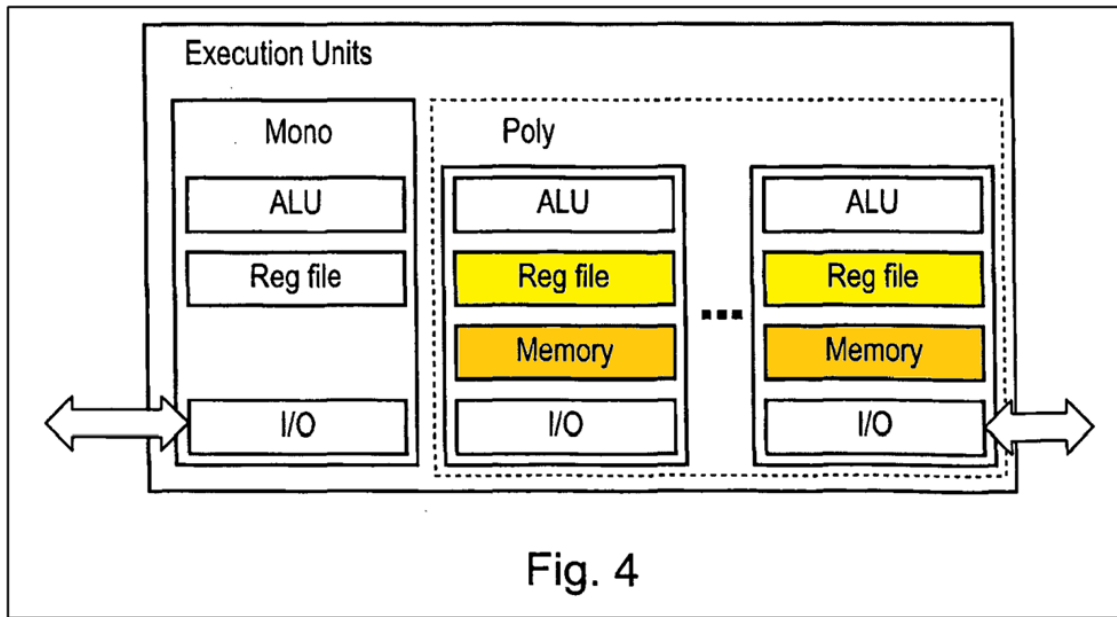
channels which are connected to every PE in the array (as discussed above).

Stuttard, 14:30-34, 15:17-25, 17:20-23. Franzon, ¶119.

Thus, the PIO, SIO, and L/S AVCI and their respective ports on the bus provide connections between every PE in the array and the bus. As discussed above, the bus connects the MTAP processor to, for example, a host interface. Stuttard, 22:3-17. The bus (or alternatively, the bus plus the I/O engines and channels), therefore, is an input-output unit connected to the first and second edge PEs as claimed (because it is connected to all the poly PEs). Franzon, ¶120.

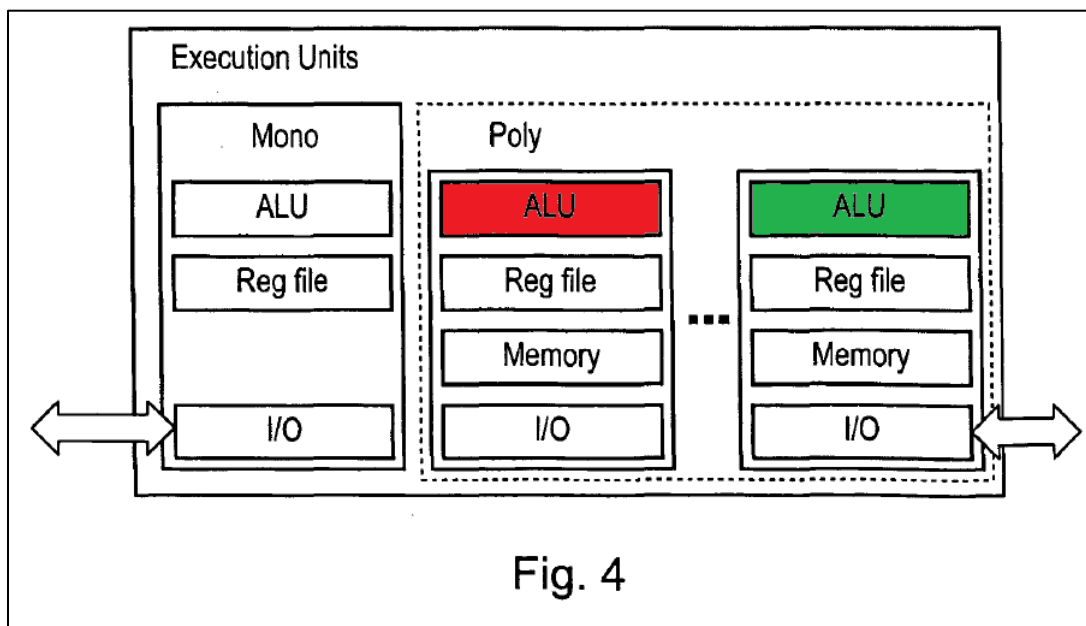
7. [1F]: Each Processing Element of Limitation [1C] Has a “memory local to” It

Stuttard’s Figure 4 (annotated below) illustrates that “[e]ach PE [] has its own independent *local memory*” and addressable “register file,” each of which is memory local to the PE, as claimed. Stuttard, 10:14-19, 5:10-11, 5:29-30, 6:18-20, 7:14-16, 10:1-4, 15:9-16, 17:12-20, FIGS. 3-4, 7; Franzon, ¶121. POSAs understood a register file is an array of registers that is addressable as memory. Franzon, ¶121 (citing corroborating EX1058, [0003] (“‘register file’ is an array of registers that may be addressed as a unit”); EX1075, 2:14-16 (“typical register file is a memory array”).).



8. [1H]: Each Processing Element of Limitation [1C] Comprises an “arithmetic unit”

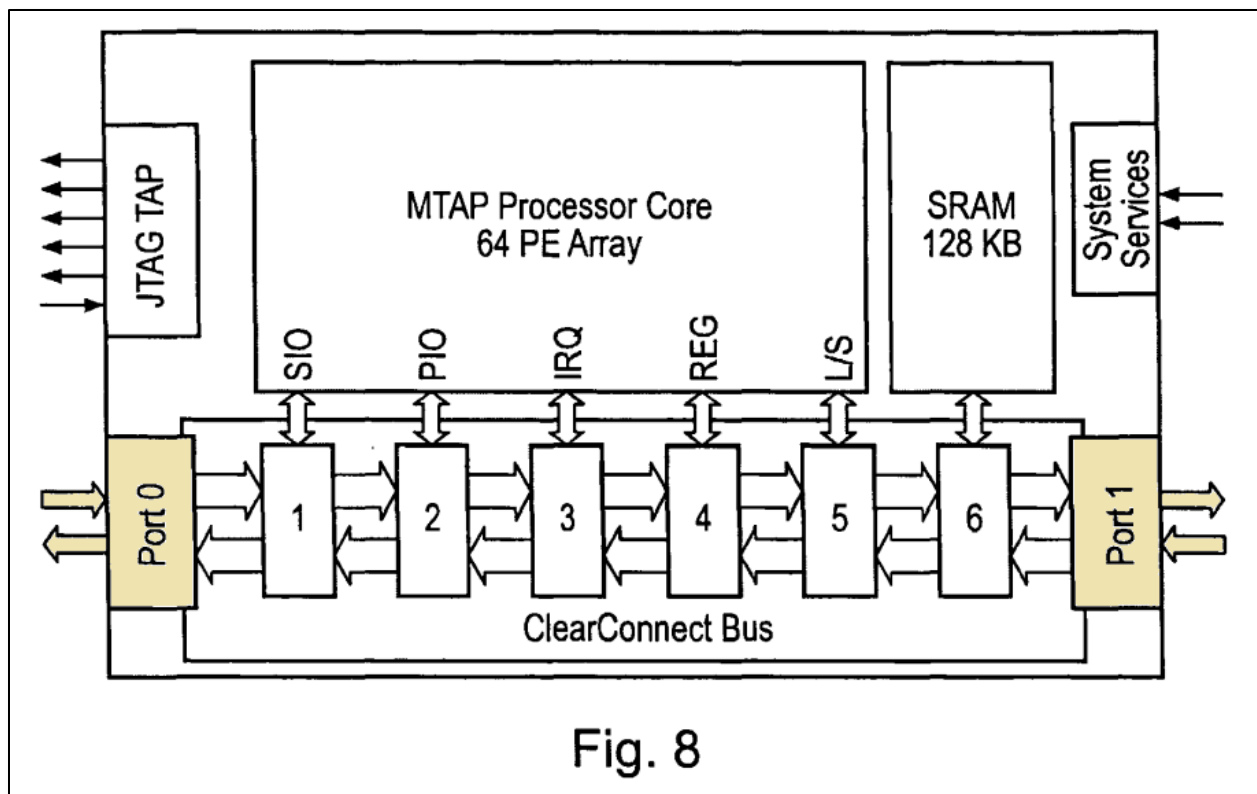
As Stuttard’s Figure 4 (annotated below) illustrates, “[*e*]ach *PE*” comprises “an ALU” (*arithmetic logic unit*). Stuttard, 7:14-16, 5:29-30, 10:1-4, 10:14-19, 17:12-23, Claims 9-10, FIGS. 3-4, 7.



Thus, limitation [1C]’s two edge and two interior PEs each comprise an ALU, meeting the claimed first through fourth arithmetic units. Franzon, ¶123.

9. [1I]: “a host connection at least partially connecting the input-output unit with the host computer;”

As discussed above (§V.B.6), the Stuttard-Shirazi processor chip’s bus (input-output unit) has on-chip nodes (e.g., $1 \leftrightarrow \text{SIO}$, $2 \leftrightarrow \text{PIO}$, etc., in Figure 8 below) connected to the processor core and has “external ports” (Ports 0 and 1 annotated below) “connect[ed] to an FPGA [field programmable gate array] to provide other functions, such as...a **host interface**.” Stuttard, 22:8-19, 21:13-15 (processor has “interfaces to the host system”); Franzon, ¶124.



The PIO node is used “to transfer data to and from the outside world,” including to/from “a host processor.” Stuttard, 20:5-9, 20:22-30, 17:22-23, 19:12-17; Franzon, ¶125. Similarly, the L/S (load/store) node provides an “external interface[]” for “instruction fetching.” Stuttard, 13:6-13 POSAs understood it was conventional for a host system that provides data (as in Stuttard) to also provide instructions for processing that data. Franzon, ¶126. Stuttard notes as background information that SIMD co-processors, of which Stuttard’s is an example (*see, e.g.*, Stuttard, Abstract), conventionally receive data and instructions from a “general purpose [host] processor.” Stuttard, 1:16-24. These communications with the host occur in Stuttard via the bus’s external ports. Franzon, ¶¶126-127.

Thus, the external ports connecting the on-chip bus (input/output unit) to the host system (providing the “host interface” that Stuttard describes—Stuttard, 22:8-19, 21:13-15) are one or more host connections at least partially connecting the input-output unit (including the bus) with the host computer, as claimed. Franzon, ¶128.

10. [1J]: “wherein the first, second, third and fourth arithmetic units each comprises a corresponding multiplier circuit adapted to receive as a first input to the corresponding multiplier circuit a first floating point value having a first binary mantissa of width no more than 11 bits and a first binary exponent of width at least 6 bits, and to receive as a second input to the corresponding multiplier circuit a second floating point value having a second binary mantissa of width no more than 11 bits and a second binary exponent of width at least 6 bits;”

As discussed above (§V.B.8), each PE in the Stuttard-Shirazi poly unit includes an ALU (meeting the claimed first through fourth arithmetic units). As shown in Stuttard’s Figure 7 (annotated below), each ALU has an FPU (floating-point unit), which comprises circuits for performing floating-point operations, including floating-point multiplication. Stuttard, 12:7-15, 13:21-28 (disclosing instructions for “floating point adds, subtracts, **multiplies**, divides”), 17:24-33 (ALUs can include “extensions” “to accelerate functions for floating point”); Franzon, ¶129.

