Trials@uspto.gov
571-272-7822

Paper 72
Date: April 9, 2020

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

FG SRC LLC,[1]
Patent Owner.

IPR2018-01601[2]
Patent 7,225,324 B2

Before KALYAN K. DESHPANDE, JUSTIN T. ARBES, and CHRISTA P. ZADO, *Administrative Patent Judges.*

ARBES, *Administrative Patent Judge*.

JUDGMENT
Final Written Decision
Determining All Challenged Claims Unpatentable
35 U.S.C. § 318(a)

---

[1] Patent Owner filed updated mandatory notice information indicating that DirectStream, LLC ("DirectStream") assigned the challenged patent to FG SRC LLC. Paper 69, 1. Accordingly, the caption for this proceeding has been changed.

[2] Cases IPR2018-01602 and IPR2018-01603 have been consolidated with this proceeding.

I. INTRODUCTION

A. Background and Summary

Petitioner Microsoft Corporation filed three Petitions, collectively requesting *inter partes* review of claims 1–5, 7–9, 15, 17, 18, and 20–24 of U.S. Patent No. 7,225,324 B2 (Ex. 1001, "the '324 patent") pursuant to 35 U.S.C. § 311(a), as listed in the following chart.[3]

Case Number	Challenged Claims	Petition
IPR2018-01601	1, 8, 9, and 20	Paper 1 ("Pet.")
IPR2018-01602	1, 7, 15, 17, and 24	Paper 1 ("-1602 Pet.")
IPR2018-01603	1–5, 18, and 21–23	Paper 1 ("-1603 Pet.")

On April 12, 2019, we instituted an *inter partes* review as to all challenged claims on all grounds of unpatentability asserted in the Petitions, and exercised our authority under 35 U.S.C. § 315(d) to consolidate the three proceedings and conduct the proceedings as one trial. Paper 21 ("Decision on Institution" or "Dec. on Inst."). Patent Owner FG SRC LLC subsequently filed a Patent Owner Response (Paper 36, "PO Resp."), Petitioner filed a Reply (Paper 49, "Reply"), and Patent Owner filed a Sur-Reply (Paper 59, "Sur-Reply"). Petitioner filed a Motion to Exclude (Paper 60, "Pet. Mot.") certain evidence submitted by Patent Owner, to which Patent Owner filed an Opposition (Paper 63, "PO Opp.") and Petitioner filed a Reply (Paper 66, "Pet. Mot. Reply"). Patent Owner filed a Motion to Exclude (Paper 61, "PO Mot.") certain evidence submitted by Petitioner, to which Petitioner filed an Opposition (Paper 62, "Pet. Opp.")

[3] Unless otherwise noted, references herein are to the exhibits filed in Case IPR2018-01601.

and Patent Owner filed a Reply (Paper 65, "PO Mot. Reply"). An oral hearing was held on February 4, 2020, and a transcript of the hearing is included in the record (Paper 71, "Tr.").

We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a). For the reasons that follow, we determine that Petitioner has shown by a preponderance of the evidence that claims 1–5, 7–9, 15, 17, 18, and 20–24 are unpatentable.

B. Related Matters

The parties indicate that the '324 patent is the subject of the following district court cases: *SRC Labs, LLC v. Microsoft Corp.*, No. 2:18-cv-00321 (W.D. Wash.), and *SRC Labs, LLC v. Amazon Web Servs., Inc.*, No. 2:18-cv-00317 (W.D. Wash.). *See* Pet. 3–4; Paper 69, 1.

C. The '324 Patent

The '324 patent[4] discloses "multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions." Ex. 1001, col. 1, ll. 38–41. Parallel processing "allows multiple processors to work simultaneously on the same problem to achieve a solution" in less time than it would take a single processor. *Id.* at col. 1, ll. 42–47. "[A]s more and more performance is required, so is more parallelism, resulting in ever larger systems" and associated difficulties, including "facility requirements, power, heat generation and reliability." *Id.* at col. 1, ll. 51–59. The '324 patent discloses that

[4] U.S. Patent No. 7,620,800 B2 (Ex. 1005), a continuation of the '324 patent, is challenged by Petitioner in Case IPR2018-01605. We enter a Final Written Decision in Case IPR2018-01605 concurrently with this Decision.

> if a processor technology could be employed that offers orders of magnitude more parallelism per processor, these systems could be reduced in size by a comparable factor. Such a processor or processing element is possible through the use of a reconfigurable processor. Reconfigurable processors instantiate only the functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ.

Id. at col. 1, l. 63–col. 2, l. 5. The '324 patent describes a known issue where each processor in a multi-processor system is allocated a portion of a problem called a "cell" and "to solve the total problem, results of one processor are often required by many adjacent cells because their cells interact at the boundary." *Id.* at col. 2, ll. 25–31. Passing intermediate results around the system to complete the problem requires using "numerous other chips and busses that run at much slower speeds than the microprocessor," diminishing performance. *Id.* at col. 2, ll. 31–37, col. 5, ll. 18–30, Fig. 1 (depicting a conventional multi-processor arrangement). In an adaptive processor-based system, however, "any boundary data that is shared between . . . functional units need never leave a single integrated circuit chip," reducing "data moving around the system" and improving performance. *Id.* at col. 2, ll. 38–48.