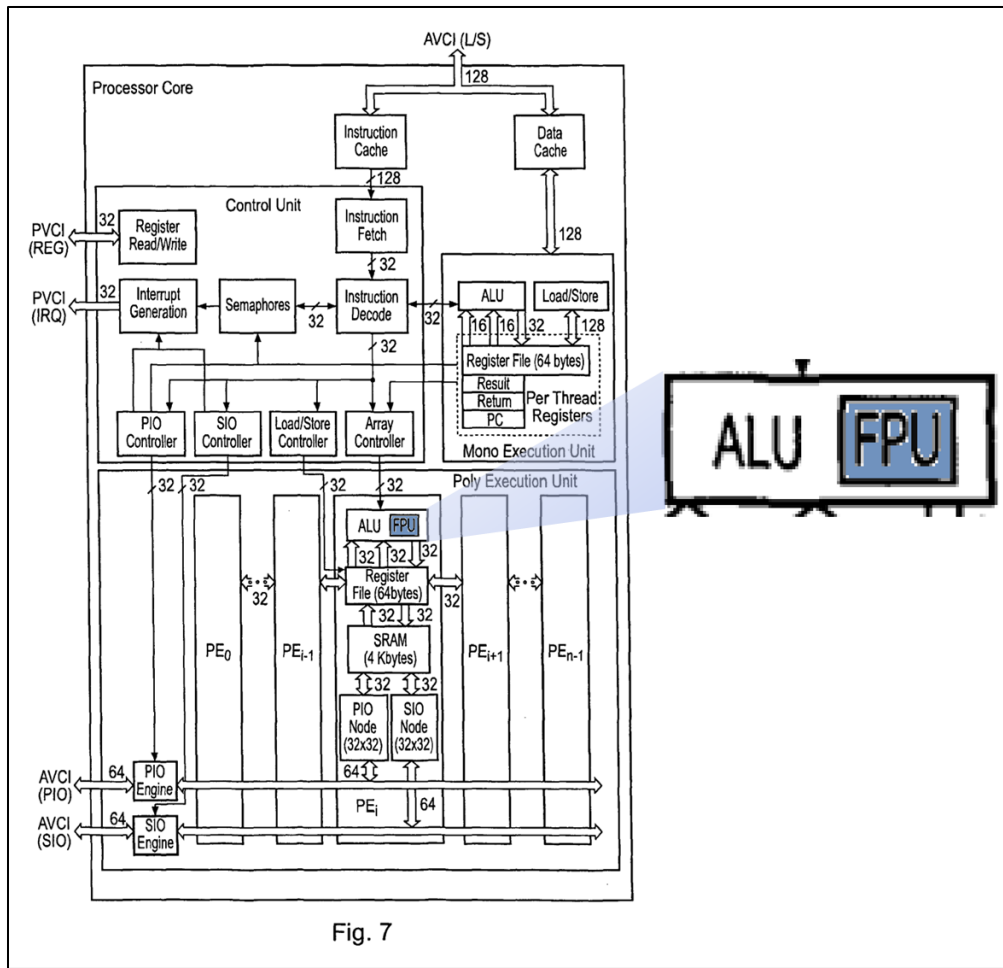
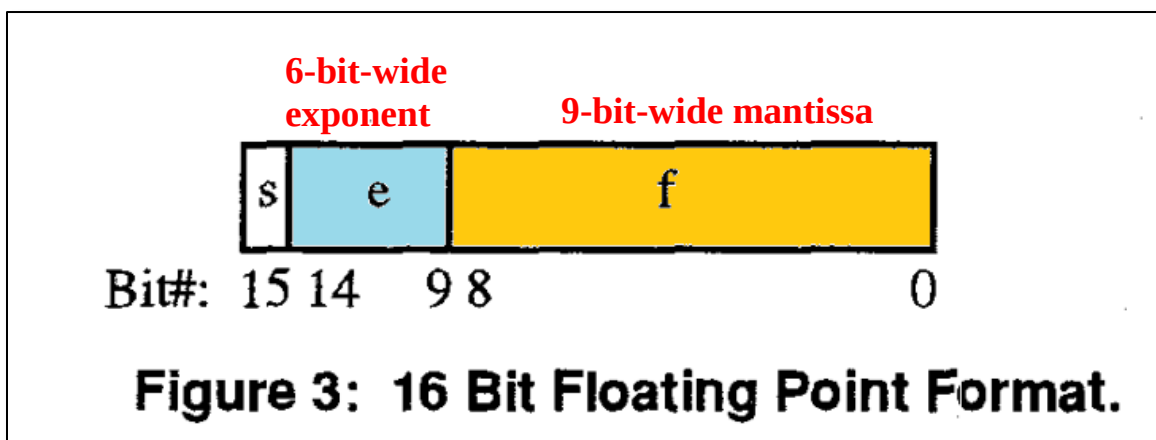
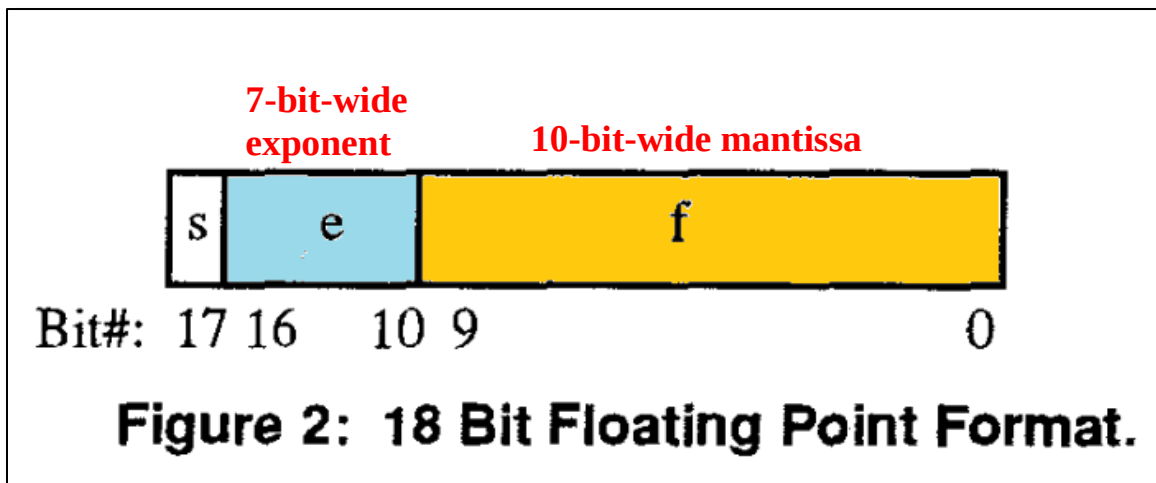


Fig. 7

Stuttard's mono unit (discussed §V.B.11 below) also “include[s] a multiplier” for floating-point operations. Stuttard, 16:13-20. POSAs understood that, like the mono unit's FPU, each FPU in the poly unit also includes a multiplier circuit for performing the floating-point multiplication that Stuttard describes both the mono unit and the poly PEs performing. Franzon, ¶130. A multiplier circuit would have also been the conventional way to perform the described multiplication in an FPU. Franzon, ¶130 (citing corroborating EX1109, 1:17-36).

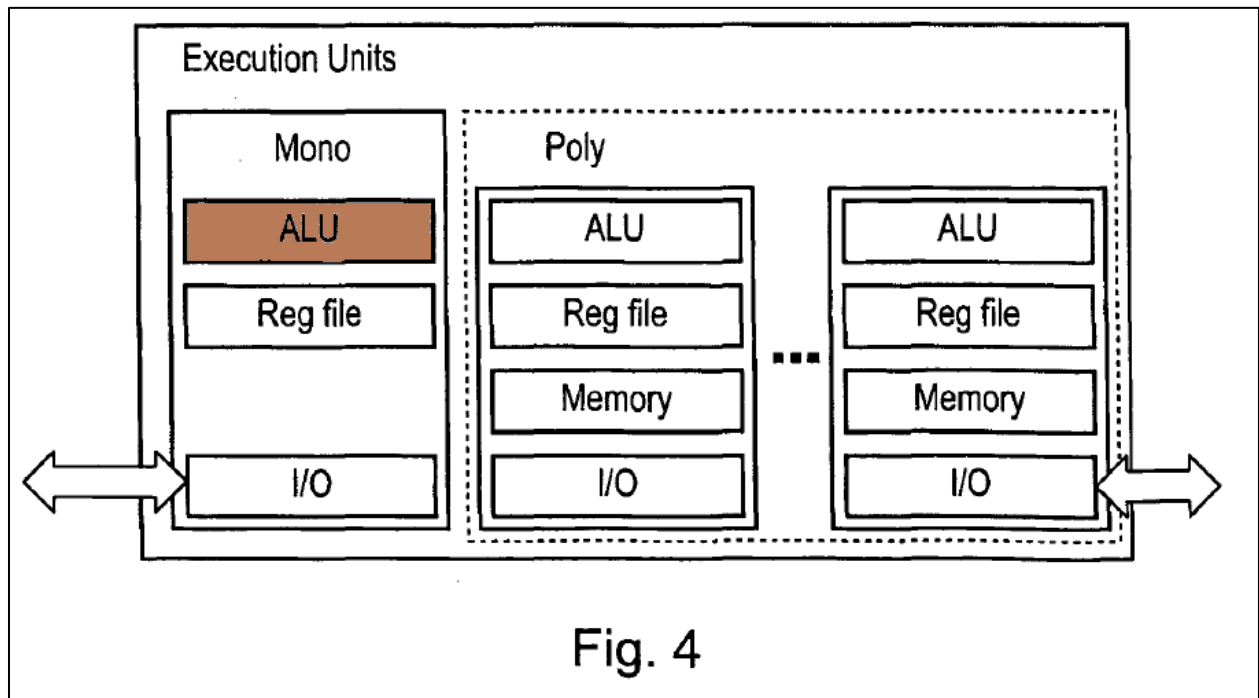
In the Stuttard-Shirazi processor, the multiplier circuit in each poly PE is adapted to receive as inputs two of Shirazi's reduced-bitwidth floating-point values, as discussed *supra* §V.A.3.a. In one implementation using Shirazi's 18-bit format, each of the two input values has a 7-bit-wide exponent, a 10-bit-wide mantissa, and one sign bit; in another implementation using Shirazi's 16-bit format, each input value has a 6-bit-wide exponent, a 9-bit-wide mantissa, and one sign bit; as illustrated in Shirazi's Figures 2-3 (annotated below); Franzon, ¶131.



Thus, each multiplier circuit in the Stuttard-Shirazi poly unit (including in the first through fourth arithmetic units) adapted to receive as first and second inputs two floating-point values having respective first and second binary exponents of width at least 6 bits and respective first and second binary mantissas of width no more than 11 bits, as claimed. Franzon, ¶131.

11. [1G], [1K]: “a fifth arithmetic unit;” “wherein the fifth arithmetic unit comprises a corresponding multiplier circuit adapted to receive as inputs to the corresponding multiplier circuit two floating point values each of width at least 32 bits;”

Stuttard’s mono unit has an ALU (*arithmetic* logic *unit*), which is a fifth arithmetic unit, as claimed. Stuttard, 7:14-24, 3:15-29, 16:13-20, FIGS. 3, 4 (annotated below), 7; Franzon, ¶122, ¶132.



This ALU performs “floating point...multiplies” (Stuttard, 13:26-28) using “extensions” that “include a **multiplier**” (Stuttard, 16:13-20), and can be “a 32-bit ALU” (Stuttard, 3:15-18). POSAs would have understood that Stuttard thus teaches providing a 32-bit floating-point multiplier circuit in the mono unit. Franzon, ¶132.

POSAs also had reason to include in each mono execution unit at least one multiplier circuit capable of multiplying two standard 32-bit-wide floating-point values. Stuttard taught that “[i]t is not necessary for the [poly] PEs to be identical to the mono execution unit in every detail: e.g. ...the **width and complexity of the ALU are likely to be different** in practice.” Stuttard, 7:14-24 (“mono execution unit is likely to require a **wider and more complex** ALU”). This would have suggested to POSAs to include in Stuttard-Shirazi’s mono unit an FPU that supported higher bitwidths than the FPUs in the poly unit. POSAs had reason to use the IEEE’s 32-bit values in a mono unit because POSAs would have wanted the Stuttard-Shirazi chip to support non-parallel data and instructions from a conventional, IEEE-standard compliant host. Franzon, ¶133 (citing as corroboration EX1012, 232-233, EX1034, 274).

Thus, POSAs would have recognized the benefit in implementing the mono unit’s floating-point multiplier circuit to receive as inputs two IEEE-standard 32-bit-wide floating-point values, meeting [1K]. Franzon, ¶134. POSAs would have

reasonably expected success because, as noted above, Stuttard taught that the mono unit's ALU can be "wider and more complex" than the poly unit's ALUs and can be "a 32-bit ALU." Stuttard, 7:14-24, 3:15-18; Franzon, ¶134.

12. [1L]: Each Multiplier Circuit of [1J] "comprises a [first/second/third/fourth] plurality of transistors and has no other transistors"

Stuttard discloses an exemplary chip "[i]mplemented on a 0.13 μ process" (Stuttard, 12:8-9), which POSAs understood refers to the 0.13-micron (130-nm) semiconductor node for fabricating transistors. Franzon, ¶¶135-136. By the 2000s, floating-point multiplier circuits were conventionally implemented using transistors. Franzon, ¶136 (citing corroborating EX1100, 769 (example floating-point unit in "90-nm SOI technology" had "transistors"))).

Because each PE in the Stuttard-Shirazi poly unit has a floating-point multiplier for multiplying values no wider than Shirazi's 18-bit (or 16-bit) floating-point values (see §V.B.10 above), POSAs would have designed each multiplier circuit to have sufficient transistors to support inputting and multiplying two of Shirazi's 18-bit (or 16-bit) values. Franzon, ¶137. Only the transistors necessary to support those maximum input widths supported by the multiplier would have been included in the circuit. Franzon, ¶137.

13. [1M]: “the multiplier circuit corresponding to the fifth arithmetic unit comprises a fifth plurality of transistors; and,”

Just as the first through fourth multiplier circuits comprise respective pluralities of transistors (§V.B.12 above), POSAs knew the multiplier circuit in the “fifth” ALU (in Stuttard-Shirazi’s mono execution unit, §V.B.11 above), comprises a fifth plurality of transistors. Franzon, ¶138.

14. [1N]: the “fifth plurality of transistors exceeds in number each of the” first through fourth “plurality of transistors”

The ’775 patent does not disclose the absolute or relative number of transistors for any multiplier circuit. It only says that “precision is costly”—e.g., a “double precision floating point multiplier may require on the order of a million transistors,” whereas fewer transistors are “required to perform a low precision multiplication.” EX1001, 7:8-13; Franzon, ¶139.

As discussed *supra* §V.B.12, Stuttard-Shirazi’s first through fourth multiplier circuits comprise the number of transistors needed to multiply two input values no wider than Shirazi’s 18-bit (or 16-bit) floating-point numbers. Shirazi, 155-56. The same “PEs are *replicated* many times” in the poly unit (Stuttard, 7:22-24); thus POSAs understood that each poly-unit PE’s floating-point multiplier circuit had about the same number of transistors. Franzon, ¶140.

Similarly, the fifth multiplier circuit (in the mono unit) has the number of transistors needed to receive and multiply two floating-point values at least 32 bits

wide. §V.B.11 above. POSAs would have understood that a multiplier circuit adapted to receive and multiply two 32-bit-wide floating-point values would conventionally require more transistors than a similarly architected multiplier circuit adapted to receive and multiply two lower-bitwidth floating-point values. Franzon, ¶141 (citing EX1076, 5:21-26 (corroborating POSA’s knowledge that “the size of the multiplier is proportional to the number of bits multiplied”)).

Thus, POSAs would have understood that in a conventional and obvious implementation of the Stuttard-Shirazi processor, the fifth plurality of transistors exceeds in number each of the first through fourth plurality of transistors, as claimed. Franzon, ¶¶142-143.

C. Claim 2: “said host computer is programmed to provide instructions to said computing chip that, when executed, cause said processing element array to perform an operation whose output is used to identify at least one image, from a plurality of images to be searched, that is similar to at least one input image.”

As discussed above (§V.B.9), the Stuttard-Shirazi host computer is programmed to provide instructions that, when executed, cause the Stuttard-Shirazi processor to perform data-processing operations. Stuttard’s processor architecture is suitable for accelerating “a number of computing applications.” Stuttard, 1:1-6, 8:26-28, 23:17-24:21 (describing two exemplary applications: “Network processing” and “Bio-informatics”). One application POSAs had reasons to

accelerate using the Stuttard-Shirazi processor is an image search-and-identification application. Franzon, ¶144.

Stuttard's processor is for "applications [that] are characterized by very high data rates." Stuttard, 8:24-9:5. POSAs knew that image-identification applications generally required very high data rates and were advantageously performed using parallel PEs, like those in Stuttard's SIMD processor. Stuttard, Title ("Unified ***SIMD Processor***"), 3:7-9; Franzon, ¶¶145-148 (citing corroborating EX1063, [0023]-[0024] ("image classification systems" require "raw processing power," and "[m]any operations in image processing are fundamentally Single Instruction Multi Data (SIMD)"), [0080]; EX1107, 822-823, 825-826 ("application areas" for a similar SIMD chip include "Image Processing" and "Search Engine"); EX1111, 1-2 ("Content Based Image Retrieval (CBIR)...allows one to find out images of a database that are (at least) partly similar to a given reference image" and "often require[s] a low response time," so "a parallel machine is the most natural choice" for "CBIR systems"); EX1103, 1101 ("pattern classification" "is time devouring" and benefits from parallel processors)).

Thus, POSAs knew that Stuttard-Shirazi's massively parallel PE array would have been particularly well-adapted to image-identification applications. Franzon, ¶145.

A contemporaneous patent application by Stuttard's Applicant taught using parallel processors like Stuttard's (EX1050, 8:19-25) to perform "image comparison" applications, like "Optical Character Recognition (OCR)" and "Identify Friend or Foe (IFF) systems," involving "rapid **comparison of an image with a 'target'**" using sum of square of difference (SSD) computations (EX1050, 18:1-8). Franzon, ¶146. *See also* EX1050, 3:30-4:6 (noting SSD requires multiplications). Other well-known image-identification applications included fingerprint identification and image-database search and retrieval. Franzon, ¶147 (citing corroborating EX1077, 1:55-60, 7:6-12; EX1078, 3:15-24, 13:28-52)). These applications conventionally required identifying at least one image, from a plurality of images (e.g., fingerprint database, or other set of images being searched), that is similar to one input image (e.g., a fingerprint of a person to be identified, or a photo something to be searched for in other photos), as claimed. Franzon, ¶¶146-147 (citing corroborating EX1077, 1:55-60, 2:19-21, 3:58-60, 7:6-12, 10:35-53; EX1078, Abstract, 1:35-38, 2:19-21, 3:15-24, 4:33-56, 5:6-8:4, 13:1-15, 13:28-52; EX1104, 81-82 ("face recognition" applications achieve "performance increase[s] through the use of massively parallel data processing," e.g., with "SIMD" "architecture"))).

Accordingly, POSAs would have readily understood the suitability and desirability of using Stuttard-Shirazi's massively parallel PE array to identify an

image, from a plurality of images to be searched, that is similar to an input image. Franzon, ¶149.

Using a massively parallel processor like Stuttard-Shirazi to perform image-matching applications would have also been applying a known technique (programming a massively parallel processor to perform image matching) to improve a similar massively parallel-processor chip like Stuttard-Shirazi in the same way (by programming it to support image-matching applications). *KSR*, 550 U.S. at 417. Franzon, ¶150.

POSAs would have reasonably expected success using the Stuttard-Shirazi processor to perform these image-identification applications because, as noted above, Stuttard's processor is for "very high data rate[]" applications (Stuttard, 8:24-28) using a "highly parallel execution unit" that could "be *easily customized* for the target application" (Stuttard, 9:9-14). Franzon, ¶151. Stuttard also disclosed "a simple extension to standard C" (a conventional programming language) "to identify data which is to be processed in parallel," thereby facilitating using the chip for various applications. Stuttard, 11:11-14; Franzon, ¶151.

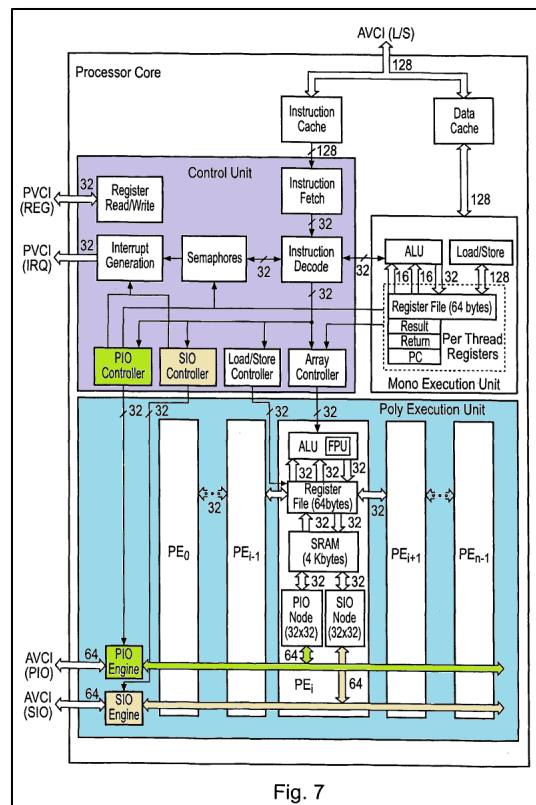
When performing such image-identification applications, the host computer provides instructions to be performed by the Stuttart-Shirazi poly-unit PEs to accelerate the image-identification application, meeting claim 2. Stuttard, 1:21-24,

2:34-3:6, 6:3-26, 7:5-10, 10:27-11:16, 13:6-13, 14:9-10, 21:13-25; Franzon, ¶¶152-153. See also §V.D.2 below (claim 3: describing control signals conveying instructions from the host to the processing elements).

D. Claim 3

1. [3A]: “the computing chip further comprises a control unit connected to the processing element array,”

As Stuttard’s Figure 7 (annotated below) illustrates, the Stuttard-Shirazi processor core comprises a control unit (purple). Stuttard, 4:20-5:28, 14:8-10, 19:22-27, 21:20-22, Claims 1-4, FIG. 3. The control unit is connected to the poly PE array (blue) via, e.g., the PIO (green) and SIO (yellow) controllers, engines, and channels. Stuttard, 14:8-26, 19:22-27. Franzon, ¶154.



2. [3B]: “the control unit comprising circuitry adapted to decode at least one instruction received from the host computer via the input-output unit, and”

Stuttard-Shirazi’s “control unit *fetches instructions* from memory [and] *decodes them.*” Stuttard, 14:9-10, 9:17-18. The PIO and SIO controllers in the control unit similarly “*decode I/O instructions*” for the PE array. Stuttard, 19:25-26. POSAs understood these decoding operations were performed by circuitry in these units. Franzon, ¶155. Thus, Stuttard-Shirazi’s control unit comprises circuitry adapted to decode at least one instruction. Franzon, ¶155.

POSAs would have understood that the instructions decoded by the control unit are received from the host computer via the bus and its AVCI (L/S) interface (i.e., the input-output unit—§V.B.6 above). Stuttard, 13:11-13 (AVCI interface “is used...for instruction fetching”); Franzon, ¶156. Stuttard also explains that with SIMD arrays, like Stuttard-Shirazi’s, the “host processor” will “send data to be processed...with an indication of the function to be performed, typically a pointer to the start of a sequence of SIMD array *instructions*” (i.e., the host computer provides the instructions to be decoded). Stuttard, 1:21-24, 9:6-33, 13:6-20, 14:23-26; Franzon, ¶156. The pathway for instructions received via the AVCI (L/S) into the Control Unit’s “Instruction Decode” block is shown in Stuttard’s Figure 7 (annotated below). Franzon, ¶156.

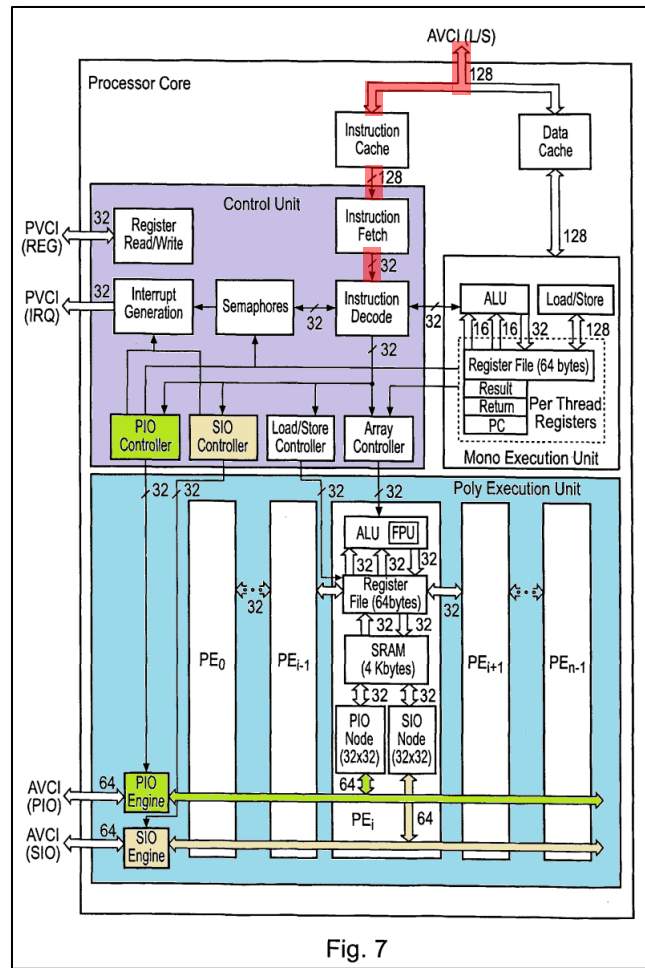


Fig. 7

3. **[3C]: “to send at least one control signal to the processing element array to cause the processing element array to operate according to the at least one instruction.”**

After the control unit “decodes” instructions, it “despatches them to the appropriate functional unit,” including the “poly execution unit.” Stuttard, 14:8-10. POSAs understood that such “despatche[d]” instructions comprise signals that cause the poly-unit PEs to operate according to those instructions, as claimed (e.g., signals causing PEs to perform instructions for “floating point adds, subtracts, multipliers divides”). Stuttard, 13:21-14:26; Franzon, ¶157. The instructions as

“despatche[d]” by the control unit thus include control signals, as claimed.

Franzon, ¶¶157-158.

- E. Claim 4: “the computing chip further comprises an instruction memory adapted to store the at least one instruction received from the host computer via the input-output unit, wherein the control unit is further adapted to retrieve the at least one instruction from the instruction memory.”**

The Stuttard-Shirazi processor “include[s] instruction and data caches to minimize the latency of memory accesses.” Stuttard, 9:26-27, 9:6-12, 22:20-27, Figure 7 (“Instruction Cache” and “Data Cache” annotated below); Figure 3 (cache annotated below).

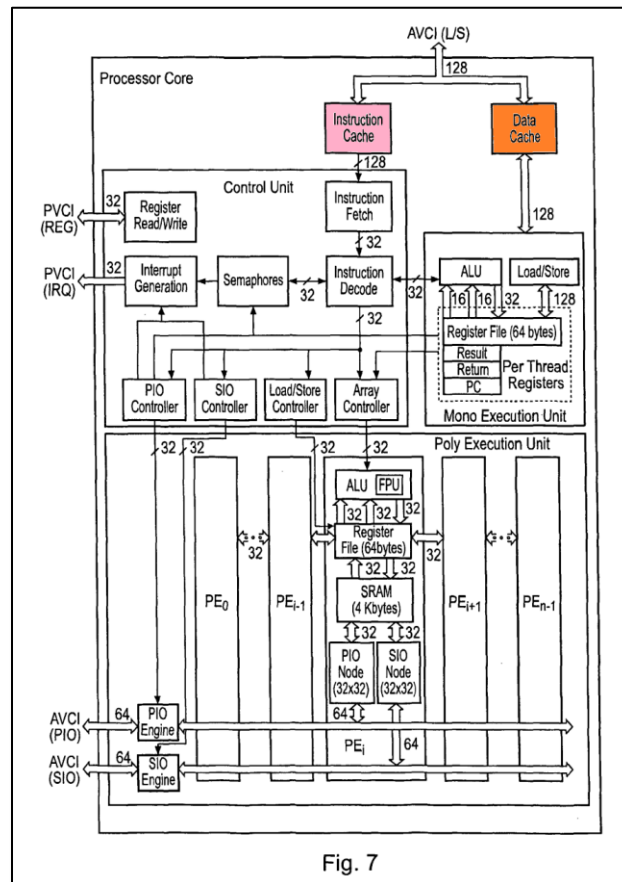
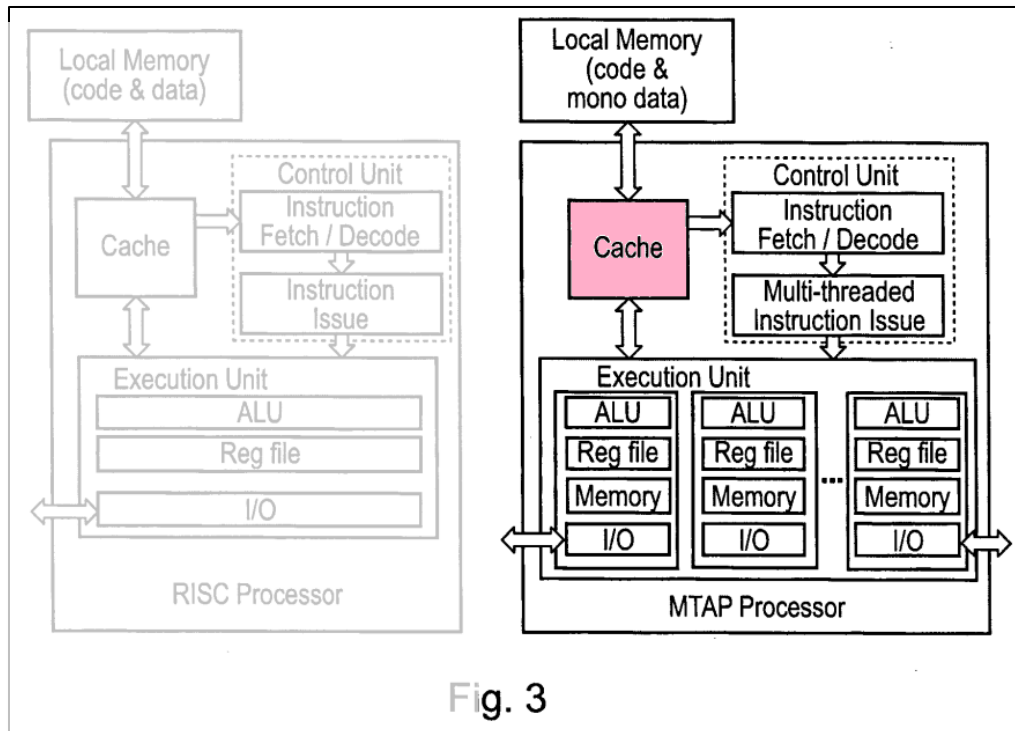


Fig. 7

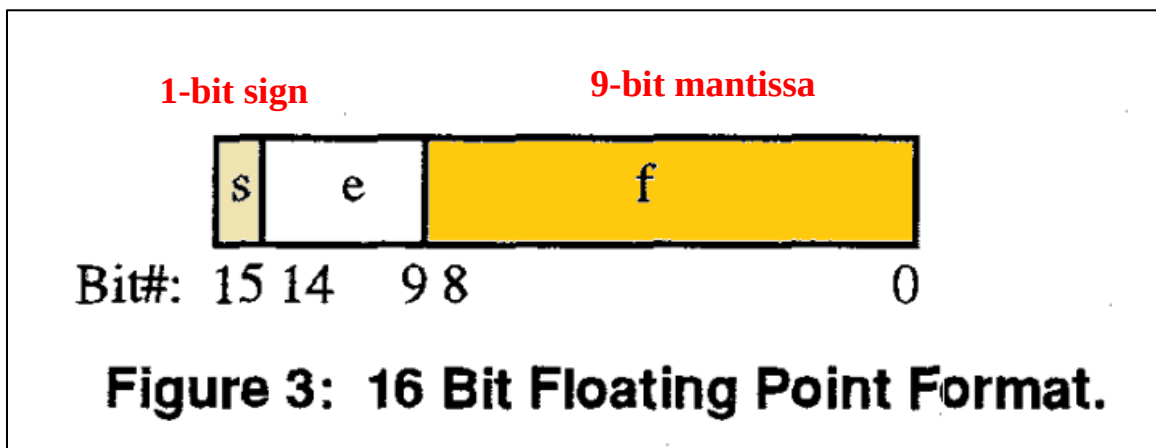
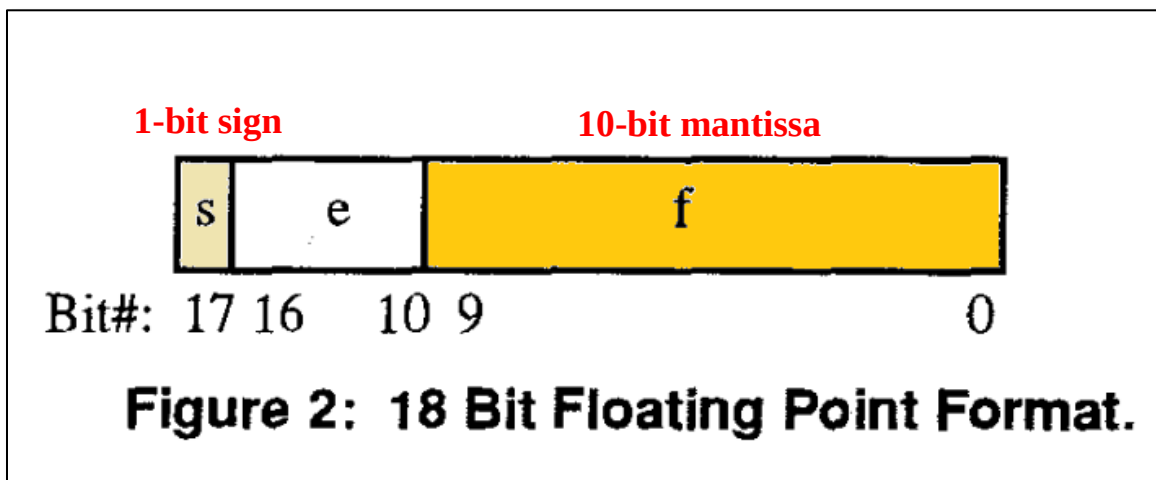


POSAs would have understood that Stuttard teaches latency is minimized when the on-chip control unit retrieves instructions from the instruction cache, rather than from the off-chip host computer, because on-chip communication to an instruction cache is generally faster than off-chip communication to the host system. Franzon, ¶¶159-161; Stuttard, 22:16-19 (“*on-chip* SRAM...*provides fast access* to code and data”).