Figure 2 of the ’324 patent is reproduced below.

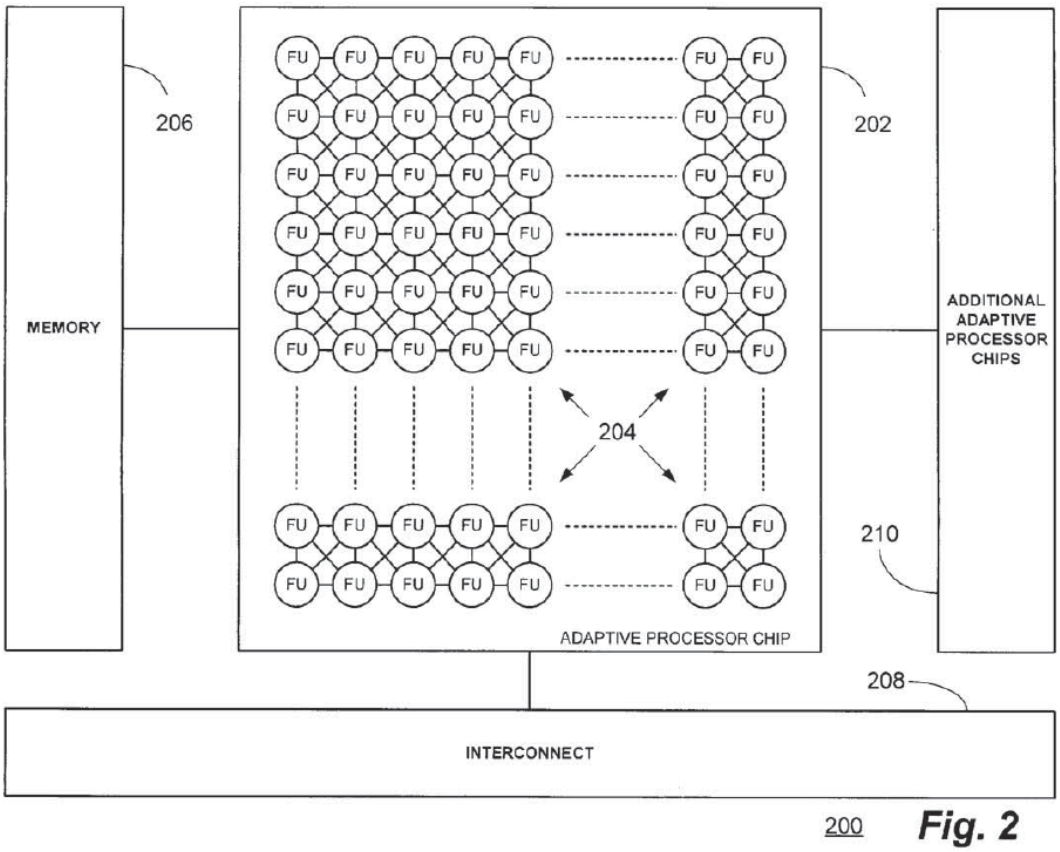


Fig. 2

Figure 2 is “a functional block diagram of an adaptive processor 200 communications path for implementing the technique of the present invention.” *Id.* at col. 5, ll. 32–35. Adaptive processor 200 includes adaptive processor chip 202, which is coupled to memory element 206, interconnect 208, and additional adaptive processor chips 210. *Id.* at col. 5, ll. 35–40. Adaptive processor chip 202 includes thousands of functional units (“FU”) 204 interconnected by “reconfigurable routing resources” inside adaptive processor chip 202, allowing functional units 204 to “exchange data at much higher data rates and lower latencies than a standard microprocessor.” *Id.* at col. 5, ll. 41–47.

Figures 4A and 4B of the ’324 patent are reproduced below.