Thus, POSAs would have understood that Stuttard’s “instruction cache” is memory adapted to store the instruction(s) discussed above (§V.D.2) that are received from the host computer via the input-output unit and then retrieved by the control unit from the cache, meeting claim 4. Franzon, ¶¶162-163.

F. Claim 5: “the width of each [first/second] binary mantissa is no more than 10 bits plus a [first/second] sign bit”

As discussed above (§V.B.10), in Stuttard-Shirazi, each poly-unit PE’s multiplier circuit is adapted to receive two floating-point values, each comprising a 10-bit-wide or 9-bit-wide (and no wider) mantissa plus one sign bit, as illustrated in Shirazi’s Figures 2-3 (annotated below):



Shirazi, 156; Franzon, ¶¶164-165.

VI. GROUND 1B: CLAIMS 7-14, 16-23 ARE UNPATENTABLE OVER STUTTARD IN COMBINATION WITH SHIRAZI

A. Scaling the Stuttard-Shirazi Processor

1. 5,000+ Processing Elements

Limitation [7C] recites a “plurality of first processing elements [that] is no less than 5000 in number.” POSAs had reasons to scale the Stuttard-Shirazi processor in a manner that would have resulted in over 5000 reduced-bitwidth PEs on one computing chip. Franzon, ¶166.

First, Stuttard taught that one “poly execution unit may consist of...*thousands* of PEs” within an MTAP processor core. Stuttard, 10:12, 12:15-16 (each processor “core can be scaled...to 256 or even *thousands of PEs*”). EX1108, which was co-authored by two of Stuttard’s inventors, similarly taught that the “poly execution unit [in an MTAP processor core]...may consist of...*thousands of identical processor element (PE) cores*.” EX1108, 32-33 (non-bold emphasis original). Franzon, ¶167.

Moreover, Stuttard’s processor architecture was expressly intended for “high performance, high data rate processing.” Stuttard, 8:24-25. The “processing power *can scale* with increasing function demands.” Stuttard, 9:2-3. “The poly execution unit,” in particular, “get[s] its performance largely from *the number of PEs brought to bear* on the problem.” Stuttard, 7:19-20.

As discussed *supra* §V.C above, POSAs would have understood Stuttard's PE array to be well-suited to image-retrieval applications. Franzon, ¶168. POSAs knew, for example, that in content-based image retrieval (CBIR) systems the "response time requires to be low for good interactivity," so such systems "could be implemented using" SIMD processors like Stuttard's (Franzon, ¶168, quoting EX1110, 607-08); and that "reduc[ing] the retrieval time...is very important to the massive CBIR systems" (Franzon, ¶168, quoting EX1113, 543). Thus, POSAs would have recognized the benefit in, and market forces driving, increasing the number of PEs on a chip to be as high as reasonably achievable (*see* §VI.A.3 below discussing POSAs' reasonable expectation of success in achieving 5000+ PEs), to increase the system's processing performance (e.g., for applications like CBIR), at lower power and/or lower cost compared to multi-chip systems having equivalent processing performance. *KSR*, 550 U.S. at 417 ("design incentives and other market forces can prompt variations"); Franzon, ¶168 (citing as additional corroboration:

- EX1114, 89 (Stuttard's Applicant noting "the current unstoppable trend to put more and more cores on a single die");
- EX1013, 394 ("[T]he more cores that can be packed into a single piece of silicon, the more parallelism that can be exploited."); 403 ("[A]n increase in the number of cores will mean more exploitation of parallelism.");

- EX1014, 9 (“[R]eduction in core area [from lower-bitwidth PEs] can be leveraged to increase the number of cores in the same silicon area.”), 1 (“reduc[ing] the size of floating point units (FPUs)[] allow[s] more of them to occupy the same silicon area”; “increased number of FPUs can significantly improve the performance of future physics accelerators”);
- EX1084, 1:25-37 (“[T]hroughput of a SIMD parallel processor is determined by the number of PEs multiplied by the throughput of each individual PE.... Generally, the more PEs per chip, the less costly the implementation for a given level of performance.”);
- EX1106, 4 (“Parallel execution in multi-core designs will then allow us to take advantage of these greater transistor densities to provide greater performance...Many simple cores can be built within the same area as a small number of large complex cores.”), *id.* (“[W]e must increase the efficiency and number of cores on a chip.”);
- EX1116, 1 (“Advances in semiconductor technology have driven the hardware industry to integrate an increasing number of cores on-chip.”)).

Thus, POSAs had motivation to include up to and beyond 5000 PEs in a Stuttard-Shirazi computing chip. Franzon, ¶169.

Second, the number of PEs recited in claim 7 carries no patentable weight, because “[i]t is well settled that the mere duplication of parts has no patentable significance unless a new and unexpected result is produced.” *In re Harza*, 274 F.2d 669, 671 (C.C.P.A. 1960). The ’775 patent only references “five thousand”

elements once (EX1001, 28:46-51), among a generalized, indiscriminate listing of many different possible numbers of elements, all but one of which are significantly lower than 5000, and none of which are limited to being on a single chip. EX1001, 28:26-29:4; Franzon, ¶170. These embodiments are otherwise indistinguishable: the patent does not allege that any particular number of PEs is critical to the alleged invention, allowed for qualitatively different functionality, or produced new or unexpected results. Franzon, ¶170.

Thus, claim 7's limitation reciting 5000+ PEs (§VI.B.2 below) carries no patentable significance, reinforcing that it would have been an obvious number of PEs to use in Stuttard-Shirazi. *Harza*, 274 F.2d at 671; *Supercell Oy v. Gree, Inc.*, PGR2018-00008, Paper 42, 47 (Jan. 2, 2019) (“[I]t is axiomatic that duplication has no patentable significance.”) (citing *Harza*). Franzon, ¶171.

Third, “it is not inventive to discover the optimum or workable ranges by routine experimentation” when the “optimized variable is a result-effective variable.” *In re Applied Materials, Inc.*, 692 F.3d 1289, 1295 (Fed. Cir. 2012) (internal quotation marks and citations omitted). Stuttard expressly taught that the number of PEs is a result-effective variable, optimized by routine experimentation: “**choosing the number of PEs** and the clock frequency” provides the designer with “a great deal of **flexibility to tune** performance, bandwidth, cost and power for the needs of a given application.” Stuttard, 12:18-21; *see also* 9:13-14 (Stuttard’s

processor is “*easily customized* for the target application”); Franzon, ¶¶171-172 (citing corroborating EX1084, 1:25-36 (“[T]hroughput of a SIMD parallel processor is determined by the number of PEs.”)).

Thus, selecting the number of PEs in a Stuttard-Shirazi chip would have been routine optimization of a result-effective variable, such that arriving at 5000+ PEs would have been obvious. *Applied Materials*, 692 F.3d at 1295; *Genentech, Inc. v. Hospira, Inc.*, 946 F.3d 1333, 1341 (Fed. Cir. 2020) (“[I]t is not inventive to discover the optimum or workable ranges by routine experimentation.”); *E.I. DuPont de Nemours & Co. v. Synvina C.V.*, 904 F.3d 996, 1010 (Fed. Cir. 2018) (“[D]iscovery of an optimum value of a result effective variable in a known process is ordinarily within the skill of the art.”); *In re Urbanski*, 809 F.3d 1237, 1242-43 (Fed. Cir. 2016) (upholding obviousness finding where there was “no evidence” that “claimed range[]” of result-effective variable was “critical”).

2. Plurality of 32-Bit Floating-Point Multipliers

Limitation [7H] (§VI.B.7 below) recites a “plurality of second arithmetic units each compris[ing]” a “multiplier circuit adapted to receive...two floating point values each of width at least 32 bits.” POSAs had reasons to implement the scaled Stuttard-Shirazi processor in a manner having a plurality of 32-bit-wide multipliers. Franzon, ¶173.

First, one way POSAs would have achieved the known goal of increasing the number of PEs on the chip (§VI.A.1 above) would have been to replicate an existing processor core design (with its corresponding ratio of poly to mono execution units) multiple times on the chip. POSAs knew this to be a conventional technique in the art to scale up the capabilities of a computing chip, by leveraging functional blocks from pre-existing designs and thereby saving cost and time by avoiding unnecessary redesign. Franzon, ¶174 (citing corroborating EX1068, [0002]-[0007] (characterizing as “background” knowledge that “[m]odern microprocessor systems derive significant performance enhancements from the implementation of multiple, identical cores on a single integrated circuit (IC)” by making “identical replications of the core...on a single die” which results in “lower design costs”); EX1069, [0004]-[0005] (characterizing as “background” knowledge that “[c]hip multithreading (CMT) processors and chips may include a number of cores” by repeating the “same [core] design copied over multiple times on a die”); EX1106, 4 (“[S]tandard, repeated tiles...into a common infrastructure should allow more reuse of designs between generations of processors.”)).

Thus, duplicating Stuttard-Shirazi’s MTAP core to achieve more PEs on the chip would have also been using a known technique to improve similar devices in the same way. *KSR*, 550 U.S. at 417; Franzon, ¶175.

Second, POSAs would have recognized the benefit in having multiple mono units on a chip to provide several full-precision FPUs, in addition to thousands of reduced-bitwidth FPUs. Franzon, ¶176. For example, Stuttard discloses an exemplary arithmetic operation where each poly-unit PE multiplies a single value stored in the mono unit. Stuttard, 3:12-14. POSAs would have understood that having multiple mono units would have allowed subsets of the chip’s poly-unit PEs to multiply different values stored in different mono units over a shorter time period, compared to a chip having a single mono unit. Franzon, ¶¶176-177 (citing as further corroboration EX1106, 6 (describing a multi-core architecture having a “mix of element types matched to the significant workloads in the segment it is to serve,” where “the number of each and relative number of each type could be varied”); EX1116, 1 (noting recognition in the art that “higher energy efficiency” is achieved by “asymmetric (or heterogenous) architectures”); EX1117, 746-49 (“A many-core system on a die...may have a few large cores to deliver higher single-thread performance, but will predominantly have large number of small cores.”)).

POSAs also understood that having several full-precision FPUs on the chip along with a large number of reduced-bitwidth PEs would “provid[e]...granularity options,” allowing the chip to handle a wider range of tasks and data types. Franzon, ¶176 (quoting corroborating EX1124, 73-76 (“researchers have proposed

heterogenous multicore (also known as asymmetric multicore) processors as a means to improve power efficiency or performance” by having different sized processing elements on a die; “expect[ing] this trend to continue”)).

Stuttard also discloses an exemplary MTAP processor with one mono unit having “[s]upport for 8 threads.” Stuttard, 22:20-30. Stuttard further teaches that “[t]he number of threads is a configurable option” and that “threads are prioritized” so the “processor will run the highest priority thread,” which can “pre-empt a lower priority thread.” Stuttard, 14:18-22. Thus, POSAs also would have wanted to increase the number of mono units on the chip to support parallel processing of additional, highest-priority threads on the chip—in particular threads that could not be efficiently parallelized (i.e., computations better suited for the mono units). Franzon, ¶¶178-179 (citing corroborating EX1119, 1:23-25; EX1122, [0013] (“There is generally a desire to have as many processing cores as possible each concurrently processing as many threads as possible in order to obtain the greatest processing power and efficiency from the processor.”); EX1012, 234 (describing different phases of physics simulations for video game applications, some of which are “not easily mapped to parallel architectures” while others “exhibit[] massive fine-grain (FG) parallelism”)).

Third, the recitation of a plurality of (rather than a single) 32-bit floating-point multipliers on the chip is merely a duplication of parts carrying no patentable

weight, and also is an obvious result-effective variable, for the same legal reasons discussed above regarding the recited number of PEs on the chip. §VI.A.1 above (citing cases). Franzon, ¶180.

Thus, POSAs had reasons to implement the scaled Stuttard-Shirazi processor as one chip having a plurality of mono execution units and multiple poly execution units collectively including over 5000 reduced-bitwidth PEs. Franzon, ¶181.

Stuttard teaches scaling a single MTAP processor core's poly unit to "thousands of PEs." Stuttard, 12:15-16. "Thousands," at a minimum, includes at least 2000 PEs in one processor core (and POSAs would have understood it to indicate more).

Franzon, ¶181. If a POSA following this teaching scaled a Stuttard-Shirazi processor core to, e.g., 2048 PEs (e.g., a 64x32 PE array—just barely "thousands" of PEs), then duplicating this core three times on a chip would have provided an array of 6144 reduced-bitwidth PEs and 3 full-precision mono units (each having a 32-bit floating-point multiplier circuit, as discussed *supra* §VI.A.2, §V.B.11).

Franzon, ¶182. As another alternative example, an exemplary MTAP processor core discussed in Stuttard had one mono execution unit and 256 poly-unit PEs.

Stuttard, 10:12-19, 12:15-16. Duplicating this architecture 20 times on a chip would have provided 5120 reduced-bitwidth PEs and 20 mono execution units.

Franzon, ¶183.

3. Reasonable Expectation of Success

POSAs would have reasonably expected success in scaling the Stuttard-Shirazi processor as discussed *supra* §VI.A.2. Franzon, ¶184.

Stuttard's processor architecture is designed for "easy scalability" (Stuttard, 1:5-6) by being "highly modular" to "allow[] many details of a specific implementation to be easily customized for the target application" (Stuttard, 9:13-14). *See also* Stuttard, 8:29-32 ("The MPAP processor...has a clear scalability path for future requirements."). Stuttard teaches that a "designer has a great deal of flexibility" in "choosing the number of PEs." Stuttard, 12:15-21.³ Stuttard

³ Stuttard at 12:18 states, "Figure 6 shows how performance and power scale with the number of PEs." *See also* Stuttard, 6:34, 10:5-6. This is a typographical error because the referenced figure was never included in the Stuttard application. *See generally* EX1095, 32 (specification from Stuttard's original PCT application at 4:15-26 (describing nine figures)), *id.*, 51-62 (only eight figures included in application)). POSAs would have understood that what published as Stuttard's Figure 6 illustrates the "graphical representation of the results of running an example program" (EX1095, 32 (original specification at 4:21-22)) that Stuttard describes at 11:18-12:6 ("calculat[ing] 64 values of a sine function across the PE

described simulating “the number of PEs, size of memory, and other options” in software “before the system architecture is fully defined” and put into “silicon implementation” (i.e., before fabricated as a real chip) (Stuttard, 24:28-33), which POSAs understood facilitated designing the number of PEs on the chip. Franzon, ¶185.

In 2008, Intel disclosed a microprocessor chip based on its 65-nm process node having over 2 billion transistors. EX1039, 18. By 2008, TSMC, another semiconductor fabrication company, had disclosed a 40-nm process node. See EX1038, 3 (noting Altera had introduced commercial “40-nm devices” (e.g., “FPGAs and ASICs”) using TSMC’s 40-nm node); EX1043 (2008 TSMC press release noting 40-nm process had moved to “volume production”). The 40-nm process node achieved “2.35 times density improvement compared to devices built on 65nm process technology.” EX1038, 71. Thus, POSAs knew that 2.35 times as many transistors were implementable on a chip using the 40-nm node than using the 65-nm node, resulting in over 4 billion transistors on a chip at the 40-nm node. Franzon, ¶¶187-188.

array”), and Stuttard has no figure showing how performance and power scale with the number of PEs. Franzon, ¶113, ¶186.

Thus, even if claims 7-24 were entitled to the earliest claimed June 2009 priority date, by then POSAs would have reasonably expected to be able to make one integrated circuit on a chip with over 4 billion transistors using a 40-nm node, and even more transistors if they used the denser 32-nm and 22-nm process nodes that had by then been demonstrated in the prior art. Franzon, ¶189. For example, TSMC's 40-nm node achieved an "SRAM cell size at $0.242\mu\text{m}^2$ " (EX1038, 71), whereas a 32-nm node later in 2008 achieved a " $0.171\mu\text{m}^2$ [SRAM] cell size" (EX1045, 1) and a $0.157\mu\text{m}^2$ SRAM cell (EX1094, 67), which evidences additional scaling by a factor of 1.4 or 1.5 times over the 40-nm node. Franzon, ¶¶190-192. This would have resulted in over 6 billion transistors on a chip at the 32-nm node. Further in 2008, a 22-nm node demonstrated a " $0.1\mu\text{m}^2$ 6T-SRAM cell" (EX1115, 1), evidencing additional scaling by a factor of 2.42 times over TSMC's 40-nm SRAM cell, which would have resulted in nearly 10 billion transistors on a chip at the 22-nm node. Franzon, ¶¶192-193. Further, the 22-nm node continued later in 2009 (before the earliest date to which claims 7-24 could actually be entitled to claim priority—see *supra* §III.E) to achieve "double the density of the previous 32nm generation." EX1126, 1; Franzon, ¶¶194-195.

In the early 2000s, Stuttard's Applicant demonstrated to POSAs that a 64-poly-PE MTAP core could be fabricated on a chip using a "0.13-micron" (130-nm) process node, and that this core would require "40 million" transistors. EX1118, 1

(describing CS301 processor); Franzon, ¶196 (citing EX1107, 822 (showing CS301 had MTAP architecture similar to Stuttard's)). As discussed above, POSAs knew before June 2009 that over 4 billion transistors—which is 100 times as many as the 40 million used in the above-discussed 64-poly-PE MTAP core disclosed in EX1118—could be fabricated on a single chip using the 40-nm process node (and even more could be fabricated using a 32-nm or 22-nm process node). As transistors are the basic building blocks of digital circuitry, POSAs would have expected that 100 times as many transistors allows for fabricating approximately 100 times as many of a given component. Franzon, ¶¶196-197. Thus, POSAs would have reasonably expected success in implementing up to (at least) one hundred 64-poly-PE MTAP cores on a single chip—including, e.g., 6400 poly PEs and 100 mono execution units. Franzon, ¶197.

As Dr. Franzon further explains, this expectation is additionally confirmed by the POSA's understanding that the small die size of EX1118's 64-poly-PE implementation at a 130-nm could be scaled approximately ten times larger in 2008, coupled with an approximately factor-of-ten scaling ratio between the 130-nm and 40-nm nodes. Franzon, ¶¶198-201 (discussing EX1124, 74; EX1039, 18; EX1097, 101; EX1083, 71).

The POSA's expectation of success would have been further bolstered by their understanding that Stuttard-Shirazi's reduced-bitwidth floating-point units

would have likely required *less*, but certainly no more, chip area than the 32-bit floating-point units within the types of poly PEs described in the above-discussed EX1118 and EX1107 (at 822-23). Franzon, ¶¶197-198 (citing corroborating EX1106, 4 (“Many simple cores can be built within the same area as a small number of large complex cores.”); EX1014, 9 (“[R]eduction in core area [from lower-precision PEs] can be leveraged to increase the number of cores in the same silicon area.”), 1 (“[R]educ[ing] the size of floating point units (FPUs)[] allow[s] more of them to occupy the same silicon area.”)).

Thus, POSAs would have reasonably expected success in implementing the scaled Stuttard-Shirazi device to include well over 5000 poly PEs (having reduced-bitwidth FPUs), with multiple mono units (each having a full-precision (32-bit) FPU) on one chip by the time of the ’775 patent’s alleged invention. Franzon, ¶202.

Thus, before June 2009, POSAs would have reasonably expected success in fabricating the scaled Stuttard-Shirazi processor as one chip on which a 2048-PE Stuttard-Shirazi processor core was duplicated 3 or more times—or alternatively on which a 256-PE Stuttard-Shirazi processor core was duplicated 20 or more times—resulting in a chip having at least 6144 (or 5120) reduced-bitwidth PEs and a plurality of mono execution units (each with a full-precision (32-bit) FPU), as discussed *supra* §VI.A.2. Franzon, ¶202.

Indeed, the '775 patent identifies no challenges or special fabrication techniques required to achieve 5000+ PEs and a plurality of 32-bit multipliers on a chip, reinforcing that only ordinary skill was required for such integration. Franzon, ¶203. *Ex parte Sarkar*, Appeal No. 2021-003873, 2022 WL 337233, at *6 (PTAB Feb. 2, 2022) (“[POSAs’] reasonable expectation of success in making the proposed modification is supported by the fact that Appellant’s own Specification lacks any specific discussion of how hospitalization data and associated probabilities are incorporated into the Bayesian network.”) (quotation marks omitted) (citing *In re Epstein*, 32 F.3d 1559, 1568 (Fed. Cir. 1994)).

B. Claim 7

1. [7PRE] “A computing system, comprising:” [7A] “a host computer;” [7B] “a computing chip comprising:”

The scaled Stuttard-Shirazi system (§VI.A above) meets [7PRE], [7A], and [7B] for the same reasons discussed *supra* §§V.B.1-V.B.3 ([1PRE]-[1B]).

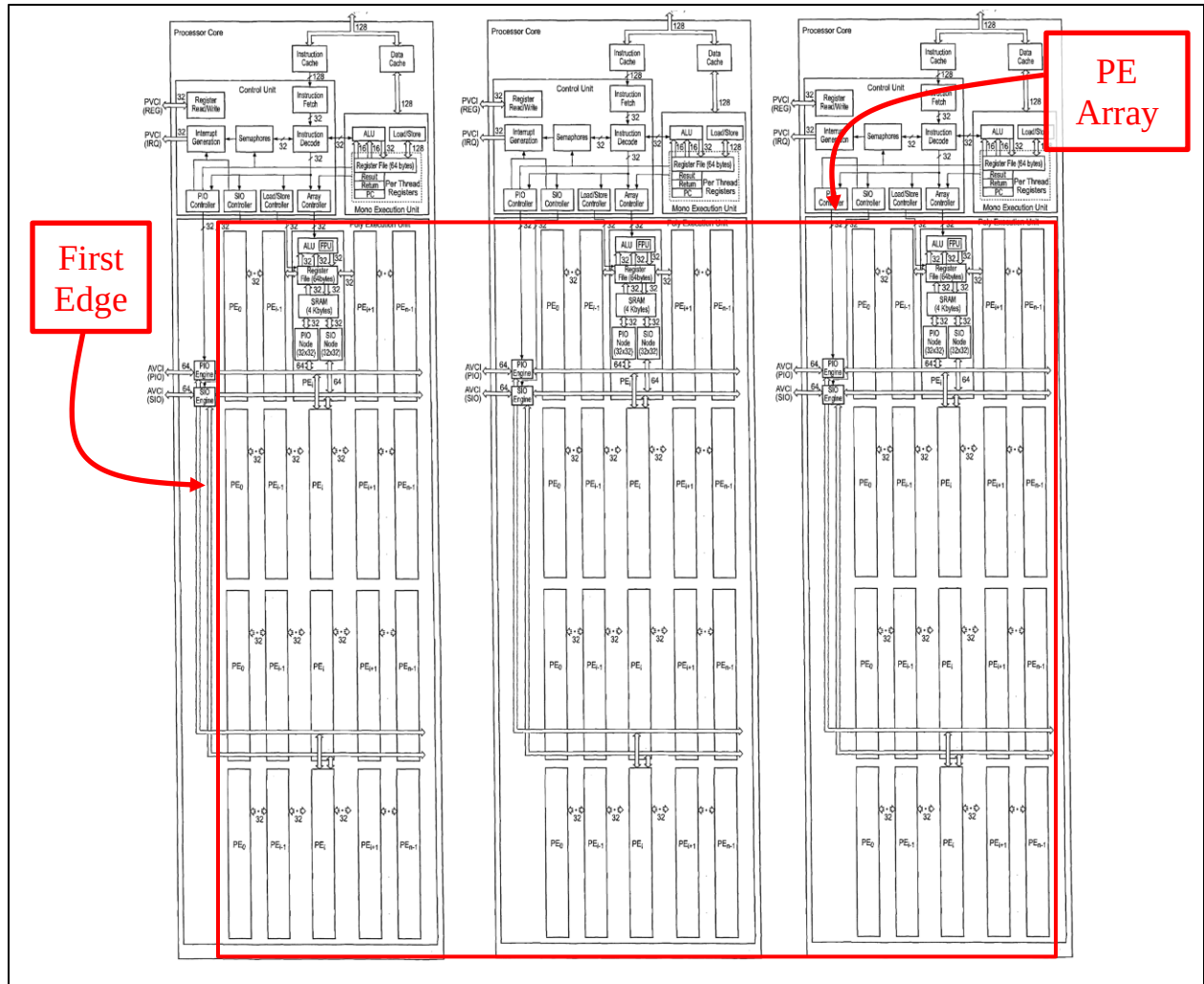
Franzon, ¶¶204-205.

2. **[7C]: “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;”**

As discussed *supra* §VI.A.1, [7C]’s “no less than 5000 in number” recitation confers no patentable weight. But even if given weight, the limitation is met because the scaled Stuttard-Shirazi chip (§VI.A above) comprises an array with, e.g., 6144 (or alternatively 5120) reduced-bitwidth PEs (*see supra* §VI.A.2), which is not less than 5000 in number, as claimed. Franzon, ¶206.

For the same reasons discussed *supra* §V.B.5 (concerning [1D] reciting first and second edge PEs and first second interior PEs), scaled Stuttard-Shirazi comprises a first subset of the 6144 (or 5120) PEs at a first edge of the PE array (the same edge identified *supra* §V.B.5) and a second subset of the PEs positioned in the interior of the PE array, as claimed. Franzon, ¶207.

A schematic illustration of one exemplary implementation of the scaled Stuttard-Shirazi processor (having three mono-unit PEs and $3 \times 2048 = 6144$ poly-unit PEs) is shown below, with the PE array outlined in red and an exemplary “first edge” indicated on its left-hand side. Franzon, ¶208.

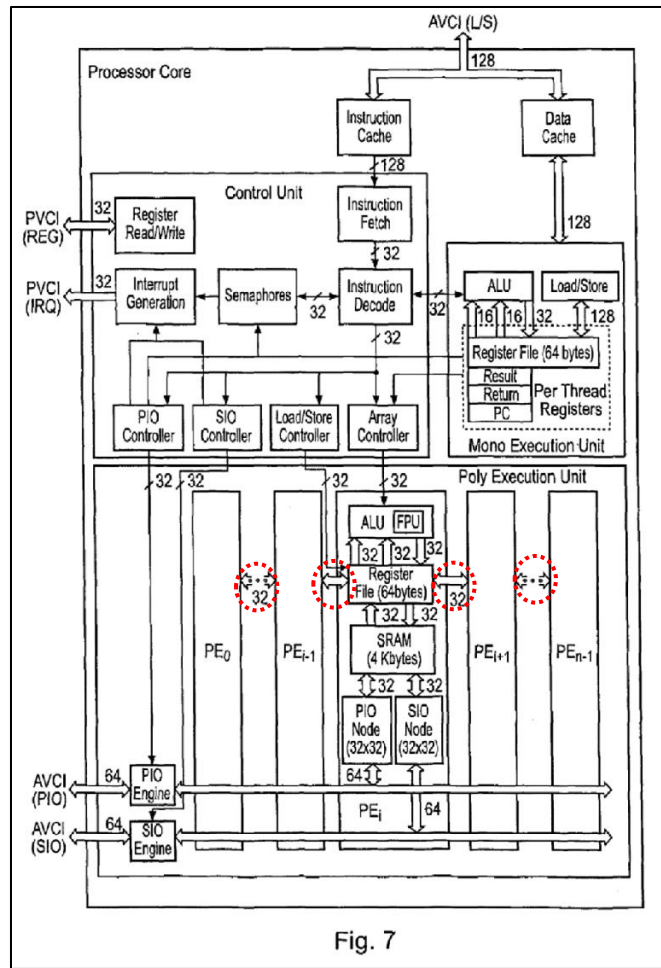


3. [7D]: “an input-output unit connected to each of the first subset of the plurality of first processing elements;”

For the same reason Stuttard-Shirazi’s bus (input-output unit) is connected to its first and second edge PEs (§V.B.6 above), scaled Stuttard-Shirazi’s bus (input-output unit) is connected to each MTAP core on the chip, including each of the first subset of the plurality of first PEs (§VI.B.2 above), as claimed. See Stuttard, 22:8-15, FIG. 8. Franzon, ¶109.