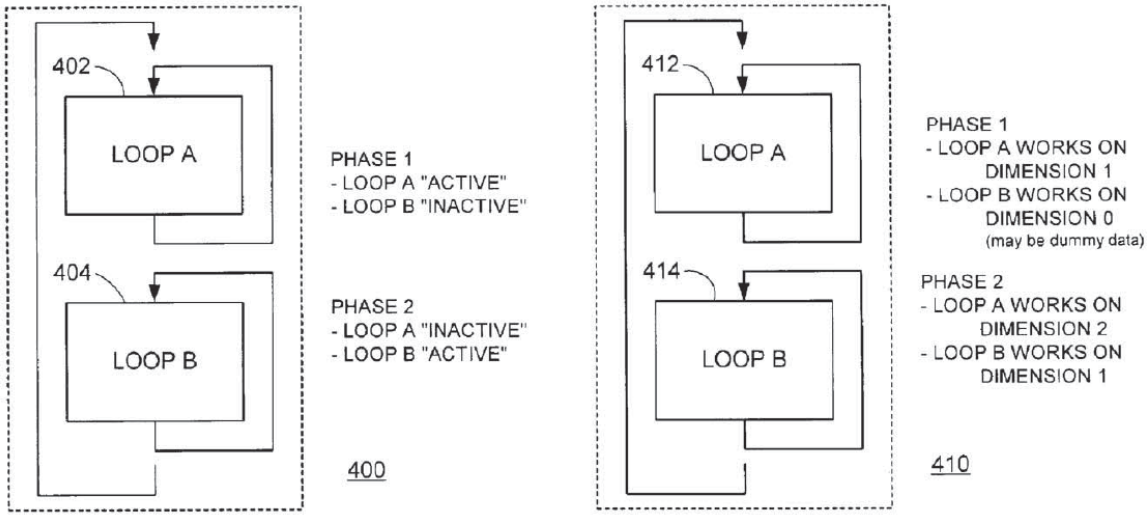


Fig. 4A
Prior Art

Fig. 4B

Figure 4A depicts conventional sequential processing operation 400 where “nested Loops A (first loop 402) and B (second loop 404) are alternately active on different phases of the process.” *Id.* at col. 6, ll. 1–5. Because first loop 402 must be completed before beginning second loop 404, “all of the logic that has been instantiated is not being completely utilized.” *Id.* at col. 6, ll. 6–12. Figure 4B depicts “multi-dimensional process 410 in accordance with the technique of the present invention.” *Id.* at col. 6, ll. 13–16. “[M]ulti-dimensional process 410 is effectuated such that multiple dimensions of data are processed by both Loops A (first loop 412) and B (second loop 414) such that the computing system logic is operative on every clock cycle.” *Id.* at col. 6, ll. 16–20. A “dimension” of data can be “multiple vectors of a problem, multiple plans of a problem, multiple time steps in a problem and so forth.” *Id.* at col. 6, ll. 27–30. The ’324 patent discloses that available resources are utilized more effectively in the multi-dimensional process by “hav[ing] an application evaluate a problem in a data flow sense. That is, it will ‘pass’ a subsequent dimension of a given

problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop." *Id.* at col. 6, ll. 21–27.

The '324 patent states that the disclosed process can be utilized for a variety of applications. *Id.* at col. 9, ll. 18–29. For example, seismic imaging applications, which "process echo data to produce detailed analysis of subsurface features" for oil and gas exploration, would "particularly benefit from the tight parallelism that can be found in the use of adaptive or reconfigurable processors" because they "use data collected at numerous points and consisting of many repeated parameters" and "the results of the computation on one data point are used in the computation of the next." *Id.* at col. 9, ll. 35–44; *see id.* at col. 6, l. 31–col. 7, l. 41, Figs. 5A–5B, 6A–6B (describing a seismic imaging function that can be adapted to utilize the disclosed parallelism, where computational process 610 "loops over the depth slices as indicated by reference number 622 and loops over the shots as indicated by reference number 624"). Also, reservoir simulation applications, which "process fluid flow data in . . . oil and gas subsurface reservoirs to produce extraction models," would benefit from the disclosed process because they define a three dimensional set of cells for the reservoir, utilize repeated operations on each cell, and "information computed for each cell is then passed to neighboring cells." *Id.* at col. 10, ll. 2–13; *see id.* at col. 7, l. 42–col. 8, l. 26, Figs. 7A–7D (describing "process 700 for performing a representative systolic wavefront operation in the form of a reservoir simulation function" in which "the computation of fluid flow properties are communicated to neighboring cells 710" without storing data in memory, "a set of cells can reside in an adaptive processor," and "the pipeline of computation can extend across multiple adaptive processors,"

where the process involves nested loops and systolic walls 712 and 714 of computation at different time sets), col. 8, l. 27–col. 9, l. 17, Figs. 8A–8C, 9A–9C (describing two other processes for performing "a representative systolic wavefront operation"). Finally, the disclosed process may be used for genetic pattern matching applications, which "look[] for matches of a particular genetic sequence (or model) to a database of genetic records," performing repeated operations to "compare[] each character in the model to the characters in [a particular] genetic record." *Id.* at col. 11, ll. 55–64.

D. Illustrative Claims

Claim 1 of the '324 patent is independent. Claims 2–5, 7–9, 15, 17, 18, and 20–24 each depend from claim 1. Claims 1 and 15 recite:

> 1. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:
>
> transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor;
>
> instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are instantiated and wherein each instantiated functional unit at the at least one reconfigurable processor interconnects with each other instantiated functional unit at the at least one reconfigurable processor based on reconfigurable routing resources within the at least one reconfigurable processor as established at instantiation, and wherein systolically linked lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor;

> utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and
>
> substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a second computational loop wherein said systolic implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.
>
> 15. The method of claim 1 wherein instantiating includes establishing a stream communication connection between functional units.