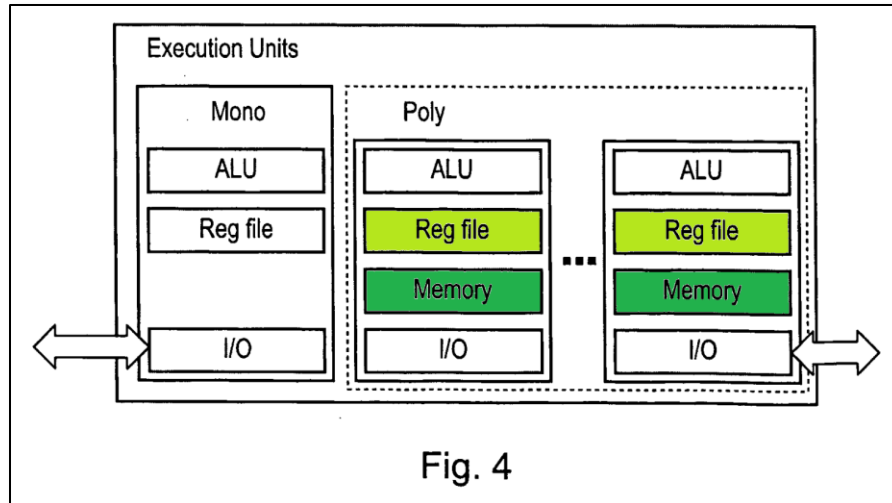
4. [7E]: “a plurality of processing element connections, each processing element connection connecting one of the plurality of first processing elements with another of the plurality of first processing elements, wherein each of the plurality of first processing elements is connected to at least one other of the plurality of first processing elements by at least one of the plurality of processing element connections;”

As discussed above (§VI.B.2), the plurality of first processing elements are the Stuttard-Shirazi poly PEs. Franzon, ¶210. Stuttard teaches that the poly PEs “communicate with one another via” a path “that **connects** the register file of each PE with the register files of its left and right neighbors.” Stuttard, 20:31-21:2. Stuttard’s Figure 7 shows connections between neighboring PEs in one poly unit as double-sided arrows (circled below), which are a plurality of PE connections, as claimed. Scaled Stuttard-Shirazi has the same connections in each of its poly units. Franzon, ¶210. As illustrated below, each PE is connected to at least one of its neighboring PEs by at least one of these PE connections, meeting [7E]. Franzon, ¶210.



5. [7F]: “a plurality of memory units, wherein each of the plurality of first processing elements is associated with a corresponding one of the plurality of memory units, and wherein each of the plurality of memory units is local to its associated one of the plurality of first processing elements;”

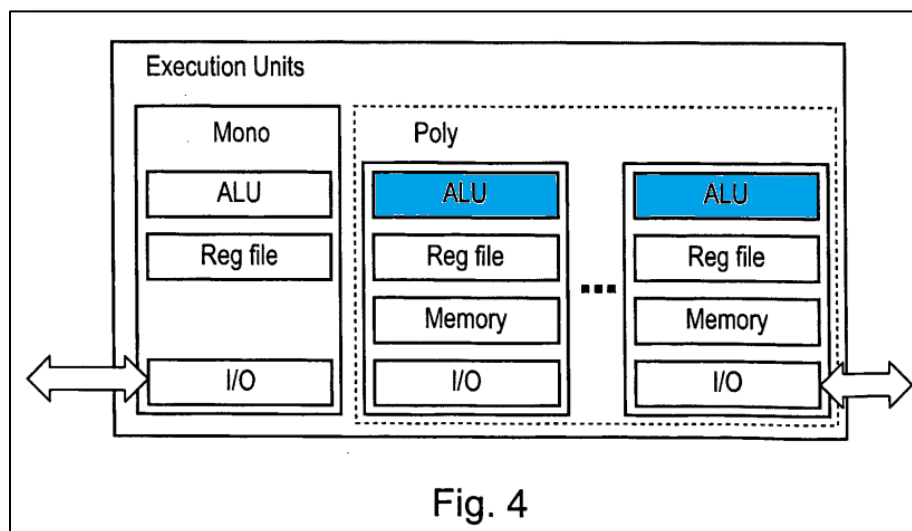
As discussed *supra* §V.B.7 and illustrated in Stuttard’s Figure 4 (annotated below), “[e]ach PE [] has its own independent **local memory**” and addressable register files, which are each a memory unit local to its associated PE, including each of the first PEs, as claimed. Stuttard, 10:14-19, 5:29-30, 7:14-16, 10:1-4, 17:12-23, FIGS. 3-4, 7; Franzon, ¶211.



6. [7G]: “a plurality of first arithmetic units, wherein each of the plurality of first processing elements has positioned therein at least one of the plurality of first arithmetic units;”

For the same reasons discussed *supra* §V.B.8 ([1H]), each poly PE in scaled Stuttard-Shirazi has positioned therein a first arithmetic unit, as claimed. Stuttard, 5:29-30, 7:14-16, 10:1-4, 10:14-19, 17:12-23, FIG. 3, 4 (ALUs annotated below).

Franzon, ¶212.



7. [7H]: “a plurality of second processing elements; and a plurality of second arithmetic units, wherein each of the plurality of second processing elements has positioned therein at least one of the plurality of second arithmetic units; and”

In Stuttard, “one PE [processing element] forms the mono execution unit.”

Stuttard, 10:5-9, 16:13-24, 3:15-18 (“[M]ono execution unit has a 32-bit ALU.”).

Scaled Stuttard-Shirazi (§VI.A above), includes a plurality (e.g., 3, or alternatively 20—*see supra* §VI.A.2) of mono execution units meeting the claimed “second processing elements.” Franzon, ¶213.

As Stuttard’s Figure 4 (annotated below) illustrates, each mono execution unit has positioned therein an ALU (*arithmetic logic unit*), meeting the claimed “second arithmetic units.” Stuttard, 3:15-18, 7:17-24, 16:13-24, FIGS. 3, 4, 7; Franzon, ¶214.

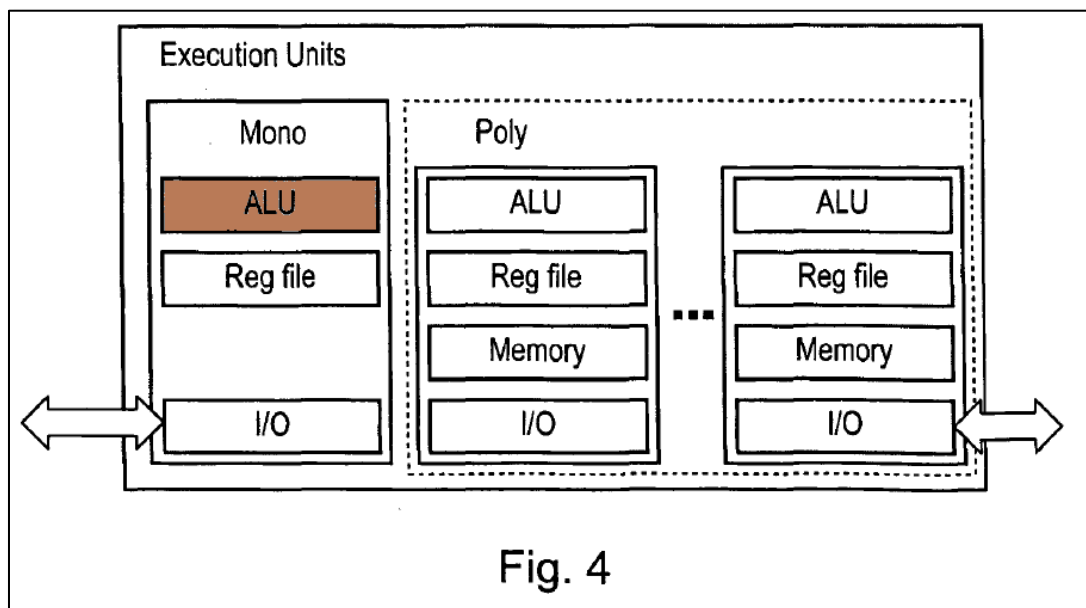


Fig. 4

8. **[7I]: “a host connection at least partially connecting the input-output unit with the host computer;”**

Scaled Stuttard-Shirazi has the same host connection and meets [7I] for the same reasons discussed *supra* §V.B.9 for identical limitation [1I]. Franzon, ¶215.

9. **[7J]: “wherein the plurality of first arithmetic units each comprises a first corresponding multiplier circuit adapted to receive” the Same Inputs As Recited in [1J]**

Scaled Stuttard-Shirazi meets [7J] for the same reasons Stuttard-Shirazi meets [1J] (§V.B.10 above). Franzon, ¶¶216-217.

10. **[7K]: “wherein the first multiplier circuits corresponding to the plurality of first arithmetic units each comprises a first respective plurality of transistors and has no other transistors;”**

Scaled Stuttard-Shirazi meets [7K] for the same reasons Stuttard-Shirazi meets [1L] (§V.B.12 above). Franzon, ¶¶218-219.

11. **[7L]: “wherein the plurality of second arithmetic units each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;”**

The plurality of second arithmetic units in Scaled Stuttard-Shirazi’s plurality of mono units meet [7L] for the same reasons the fifth arithmetic unit in Stuttard-Shirazi meets [1K] (§V.B.11 above). Franzon, ¶¶220-221.

12. [7M]: “wherein the second multiplier circuits corresponding to the plurality of second arithmetic units each comprises a second respective plurality of transistors;”

Each of Scaled Stuttard-Shirazi’s mono units’ 32-bit multiplier circuit comprises a second respective plurality of transistors for the same reasons Stuttard-Shirazi’s mono-unit multiplier circuit comprises a fifth plurality of transistors (§V.B.13 above) ([1M]). Franzon, ¶¶222-223.

13. [7N]: “wherein each of the second respective pluralities of transistors of the second multiplier circuits corresponding to the plurality of second arithmetic units exceeds in number each of the first respective pluralities of transistors of the first multiplier circuits corresponding to the plurality of first arithmetic units.”

Each second respective plurality of transistors in scaled-Stuttard-Shirazi exceeds in number each of the first respective pluralities of transistors for the same reasons Stuttard-Shirazi’s fifth plurality of transistors exceeds each of the first through fourth plurality of transistors for [1N] (§V.B.14 above). Franzon, ¶¶224-226.

C. Claim 8

1. [8A]: “other than the plurality of second processing elements, the computing chip has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and”

In Scaled Stuttard-Shirazi (§VI.A above), the only PEs that comprise a multiplier circuit adapted to receive as inputs “two floating-point values each of

width at least 32 bits” are the multiplier circuits in the plurality of mono units (claimed plurality of second PEs). §VI.B.11 above. The only other floating-point multiplier circuits on the chip are the multipliers in the poly units (claimed plurality of first PEs—§V.B.11, §VI.B.2 above), which are adapted to only receive Shirazi’s floating-point values of 16- or 18-bit width (§V.B.10, §VI.B.9). Thus, Scaled Stuttard-Shirazi meets [8A]. Franzon, ¶227.

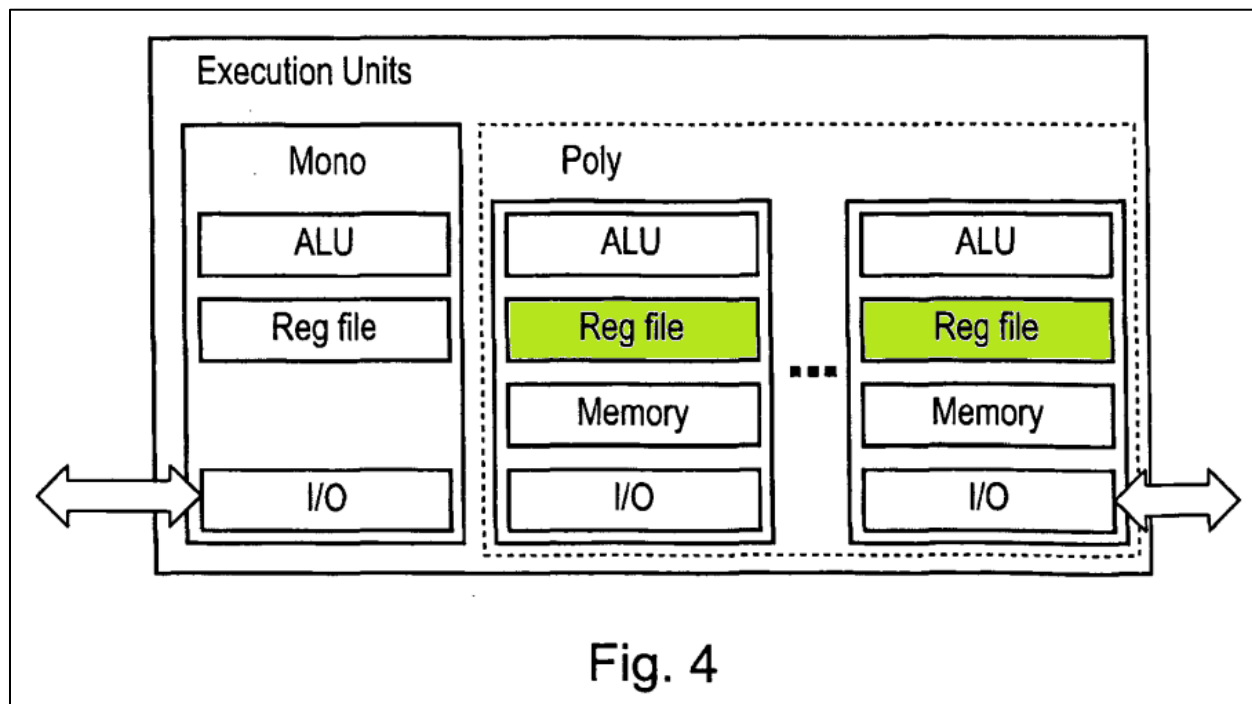
2. [8B]: “wherein the plurality of first processing elements exceeds in number, by at least 100, the plurality of second processing elements.”

In Scaled Stuttard-Shirazi (§VI.A above), the plurality of first PEs (in the poly units) number 6144 (or alternatively 5120) PEs (§VI.B.2 above), which exceeds in number, by at least 100, the three (or alternatively 20) PEs in the plurality of second PEs (in the mono units, §VI.B.7 above), meeting [8B]. Franzon, ¶¶228-229.

D. Claim 9: “the plurality of memory units each comprises a corresponding register.”

As Stuttard’s FIG. 4 (annotated below) illustrates, each PE in a poly unit comprises a “register file” (“Reg file”), which is “an array of bytes which can be addressed as registers” (Stuttard, 15:9-16), meeting claim 9. See §V.B.7 above (explaining register files are addressable “memory units”). Stuttard, 5:29-30 (“Each processing element of the SIMD array may comprise...a register file....”),

7:14-24, 10:1-4, 10:18-19, 15:21-25, 17:21-23, 20:31-21:12. Franzon, ¶¶230-231 (citing corroborating EX1058, [0003]; EX1075, 2:14-16).



E. Claim 10

Claim 10 recites identical limitations as claim 2, and is met by Scaled Stuttard-Shirazi for the same reasons Stuttard-Shirazi meets claim 2 (§V.C above). Franzon, ¶¶232-233.

F. Claim 11: “an aggregate maximum bandwidth of the plurality of processing element connections exceeds a maximum bandwidth of the host connection.”

Stuttard teaches that a 64-PE array, running at 400 MHz, has an Inter-PE Bandwidth of 50 Gigabytes/sec (i.e., aggregate bandwidth of 3,200 Gigabytes/sec). Stuttard, 10:15-19; Franzon, ¶234. Scaled Stuttard-Shirazi has at least 80 times as

many parallel PEs as a 64-PE array, and thus at least 80 times the aggregate bandwidth as un-scaled Stuttard (i.e., 200+ Terabytes/sec). Franzon, ¶234.

As discussed above (§VI.B.8), Scaled Stuttard-Shirazi's host connection is provided by "external ports" connecting the input-output unit's bus to the host system. Stuttard discloses that more details for the bus unit can be found in "copending patent application GB 0321186.9." Stuttard, 8:7-8. GB 0321186.9 (EX1051) was published as WO2002/065700 (EX1049). See EX1051, codes (21), (87). EX1049 (at 39:8-21) disclosed that a ClearConnect bus for a processor like Stuttard's had two datapaths with "raw bandwidth [of] 51.2 Gbits/sec" and another with "raw bandwidth [of] 102.4 Gbits/sec," which are orders of magnitude below Scaled Stuttard-Shirazi's 200+ Terabytes/sec aggregate bandwidth for the poly PEs discussed above. Franzon, ¶235.

POSAs would have understood that the maximum bandwidth of the host connection would be no wider than (and likely significantly narrower than) the maximum bandwidth of the ClearConnect bus itself, and thus the aggregate maximum bandwidth of the poly PEs in Scaled Stuttard-Shirazi exceeds the maximum bandwidth of the host connection, as claimed. Franzon, ¶¶236-237 (citing corroborating EX1120, 85-86 (noting the "relatively low bandwidth available between the [host] CPU and [accelerator] GPU" in accelerator systems with "SIMD[] capabilities").

G. Claims 12-14

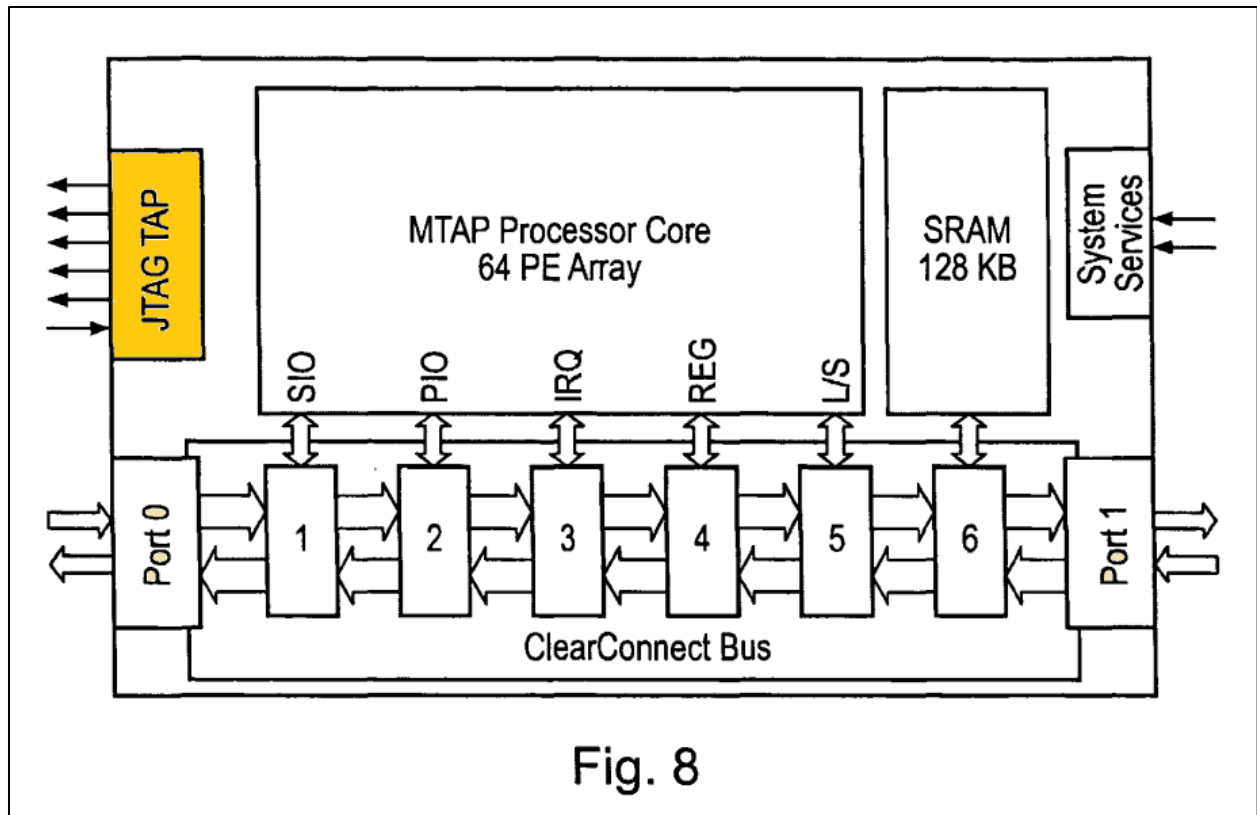
Claims 12-14 recite identical limitations as claims 3-5, respectively. Scaled Stuttard-Shirazi meets claim 12-14 for the same reasons Stuttard-Shirazi meets claims 3-5, respectively (§§V.D-V.F). Franzon, ¶¶238-243.

H. Claim 16

1. [16PRE]: “A circuit board connected to a host computer, the circuit board comprising:”

POSAs had reasons to mount two Scaled Stuttard-Shirazi computing chips on a circuit board connected to the host computer.

First, Stuttard taught mounting its MTAP chips to a circuit board. Specifically, Stuttard’s “evaluation chip EVI” is a “simple embodiment of” Stuttard’s invention that used the ClearConnect “bus to interface various ports of the processor...to external pins.” Stuttard, 22:4-7. POSAs understood that it was conventional for “external pins” to form an electrical connection between I/O ports and a circuit board. Franzon, ¶¶244-245 (citing corroborating EX1096, 1:16-28; EX1089, 2-3). Stuttard’s Figure 8 (annotated below) shows that the evaluation chip had a “JTAG TAP,” which POSAs understood is an IEEE standard for testing “interconnections...after installation on a **printed circuit board**.” Franzon, ¶246 (quoting corroborating EX1059, [0001]-[0002]).



Thus, Stuttard taught connecting an MTAP chip to a circuit board. Franzon, ¶246.

Second, Stuttard taught that its system bus allowed a “connect[ion] from *one chip to another*,” which POSAs knew would have conventionally been implemented by mounting the multiple chips on a common circuit board. Stuttard, 22:12-15; Franzon, ¶247 (citing corroborating EX1108, 34 (noting “ClearSpeed Advance board having two CSX600 chips”), EX1107, 822-23, FIG. 2 (“[B]oard includes 2 SIMD chips.”)).

Third, it was background knowledge for POSAs that incorporating multiple SIMD processors, like Scaled Stuttard-Shirazi, on a circuit board connected to a

host computer would have been using a known technique (circuit boards) to improve similar devices (co-processors) in the same way (facilitating incorporation of the co-processors into a system with a host computer). *KSR*, 550 U.S. at 417. Franzon, ¶248 (citing corroborating EX1070, [0001]-[0003]).

In this implementation, two Scaled Stuttard-Shirazi chips are mounted to a common circuit board connected to the host computer using a conventional connection between a host computer and circuit board (e.g., a PCI slot), with all other aspects of each Scaled Stuttard-Shirazi chip remaining unchanged. Franzon, ¶249.

POSAs would have had a reasonable expectation of success in this implementation because, as discussed above, it was conventional to incorporate multiple accelerator computing chips, like Scaled Stuttard-Shirazi, onto circuit boards connected to a host computer. Franzon, ¶250.

2. [16A]: “a plurality of processing element arrays, each comprising:”

The implementation discussed *supra* §VI.H.1 meets [16A] because it comprises two Scaled Stuttard-Shirazi chips on one circuit board, each chip having a processing element array, as discussed for [7C] (§VI.B.2 above). Franzon, ¶251.

3. [16B]-[16H]

Limitations [16B]-[16H] are identical to corresponding limitations in claim 7. The implementation discussed *supra* §VI.H.1 including two Scaled Stuttard-Shirazi chips on one board meets these limitations for the same reasons one of the Scaled Stuttard-Shirazi chips meets the corresponding limitations in claim 7, as set forth in the table below:

Claim 16 Limitation:	Identical to Claim 7 Limitation:	Addressed in Above Section(s):
[16B]	[7C]	§VI.B.2, §V.B.5
[16C]	[7E]	§VI.B.4, §V.B.4
[16D]	[7F]	§VI.B.5, §V.B.7
[16E]	[7G]	§VI.B.6, §V.B.8
[16F]	[7J]	§VI.B.9, §V.B.10
[16G]	[7D]	§VI.B.3, §V.B.6
[16H]	[7I]	§VI.B.8, §V.B.9

Franzon, ¶¶252-253.

I. Claims 17-19

Claims 17-19 depend from claim 16 and add to claim 16 the same limitations that claim 8 adds to claim 7, plus that the total count of first PEs exceeds the second PEs in number by:

- “at least 20 more than three times” (claim 17);

- “at least 1000 more than five times” (claim 18); and
- “at least 100 more” (claim 19).

Franzon, ¶¶254-255.

The limitations identical to claim 8 are met by the mono units in the Scaled Stuttard-Shirazi chips as discussed *supra* §VI.C. The additional limitations above are met because, as addressed *supra* §VI.A.2, each Scaled Stuttard-Shirazi chip on the circuit board comprises 6144 (or 5120) first PEs (in the poly units) and three (or 20) second PEs (in the mono units). Two Scaled Stuttard-Shirazi chips on the circuit board therefore collectively have 12,288 (or 10,240) first PEs vs. six (or 40) second PEs, which meets each of the “total count” limitations of claims 17-19.

Franzon, ¶¶255-256.

J. Claims 20-23

Claims 20, 21, 22, and 23 recite the identical limitations as claims 10, 12, 13, and 14, respectively. The implementation discussed *supra* §VI.H.1 with two Scaled Stuttard-Shirazi chips on a board meets claims 20, 21, 22, and 23 for the same reasons Scaled-Stuttard-Shirazi meets claims 10, 12, 13, and 14, discussed *supra* §VI.E (claim 10), §VI.G (claims 12-14) respectively. Franzon, ¶¶257-259.

VII. PUBLIC ACCESSIBILITY OF CITED PRINTED PUBLICATIONS

A. Prior to June 2008

The exhibits discussed in this §VII.A are §102(b) prior art even if all challenged claims were entitled to the June 2009 claimed priority date.

The exhibits below bear copyrights indicating they were published before June 2008 by the ACM and/or IEEE, which are “well-known, reputable compiler[s] and publisher[s] of scientific and technical publications.” *Ericsson Inc. v. Intellectual Ventures I LLC*, IPR2014-00527, Paper 41, 10-11 (May 18, 2015); *Hulu, LLC v. Sound View Innovations, LLC*, IPR2018-01039, Paper 29, 19 (Dec. 20, 2019); *Microsoft Corp. v. IPA Techs. Inc.*, IPR2019-00810, Paper 44, 159 (Oct. 15, 2020).

- EX1012, 232 (“Copyright 2007 ACM”);
- EX1034, 273 (“© 2000 IEEE”);
- EX1036 (“Shirazi”), 155 (“© 1995 IEEE”);
- EX1105, 206 (“© 1998 IEEE”);
- EX1100, 759 (“© 2006 IEEE”);
- EX1107, 822 (“© 2005 IEEE”), iii (2005 Library of Congress Copyright Office stamp);

- EX1111—see Franzon, Appendix §II.C.31 (explaining EX1112, 4 (“©2006 by the” IEEE), 3 (2006 Library of Congress Copyright Office stamp) shows copyright information for EX1111);
- EX1120, 2 (“© 2005 By The” IEEE), 1 (2005 library stamp);
- EX1125, 170 (“© 2000 IEEE”).

Further, testimony from IEEE representatives demonstrates public accessibility of the following publications as of the corresponding years below:

- EX1013 (2007)—see EX1127, ¶¶6-12;
- EX1034 (2000)—see EX1035, ¶¶6-12;
- EX1036 (“Shirazi”) (1995)—see EX1037, ¶¶6-12;
- EX1092 (2003)—see EX1093, ¶¶6-12;
- EX1094 (by June 19, 2008)—see EX1130, ¶¶6-12;
- EX1097 (2001)—see EX1131, ¶¶6-12;
- EX1100 (2006)—see EX1133, ¶¶6-12;
- EX1125 (2000)—see EX1136, ¶¶6-12.

See also Bolin, ¶¶67-80 (discussing EX1012-EX1013); Franzon, Appendix §II.C.1, §II.C.2, §II.C.43, §II.C.44, §II.C.45, §II.C.20, §II.C.21, §II.C.26, §II.C.28, §II.C.23, §II.C.31, §II.C.38, §II.C.40 (discussing EX1012, EX1013, EX1034, EX1036, EX1092, EX1094, EX1097, EX1105, EX1107, EX1100, EX1111, EX1120, EX1125, respectively).

EX1098 was published in 2007. EX1098 (at 425) bears “©2007 by the IETE,” indicating it was published in 2007 by the IETE, considered to be the IEEE of India. EX1098 (at 1, 383) also bears 2007 library stamps indicating it was publicly accessible in a library. Franzon, Appendix §II.C.22.

EX1088 bears (on every page) “Copyright 2000 – 2001 by the VSI Alliance, Inc.” and was listed and available for purchase on the VSI Alliance’s website by 2002. Franzon, Appendix §II.C.19; EX1047, 6-8; EX1048, 100-106.

EX1014 was publicly accessible from an author’s website in 2008. Franzon, Appendix §II.C.3; EX1032, 4-44; *see also* EX1013 (discussed above; citing EX1014 in 2007).

EX1106 (at 12) bears “Copyright © 2006 Intel Corporation” and additionally was publicly accessible from Intel’s website by 2007. Franzon, Appendix §II.C.27; EX1047, 66-78.

EX1117 bears “Copyright 2007 ACM,” indicating it was published in 2007 by the ACM, whose publications “are distributed widely and intended to be accessible to the public.” *Microsoft*, IPR2019-00810, Paper 44, 81. EX1117 was also publicly accessible from ACM’s Digital Library in 2007. Bolin, ¶¶65-99; Franzon, Appendix §II.C.36.

EX1118 was publicly accessible from arstechnica.com in 2004. Franzon, Appendix §II.C.37; EX1123, 5.

B. Prior to June 19, 2009

The exhibits discussed in this §VII.B are at least §102(a) prior art to all challenged claims, and are §102(b) prior art to at least claims 7-24 whose priority date is no earlier than June 2010 (*supra* §III.E).

EX1038 (at 72) bears “© 2008 Altera Corporation” and “June 2008” and was additionally publicly accessible from Altera’s website by December 2008. Franzon, Appendix §II.C.9; EX1047, 81-153; EX1048, 108-13.

The exhibits below bear copyrights on the cited pages, indicating they were published before June 2009 by the IEEE.

- EX1039, 18 (“© 2008 IEEE”), *see also* pdf pp. 1, 4 (January 2009 library stamp);
- EX1094, 67 (“©2008 IEEE”);
- EX1103, 1011 (“©2008 IEEE”);
- EX1104, 81 (“© 2008 IEEE”);
- EX1110, 607 (“© 2008 IEEE”);
- EX1113, 539 (“© 2008 IEEE”).

See also Franzon, Appendix §II.C.10, §II.C.20, §II.C.24, §II.C.25, §II.C.30, §II.C.32.

An IEEE representative testified that the IEEE made EX1039, EX1045, and EX1115 publicly accessible in 2008. EX1128, ¶¶6-12; EX1129, ¶¶6-12; EX1135, ¶¶6-12; Franzon, Appendix §II.C.10, §II.C.16, §II.C.34.

EX1043 bears “Issued on: 2008/11/17” and was publicly available accessible from TSMC’s website by that date. Franzon, Appendix §II.C.14.

EX1108 (at 32) and EX1114 (at 3, 89) bear copyrights indicating they were published before June 2009 by Springer, “a well-known, reputable compiler of scientific and technical publications.” Bolin, ¶¶62. These exhibits were also publicly accessible from Springer’s online database (“SpringerLink”) before June 2009. Bolin, ¶¶60-65, ¶¶105-111.