E. Evidence

The pending grounds of unpatentability in the instant *inter partes* review are based on the following prior art:

> Jean-Luc Gaudiot, "Data-Driven Multicomputers in Digital Signal Processing," *Proceedings of the IEEE, Special Issue on Hardware and Software for Digital Signal Processing*, vol. 75, no. 9, Sept. 1987, pp. 1220–1234 (Ex. 1010, "Gaudiot");
>
> Duncan A. Buell, Jeffrey M. Arnold, & Walter J. Kleinfelder, SPLASH2: FPGAS IN A CUSTOM COMPUTING MACHINE (1996) (Ex. 1007, "Splash2");
>
> Carl Ebeling *et al.*, "Mapping Applications to the RaPiD Configurable Architecture," *Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines*, Apr. 16–18, 1997, pp. 106–115 (Ex. 1009, "RaPiD");
>
> Michael Rencher & Brad L. Hutchings, "Automated Target Recognition on SPLASH 2," *Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines*, Apr. 16–18, 1997, pp. 192–200 (Ex. 1011, "Chunky SLD");
>
> Yong-Jin Jeong & Wayne P. Burleson, "VLSI Array Algorithms and Architectures for RSA Modular Multiplication," *IEEE*

> *Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 5, no. 2, June 1997, pp. 211–217 (Ex. 1061, "Jeong"); and
>
> D. Roccatano *et al.*, "Development of a Parallel Molecular Dynamics Code on SIMD Computers: Algorithm for Use of Pair List Criterion," *Journal of Computational Chemistry*, vol. 19, no. 7, May 1998, pp. 685–694 (Ex. 1012, "Roccatano").[5]

Petitioner filed a declaration from Harold Stone, Ph.D. (Ex. 1003) with its Petition and a reply declaration from Dr. Stone (Ex. 1076) with its Reply. Patent Owner filed declarations from Jon Huppenthal (Ex. 2100), Houman Homayoun, Ph.D. (Exs. 2029, 2111), and Tarek El-Ghazawi, Ph.D. (Ex. 2164).

F. Asserted Grounds

The instant *inter partes* review involves the following grounds of unpatentability:

Petition(s)	**Claim(s) Challenged**	**35 U.S.C. §**	**Reference(s)/Basis**
IPR2018-01601, IPR2018-01602, IPR2018-01603	1, 15, 18, 21, 22	102(a), 102(b)[6]	Splash2
IPR2018-01601, IPR2018-01602, IPR2018-01603	1, 15, 18, 21, 22	103(a)	Splash2

[5] When citing the prior art references and other exhibits, we refer to the page numbers in the bottom-right corner added by the filing party. *See* 37 C.F.R. § 42.63(d)(2).

[6] The Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) ("AIA"), amended 35 U.S.C. §§ 102 and 103. Because the challenged claims of the '324 patent have an effective filing date before the effective date of the applicable AIA amendments, we refer to the pre-AIA versions of 35 U.S.C. §§ 102 and 103.

Petition(s)	Claim(s) Challenged	35 U.S.C. §	Reference(s)/Basis
IPR2018-01601, IPR2018-01602, IPR2018-01603	1, 15, 18, 21, 22	103(a)	Splash2, Gaudiot
IPR2018-01601	8, 9	103(a)	Splash2, RaPiD
IPR2018-01601	8, 9	103(a)	Splash2, RaPiD, Gaudiot
IPR2018-01601	20	103(a)	Splash2, Jeong
IPR2018-01601	20	103(a)	Splash2, Jeong, Gaudiot
IPR2018-01602	7, 17, 24	103(a)	Splash2, Chunky SLD
IPR2018-01602	7, 17, 24	103(a)	Splash2, Chunky SLD, Gaudiot
IPR2018-01603	2–5, 22, 23	103(a)	Splash2, Roccatano
IPR2018-01603	2–5, 22, 23	103(a)	Splash2, Roccatano, Gaudiot

II. ANALYSIS

A. Motions to Exclude

The party moving to exclude evidence bears the burden of proof to establish that it is entitled to the relief requested—namely, that the material sought to be excluded is inadmissible under the Federal Rules of Evidence. *See* 37 C.F.R. §§ 42.20(c), 42.62(a). For the reasons discussed below, Petitioner's Motion is granted-in-part, denied-in-part, and dismissed-in-part, and Patent Owner's Motion is denied-in-part and dismissed-in-part.

1. Petitioner's Motion to Exclude

Exhibits 2100, 2066, 2076, and 2092: Petitioner moves to exclude the entirety of the declaration (Ex. 2100) of Mr. Huppenthal, one of the named inventors of the '324 patent, "as not being relevant to any issue on which trial has been instituted, and for lacking foundation, containing hearsay, and/or causing undue prejudice." Pet. Mot. 3–6. Petitioner argues that the declaration includes "irrelevant narrative discussion of [Mr. Huppenthal's] participation in reconfigurable computing" and statements "either based on hearsay or lack of personal knowledge." *Id.* at 3–4. Petitioner also moves to exclude paragraphs 80 and 82–86 of the declaration based on Mr. Huppenthal's alleged "refusal to answer questions concerning those portions of the declaration" during cross-examination. *Id.* at 1–3.

Petitioner also moves to exclude three transcripts (Exs. 2066, 2076, 2092) of depositions of Petitioner's declarants from other *inter partes* reviews as "not being relevant to any issue on which trial has been instituted, for containing hearsay, and/or causing undue prejudice." *Id.* at 6–7. Petitioner argues that allowing the transcripts in the record would be "highly prejudicial as they present themselves with the indicia of expert testimony while being totally devoid from the necessary context of the matters from which they originate." *Id.* Patent Owner cites Exhibits 2066 and 2076 in its Response, but does not cite Exhibit 2092 in its Response or Sur-Reply.

Petitioner's Motion is dismissed as moot, as we do not rely on the cited portions of the testimony in a manner adverse to Petitioner in this Decision. As explained below, even if the testimony is considered, we are not persuaded by Patent Owner's arguments regarding the state of the art or alleged nonobviousness of the challenged claims, and Patent Owner has not

shown proof of secondary considerations that would support a conclusion of nonobviousness. *See infra* Sections II.E–II.J.

Exhibit 2111: Petitioner moves to exclude paragraphs 41, 145, 180, 181, and 247 of the declaration of Dr. Homayoun, which refer to Exhibits 2066 and 2100. Pet. Mot. 8–9. Because we do not exclude those exhibits, we also dismiss as moot Petitioner's Motion with respect to Exhibit 2111.

Exhibits 2067–2075, 2077, 2079–2099, 2101–2103, 2105, 2106, 2109, 2110, 2112–2133, 2139–2151, 2155, 2161–2163, and 2168: Petitioner moves to exclude a number of exhibits as "not being relevant to any issues on which trial has been instituted, lacking foundation, and/or causing undue prejudice" because the exhibits were not discussed or cited, or "only cited superficially," in Patent Owner's Response and Sur-Reply. Pet. Mot. 7–8. Petitioner's Motion is dismissed as moot, as we do not rely on the exhibits in a manner adverse to Petitioner in this Decision. We note, however, that in evaluating Petitioner's asserted grounds of unpatentability, we only consider substantive arguments made by the parties in their papers during trial (i.e., the Petitions, Response, Reply, and Sur-Reply). To the extent a document is filed in the record but never discussed in a paper, there is no substantive argument pertaining to that document that can be considered.

Exhibit 2168: Petitioner moves to exclude Exhibit 2168 under Federal Rules of Evidence 401–403 as cumulative of Exhibit 1007. Pet. Mot. 8. Both exhibits are copies of Splash2. Dr. El-Ghazawi refers to the document in his declaration with the numeral "1007." Ex. 2164 ¶ 40. To ensure a clear record, we grant Petitioner's Motion, expunge Exhibit 2168, and refer herein to Exhibit 1007. *See* 37 C.F.R. §§ 42.7(a), 42.6(d) ("A document already in the record of the proceeding must not be filed

again, not even as an exhibit or an appendix, without express Board authorization.").

Portions of Patent Owner's Response: Petitioner moves to exclude portions of Patent Owner's Response referring to the exhibits that Petitioner seeks to exclude. Pet. Mot. 9. Patent Owner's Response is a paper with attorney arguments, not evidence that may be excluded.[7] Further, we do not exclude any of the exhibits referred to in the identified portions of the Response. Petitioner's Motion is denied as to Patent Owner's Response.

2. Patent Owner's Motion to Exclude

Exhibits 1074, 1077, and 1079: Patent Owner moves to exclude three technical documents filed by Petitioner with its Reply.

First, Patent Owner moves to exclude Exhibits 1074 and 1079 as unauthenticated under Federal Rule of Evidence 901. PO Mot. 6–7. "To satisfy the requirement of authenticating or identifying an item of evidence, the proponent must produce evidence sufficient to support a finding that the item is what the proponent claims it is." Fed. R. Evid. 901(a). Certain evidence, though, is "self-authenticating" and "require[s] no extrinsic evidence of authenticity in order to be admitted." Fed. R. Evid. 902. Exhibit 1074 is an article by Maya Gokhale and Ron Minnich titled "FPGA Computing in a Data Parallel C," and includes an IEEE trade inscription, copyright symbol, and International Standard Book Number (ISBN) on the first page ("0-8186-3890-7/93 $03.00 © 1993 IEEE"). It is

[7] Petitioner did not seek authorization to file a motion to strike Patent Owner's Response. *See* Patent Trial and Appeal Board Consolidated Trial Practice Guide (Nov. 2019), 80–81, *available at* https://www.uspto.gov/TrialPracticeGuideConsolidated ("Trial Practice Guide").