EX1124 bears a copyright indicating it was published in 2009 by Springer, and it was publicly accessible from SpringerLink before June 19, 2009. Bolin, ¶¶81-85.

See also Franzon, Appendix §II.C.29, §II.C.33, §II.C.39 (discussing EX1108, EX1114, EX1124, respectively).

The Semantic Scholar database indicates EX1116 was published in 2008, and an April 2009 article cites EX1116. Franzon, Appendix §II.C.35.

C. Prior to June 2010

EX1126 is §102(a) prior art to at least claims 7-24 because their priority date is no earlier than June 2010 (§III.E above).

EX1126 bears indicia of publication by Intel Corporation in connection with Intel's September 2009 developer forum. EX1126 was additionally publicly accessible from Intel's website by March 2010. Franzon, Appendix §II.C.41; EX1047, 79-80; EX1048, 83-98.

VIII. SECONDARY CONSIDERATIONS

In IPRs on related patents, Singular alleged evidence of secondary considerations, which the Board found was not probative of nonobviousness. Singular "d[id] not show skepticism as to whether or how a problem could be solved;" its "alleged industry praise [wa]s too ambiguous and high-level;" it "d[id] not establish a difference between the alleged [unexpected] results and the closest prior art;" its alleged commercial success evidence was "too speculative;" and it failed to demonstrate nexus with accused products. IPR2021-00155, Paper 62 at 38-51. The same failings would apply if Singular raised the same evidence and arguments for the '775 patent.

If Singular seeks in later filings to raise secondary considerations evidence/arguments directed to the '775 patent, the Board should not consider such evidence/arguments where "Petitioner has not had an opportunity to respond." *Sony Corp. v. Creative Tech.*, IPR2016-01407, Paper 13, 24 (Dec. 6, 2016).

IX. NO BASIS EXISTS FOR DISCRETIONARY DENIAL

No prior proceeding has asserted invalidity of the '775 patent's claims,⁴ and Singular represented to Google that it will soon voluntarily dismiss (without prejudice) the only litigation asserting the '775 patent, which is in its infancy. This Petition's primary reference (Stuttard) was not previously presented to the Office, nor is it cumulative. Therefore, there is no basis for discretionary denial under §314(a) or §325(d).

X. CONCLUSION

The Board should institute review and cancel claims 1-5, 7-14, and 16-23.

Dated: December 22, 2022

Respectfully submitted,
Google LLC

/Elisabeth Hunt/
Elisabeth H. Hunt, Reg. No. 67,336
WOLF, GREENFIELD & SACKS, P.C.

⁴ Petitioner's concurrent IPR petition is addressed in a separate Ranking paper filed herewith.

XI. APPENDIX: CLAIM LISTING

The following claim listing assigns element labels (e.g., [1PRE], [1A], etc.) to certain claims for convenience of reference.

Claim 1
[1PRE] A computing system, comprising:
[1A] a host computer;
[1B] a computing chip comprising:
[1C] a processing element array comprising a first edge processing element positioned at a first edge of the processing element array, a second edge processing element positioned at the first edge of the processing element array, a first interior processing element positioned at a first location in the interior of the processing element array, and a second interior processing element positioned at a second location in the interior of the processing element array;
[1D] a first processing element connection connecting the first edge processing element with the first interior processing element; a second processing element connection connecting the second edge processing element with the second interior processing element;
[1E] an input-output unit connected to the first edge processing element and the second edge processing element;
[1F] a first memory local to the first edge processing element; a second memory local to the second edge processing element; a third memory local to the first interior processing element; a fourth memory local to the second interior processing element; and,
[1G] a fifth arithmetic unit;
[1H] wherein the first edge processing element comprises a first arithmetic unit; wherein the second edge processing element comprises a second arithmetic unit; wherein the first interior processing element comprises a third arithmetic unit; and wherein the second interior processing element comprises a fourth arithmetic unit; and,
[1I] a host connection at least partially connecting the input-output unit with the host computer;

[1J] wherein the first, second, third and fourth arithmetic units each comprises a corresponding multiplier circuit adapted to receive as a first input to the corresponding multiplier circuit a first floating point value having a first binary mantissa of width no more than 11 bits and a first binary exponent of width at least 6 bits, and to receive as a second input to the corresponding multiplier circuit a second floating point value having a second binary mantissa of width no more than 11 bits and a second binary exponent of width at least 6 bits;
[1K] wherein the fifth arithmetic unit comprises a corresponding multiplier circuit adapted to receive as inputs to the corresponding multiplier circuit two floating point values each of width at least 32 bits;
[1L] wherein the multiplier circuit corresponding to the first arithmetic unit comprises a first plurality of transistors and has no other transistors, the multiplier circuit corresponding to the second arithmetic unit comprises a second plurality of transistors and has no other transistors, the multiplier circuit corresponding to the third arithmetic unit comprises a third plurality of transistors and has no other transistors, the multiplier circuit corresponding to the fourth arithmetic unit comprises a fourth plurality of transistors and has no other transistors, and
[1M] the multiplier circuit corresponding to the fifth arithmetic unit comprises a fifth plurality of transistors; and,
[1N] wherein the fifth plurality of transistors exceeds in number each of the first plurality of transistors, the second plurality of transistors, the third plurality of transistors, and the fourth plurality of transistors.
Claim 2
The computing system of claim 1, wherein said host computer is programmed to provide instructions to said computing chip that, when executed, cause said processing element array to perform an operation whose output is used to identify at least one image, from a plurality of images to be searched, that is similar to at least one input image.
Claim 3
[3PRE] The computing system of claim 1,
[3A] the computing chip further comprises a control unit connected to the processing element array,
[3B] the control unit comprising circuitry adapted to decode at least one instruction received from the host computer via the input-output unit, and
[3C] to send at least one control signal to the processing element array to cause the processing element array to operate according to the at least one instruction.

Claim 4
The computing system of claim 3, wherein the computing chip further comprises an instruction memory adapted to store the at least one instruction received from the host computer via the input-output unit, wherein the control unit is further adapted to retrieve the at least one instruction from the instruction memory.
Claim 5
The computing system of claim 1, wherein the width of each first binary mantissa is no more than 10 bits plus a first sign bit, and wherein the width of each second binary mantissa is no more than 10 bits plus a second sign bit.
Claim 6
The computing system of claim 5, wherein the width of each first binary exponent is at least 5 bits plus a third sign bit and wherein the width of each second binary exponent is at least 5 bits plus a fourth sign bit.
Claim 7
[7PRE] A computing system, comprising:
[7A] a host computer;
[7B] a computing chip comprising:
[7C] 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;
[7D] an input-output unit connected to each of the first subset of the plurality of first processing elements;
[7E] a plurality of processing element connections, each processing element connection connecting one of the plurality of first processing elements with another of the plurality of first processing elements, wherein each of the plurality of first processing elements is connected to at least one other of the plurality of first processing elements by at least one of the plurality of processing element connections;
[7F] a plurality of memory units, wherein each of the plurality of first processing elements is associated with a corresponding one of the plurality of memory units, and wherein each of the plurality of memory units is local to its associated one of the plurality of first processing elements;
[7G] a plurality of first arithmetic units, wherein each of the plurality of first processing elements has positioned therein at least one of the plurality of first arithmetic units;

[7H] a plurality of second processing elements; and a plurality of second arithmetic units, wherein each of the plurality of second processing elements has positioned therein at least one of the plurality of second arithmetic units; and
[7I] a host connection at least partially connecting the input-output unit with the host computer;
[7J] wherein the plurality of first arithmetic units each comprises a first corresponding multiplier circuit adapted to receive as a first input to the first corresponding multiplier circuit a first floating point value having a first binary mantissa of width no more than 11 bits and a first binary exponent of width at least 6 bits, and to receive as a second input to the first corresponding multiplier circuit a second floating point value having a second binary mantissa of width no more than 11 bits and a second binary exponent of width at least 6 bits;
[7K] wherein the first multiplier circuits corresponding to the plurality of first arithmetic units each comprises a first respective plurality of transistors and has no other transistors;
[7L] wherein the plurality of second arithmetic units each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;
[7M] wherein the second multiplier circuits corresponding to the plurality of second arithmetic units each comprises a second respective plurality of transistors;
[7N] wherein each of the second respective pluralities of transistors of the second multiplier circuits corresponding to the plurality of second arithmetic units exceeds in number each of the first respective pluralities of transistors of the first multiplier circuits corresponding to the plurality of first arithmetic units.
Claim 8
[8PRE] The computing system of claim 7, wherein,
[8A] other than the plurality of second processing elements, the computing chip has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and
[8B] wherein the plurality of first processing elements exceeds in number, by at least 100, the plurality of second processing elements.
Claim 9
The computing system of claim 7, wherein the plurality of memory units each comprises a corresponding register.

Claim 10
The computing system of claim 8, wherein said host computer is programmed to provide instructions to said computing chip that, when executed, cause said processing element array to perform an operation whose output is used to identify at least one image, from a plurality of images to be searched, that is similar to at least one input image.
Claim 11
The computing system of claim 8, wherein an aggregate maximum bandwidth of the plurality of processing element connections exceeds a maximum bandwidth of the host connection.
Claim 12
The computing system of claim 11, wherein the computing chip further comprises a control unit connected to the processing element array, the control unit comprising circuitry adapted to decode at least one instruction received from the host computer via the input-output unit, and to send at least one control signal to the processing element array to cause the processing element array to operate according to the at least one instruction.
Claim 13
The computing system of claim 12, wherein the computing chip further comprises an instruction memory adapted to store the at least one instruction received from the host computer via the input-output unit, wherein the control unit is further adapted to retrieve the at least one instruction from the instruction memory.
Claim 14
The computing system of claim 7, wherein the width of each first binary mantissa is no more than 10 bits plus a first sign bit, and wherein the width of each second binary mantissa is no more than 10 bits plus a second sign bit.
Claim 15
The computing system of claim 14, wherein the width of each first binary exponent is at least 5 bits plus a third sign bit and wherein the width of each second binary exponent is at least 5 bits plus a fourth sign bit.
Claim 16
[16PRE] A circuit board connected to a host computer, the circuit board comprising:
[16A] a plurality of processing element arrays, each comprising:

[16B] a plurality of first processing elements, wherein the plurality of first processing elements of said processing element array is no less than 5000 in number, wherein each of a first subset of the plurality of first processing elements of said processing element array is positioned at a first edge of said processing element array, and wherein each of a second subset of the plurality of first processing elements of said processing element array is positioned in the interior of said processing element array;
[16C] a plurality of processing element connections, each processing element connection connecting one of the plurality of first processing elements of said processing element array with another of the plurality of first processing elements of said processing element array, wherein each of the plurality of first processing elements of said processing element array is connected to at least one other of the plurality of first processing elements of said processing element array by at least one of the plurality of processing element connections of said processing element array;
[16D] a plurality of memory units, wherein each of the plurality of first processing elements of said processing element array is associated with a corresponding one of the plurality of memory units of said processing element array, and wherein each of the plurality of memory units of said processing element array is local to its associated one of the plurality of first processing elements of said processing element array; and,
[16E] a plurality of arithmetic units, wherein each of the plurality of first processing elements of said processing element array has positioned therein at least one of the plurality of arithmetic units of said processing element array;
[16F] wherein the plurality of arithmetic units of said processing element array each comprises a first corresponding multiplier circuit adapted to receive as a first input to the first corresponding multiplier circuit a first floating point value having a first binary mantissa of width no more than 11 bits and a first binary exponent of width at least 6 bits, and to receive as a second input to the first corresponding multiplier circuit a second floating point value having a second binary mantissa of width no more than 11 bits and a second binary exponent of width at least 6 bits; and,
[16G] an input-output unit connected to at least one of the plurality of processing element arrays; and,
[16H] a host connection at least partially connecting the input-output unit with the host computer.

Claim 17
[17A] The circuit board of claim 16, wherein the circuit board further comprises a plurality of second processing elements, wherein the plurality of second processing elements each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;
[17B] wherein other than the second processing elements, the circuit board has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and,
[17C] wherein a total count of the pluralities of first processing elements of the plurality of processing element arrays exceeds in number, by at least 20 more than three times, the plurality of second processing elements.
Claim 18
[18A] The circuit board of claim 16, wherein the circuit board further comprises a plurality of second processing elements, wherein the plurality of second processing elements each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;
[18B] wherein other than the second processing elements the circuit board has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and,
[18C] wherein a total count of the pluralities of first processing elements of the plurality of processing element arrays exceeds in number, by at least 1000 more than five times, the plurality of second processing elements.
Claim 19
[19A] The circuit board of claim 16, wherein the circuit board further comprises a plurality of second processing elements, wherein the plurality of second processing elements each comprises a second corresponding multiplier circuit adapted to receive as inputs to the second corresponding multiplier circuit two floating point values each of width at least 32 bits;
[19B] wherein other than the second processing elements the circuit board has no other processing element that comprises a multiplier circuit adapted to receive as inputs to the multiplier circuit two floating point values each of width at least 32 bits; and,
[19C] wherein a total count of the pluralities of first processing elements of the plurality of processing element arrays exceeds in number, by at least 100 more, the plurality of second processing elements.

Claim 20
The circuit board of claim 18, wherein said circuit board is adapted to receive instructions from the host computer that, when executed, cause at least one of said processing element arrays to perform an operation whose output is used to identify at least one image, from a plurality of images to be searched, that is similar to at least one input image.
Claim 21
The circuit board of claim 18, wherein the circuit board further comprises a control unit connected to at least one of the plurality of processing element arrays, the control unit comprising circuitry adapted to decode at least one instruction received from the host computer via the input-output unit, and to send at least one control signal to the at least one of the plurality of processing element arrays to cause the at least one of the plurality of processing element arrays to operate according to the at least one instruction.
Claim 22
[22A] The circuit board of claim 21, wherein the circuit board further comprises an instruction memory connected to the control unit, the instruction memory comprising circuitry adapted to store the at least one instruction received from the host computer via the input-output unit, and
[22B] wherein the control unit further comprises circuitry adapted to retrieve the at least one instruction from the instruction memory.
Claim 23
The circuit board of claim 16, wherein the width of each first binary mantissa is no more than 10 bits plus a first sign bit, and wherein the width of each second binary mantissa is no more than 10 bits plus a second sign bit.
Claim 24
The circuit board of claim 23, wherein the width of each first binary exponent is at least 5 bits plus a third sign bit and wherein the width of each second binary exponent is at least 5 bits plus a fourth sign bit.

CERTIFICATE OF SERVICE UNDER 37 C.F.R. § 42.6 (E)(4)

I certify that on December 22, 2022, I will cause a copy of the foregoing document, including any exhibits or appendices filed therewith, to be served via Priority Express Mail at the following correspondence address of record for the patent:

Blueshift IP LLC
1 Broadway, 14th Floor
Cambridge, MA 02142

Date: December 22, 2022

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C.

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24, the undersigned certifies that the foregoing Petition for *Inter Partes* Review contains 13,937 words excluding a table of contents, a table of authorities, Mandatory Notices under § 42.8, a certificate of service or word count, or appendix of exhibits or claim listing. Petitioner has relied on the word count feature of the word processing system used to create this paper in making this certification.

Date: December 22, 2022

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C.