self-authenticating under at least Federal Rule of Evidence 902(6) ("Printed material purporting to be a . . . periodical.") and 902(7) ("An inscription, sign, tag, or label purporting to have been affixed in the course of business and indicating origin, ownership, or control."). Exhibit 1079 is an excerpt from a book by Kevin Skahill titled "VHDL for Programmable Logic." It includes a copyright notice ("Copyright © 1996 by Addison-Wesley Publishing, Inc."), Library of Congress catalogue information, and hand-written library catalogue information. Ex. 1079, Cover 3, v. Petitioner points out where the book is available from "several well-known book sellers." Pet. Opp. 4 & n.2. It is authenticated at least as an ancient document under Federal Rule of Evidence 901(b)(8) because it "is in a condition that creates no suspicion about its authenticity," "was in a place where, if authentic, it would likely be," and "is at least 20 years old when offered." Nothing about either exhibit suggests that it is not what it points to be, and Patent Owner does not point to anything in particular in the exhibits that would indicate otherwise. *See* PO Mot. 6–7; PO Mot. Reply 1–4.

Second, Patent Owner moves to exclude Exhibits 1074, 1077, and 1079 as containing inadmissible hearsay under Federal Rule of Evidence 802. PO Mot. 7. Patent Owner states that Petitioner in its Reply "cites each of these documents to prove the truth of technical matters allegedly asserted in such documents, *i.e.* to support Petitioner's specific factual assertions regarding a technical issue." *Id.* We are not persuaded. Patent Owner does not identify any particular "statement" in any of the exhibits that is being offered "to prove the truth of the matter asserted in the statement," and thus fails to meet its burden to prove inadmissibility as hearsay. *See* Fed. R. Evid. 801(c); 37 C.F.R. § 42.20(c). Even if Patent Owner had done so, Petitioner cites the exhibits to show what a person of ordinary skill in the art

would have known at the time of the '324 patent and, with respect to Exhibit 1074 in particular, how a person of ordinary skill in the art would have understood another reference relied on by Patent Owner that cites the article. *See* Reply 7, 17, 33; Pet. Opp. 5–6; Ex. 2167, 37–38, 208. The exhibits are not being offered for the truth of any particular matter discussed in the references. Finally, Exhibits 1074 and 1079 also are admissible under Federal Rule of Evidence 803(16), which provides as an exception to the hearsay rule "[a] statement in a document that was prepared before January 1, 1998, and whose authenticity is established." *See* 1074, 94 ("1993" date); Ex. 1079, Cover 3, v ("1996" date).

Third, Patent Owner moves to exclude Exhibit 1077 as "irrelevant to the patent and claim construction issues in dispute." PO Mot. 7–8. Exhibit 1077 is an excerpt of certain pages of the Microsoft Computer Dictionary (5th ed. 2002). Patent Owner argues that Exhibit 1077 "is extrinsic evidence pertaining to Petitioner's proffered definitions from the Microsoft Computer Dictionary of 'data structure' and 'data path,' neither of which are claim terms in the patent." PO Mot. 7. Federal Rule of Evidence 401 provides that "[e]vidence is relevant if: (a) it has any tendency to make a fact more or less probable than it would be without the evidence; and (b) the fact is of consequence in determining the action."

Petitioner submits Exhibit 1077 in support of its arguments regarding the interpretation of disputed claim language, specifically the term "stream communication." Reply 33. The meaning of this phrase is "of consequence in determining" whether challenged claim 15 is unpatentable over the asserted prior art, and Exhibit 1077 provides insight as to the meaning of words used in both parties' proposed interpretations. *See infra* Section II.C.4. Exhibit 1077 has some "tendency to make a fact more or less

probable than it would be without the evidence" and is relevant under Federal Rule of Evidence 401. Thus, there is no basis to exclude Exhibit 1077. We also note that Patent Owner appears to have filed a full copy of the dictionary as Exhibit 2065, and the same pages filed as Exhibit 1077 (pages 144–145) that Patent Owner seeks to exclude are in the exhibit that Patent Owner filed. Further, the record contains numerous other dictionary references filed by both parties, including Exhibits 1025, 1059, 2024–2026, and 2038. Patent Owner's Motion is denied as to Exhibits 1074, 1077, and 1079.

Exhibit 1076: Patent Owner also moves to exclude paragraphs 15–17 of the reply declaration of Dr. Stone, which refer to Exhibit 1074 "for the first time," under Federal Rules of Evidence 402 and 403. PO Mot. 8. Because we find no basis to exclude Exhibit 1074, we also deny Patent Owner's Motion with respect to Exhibit 1076. To the extent Patent Owner's position is that Petitioner's Reply and Dr. Stone's reply declaration exceed the proper scope of a reply, we address those arguments below. *See infra* Section II.E.2.a.2; Trial Practice Guide, 79 ("A motion to exclude is not a vehicle for addressing the weight to be given evidence—arguments regarding weight should appear only in the merits documents. Nor should a motion to exclude address arguments or evidence that a party believes exceeds the proper scope of reply or sur-reply.").

Exhibits 1075 and 1078: Patent Owner moves to exclude certain portions of the transcript of the deposition of Dr. Homayoun because the questions asked were "vague, ambiguous, call[] for a legal conclusion, and misleading." PO Mot. 9–11 (citing Ex. 1075, 27:4–12, 65:5–17). Patent Owner also moves to exclude a portion of the transcript of the deposition of Dr. El-Ghazawi because the question asked was "vague, ambiguous, and

calls for a speculative answer." *Id.* at 11 (citing Ex. 1078, 65:12–17). Patent Owner's Motion is dismissed as moot, as we do not rely on the disputed portions of the testimony in rendering our Decision.

B. Level of Ordinary Skill in the Art

In determining the level of ordinary skill in the art for a challenged patent, we look to "1) the types of problems encountered in the art; 2) the prior art solutions to those problems; 3) the rapidity with which innovations are made; 4) the sophistication of the technology; and 5) the educational level of active workers in the field." *Ruiz v. A.B. Chance Co.*, 234 F.3d 654, 666–667 (Fed. Cir. 2000). "Not all such factors may be present in every case, and one or more of them may predominate." *Id.*

Petitioner's declarant, Dr. Stone, testifies that a person of ordinary skill in the art at the time of the '324 patent would have had "an advanced degree in electrical or computer engineering, or computer science with substantial study in computer architecture, hardware design, and computer algorithms," and "at least three years' experience working in the field," or alternatively "a bachelor's degree covering those disciplines and at least four years working [in] the field." Ex. 1003 ¶ 45. According to Dr. Stone,

> [s]uch a person would also have been knowledgeable about the programming, design and operation of computer systems based on reconfigurable components such as FPGAs (field programmable gate arrays) and CPLDs (complex programmable logic devices), including computer systems for performing systolic and data driven calculations. That person would also have been familiar with hardware description languages such as [Very High Speed Integrated Circuit Hardware Description Language (VHDL)] that could be used to configure FPGAs and CPLDS that serve as components of reconfigurable computer systems. Finally, as demonstrated by many of the references

> discussed [in Dr. Stone's declaration], such a person would also have been familiar with various other areas of technology that by 2002 had relied on high performance and parallel computing systems, such as genetic sequence comparisons, image processing, data mining, and processing related to proteins and organic structures.

Id.

Patent Owner states that it "does not dispute the level of education and skill promoted by [Dr. Stone]," and Patent Owner's declarant, Dr. Homayoun, "agree[d] with Dr. Stone's assessment of the level of ordinary skill in the art," noting that such an individual also would have been "experienced in developing with high-level languages (C and Fortran), hardware description languages, and the unique problems involved with programming FPGAs and FPGA based systems." *See* PO Resp. 29; Ex. 2029 ¶ 17; *see also* Ex. 2111 ¶ 133 ("In general, I would agree to the level of education and skill promoted by [Petitioner's] expert [for the '324 patent]."). Dr. Homayoun further expands on his understanding of what a person of ordinary skill in the art at the time of the '324 patent would have known and considered. Ex. 2111 ¶¶ 133–146. For example, Dr. Homayoun testifies that in addition to the technical background set forth in Dr. Stone's definition, a person of ordinary skill in the art would have "considered all of the state of the art [described in Dr. Homayoun's declaration] in the design of computer architecture, . . . [i]ncluding the issues of reconfigurable programming, processor speed, FPGA speed, and cost/benefit analysis of overhead introduction as applied to [high performance computing (HPC)] applications." *Id.* ¶ 133. Also, according to Dr. Homayoun, an ordinarily skilled artisan would have "consider[ed] the technical problems [the '324 patent was] attempting to solve" without using

the patent "as a technical road-map to then conflate the technical problem with the solution," and "would have considered the drawbacks in HPC computing and the deficiencies in FPGA systems and computer architecture design considerations at the time of the disclosed inventions." *Id.* ¶ 140. Patent Owner similarly argues in its Response that a person of ordinary skill in the art would have "considered" all of these issues. PO Resp. 17–29.

We have evaluated all of Patent Owner's arguments and supporting evidence regarding what a person of ordinary skill in the art allegedly would have considered when reading the asserted references. Patent Owner's arguments pertain more to its criticism of Dr. Stone's analysis as allegedly failing to understand the problems solved by the '324 patent and being based on "hindsight bias" than a dispute over the "level" of ordinary skill in the art. *See id.* (also arguing that "Petitioner essentially uses the patent itself as a roadmap for stitching together various prior art references"); Sur-Reply 1–8; Tr. 64:4–65:15. It suffices at this point to conclude that a person of ordinary skill in the art would have had the *technical education and work experience* set forth in Dr. Stone's declaration (and agreed to by Dr. Homayoun). *See* Tr. 64:13–14 (Patent Owner stating that there is no dispute as to "the level of education"). Among other things, such an individual would have had *knowledge* of "multi-adaptive processing systems and techniques," "parallel processing," and field-programmable gate arrays (FPGAs). *See* Ex. 1001, col. 1, ll. 35–59; Ex. 1003 ¶ 45; Ex. 2029 ¶ 17; Ex. 2111 ¶¶ 107–119, 133; Ex. 1007, 1–9; *Custom Accessories, Inc. v. Jeffrey-Allan Indus., Inc.*, 807 F.2d 955, 962 (Fed. Cir. 1986) ("The person of ordinary skill is a hypothetical person who is presumed to be aware of all the pertinent prior art."). What that individual would have considered in evaluating particular

prior art references and making potential combinations, though, is an issue we address below in evaluating Petitioner's grounds of unpatentability.[8]

Based on the full record developed during trial, including our review of the '324 patent and the types of problems and prior art solutions described in the '324 patent, as well as the sophistication of the technology described in the '324 patent, we conclude that a person of ordinary skill in the art would have had (1) an advanced degree in electrical or computer engineering, or computer science with substantial study in computer architecture, hardware design, and computer algorithms, and at least three years of experience working in the field, or (2) a bachelor's degree covering those disciplines and at least four years working in the field. We apply that level of skill for purposes of this Decision.

C. Claim Interpretation

In this proceeding, we interpret the claims of the unexpired '324 patent using the "broadest reasonable construction in light of the

[8] We note that for Petitioner's anticipation ground based on Splash2, the level of ordinary skill in the art is relevant to understanding how a person of ordinary skill in the art would have interpreted the claims and understood the reference, but not with respect to, for example, any motivations or problems a person of ordinary skill in the art would have had in combining teachings. *See Wasica Finance GmbH v. Continental Auto. Sys., Inc.*, 853 F.3d 1272, 1284 (Fed. Cir. 2017) ("Anticipation is an inquiry viewed from the perspective of one skilled in the art."); *Vivint, Inc. v. Alarm.com Inc.*, 741 F. App'x 786, 791–792 (Fed. Cir. 2018) (unpublished) ("While anticipation is proven based on the express and inherent teachings of a single prior art reference, an obviousness analysis reaches beyond the prior art reference and takes into account other considerations such as the level of ordinary skill in the art and any objective indicia of nonobviousness.").

specification of the patent." 37 C.F.R. § 42.100(b) (2017).[9] Under this standard, we interpret claim terms using "the broadest reasonable meaning of the words in their ordinary usage as they would be understood by one of ordinary skill in the art, taking into account whatever enlightenment by way of definitions or otherwise that may be afforded by the written description contained in the applicant's specification." *In re Morris*, 127 F.3d 1048, 1054 (Fed. Cir. 1997); *see In re Smith Int'l, Inc.*, 871 F.3d 1375, 1382–83 (Fed. Cir. 2017) ("[The] broadest reasonable interpretation . . . is an interpretation that corresponds with what and how the inventor describes his invention in the specification."). "Under a broadest reasonable interpretation, words of the claim must be given their plain meaning, unless such meaning is inconsistent with the specification and prosecution history." *TriVascular, Inc. v. Samuels*, 812 F.3d 1056, 1062 (Fed. Cir. 2016). Our interpretation "'cannot be divorced from the specification and the record evidence,' and 'must be consistent with the one that those skilled in the art would reach.' A construction that is 'unreasonably broad' and which does not 'reasonably reflect the plain language and disclosure' will not pass muster." *Microsoft Corp. v. Proxyconn, Inc.*, 789 F.3d 1292, 1298 (Fed. Cir.

[9] The Petitions in this proceeding were filed on September 5, 2018, prior to the effective date of the rule change that replaces the broadest reasonable interpretation standard with the federal court claim interpretation standard. *See* Changes to the Claim Construction Standard for Interpreting Claims in Trial Proceedings Before the Patent Trial and Appeal Board, 83 Fed. Reg. 51,340, 51,340, 51,358 (Oct. 11, 2018) (amending 37 C.F.R. § 42.100(b) effective November 13, 2018) (now codified at 37 C.F.R. § 42.100(b) (2019)); *Game & Tech. Co. v. Wargaming Grp. Ltd.*, 942 F.3d 1343, 1351 (Fed. Cir. 2019) ("If, as here, the [*inter partes* review] stems from a petition filed before November 13, 2018, the claims are given the 'broadest reasonable interpretation' consistent with the specification." (citation omitted)).

2015) (citations omitted), *overruled on other grounds by Aqua Prods., Inc. v. Matal*, 872 F.3d 1290 (Fed. Cir. 2017).

Upon review of the parties' arguments during trial and the evidence as a whole, we conclude that three terms in claim 1 ("systolic," "computational loop," and "pass computed data seamlessly between said computational loops") and one term in claim 15 ("establishing a stream communication connection between functional units") require interpretation.

1. "Systolic"

Claim 1 recites "transforming an algorithm into a calculation that is *systolically* implemented by said reconfigurable computing system at the at least one reconfigurable processor," wherein "*systolically* linked lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor" and "said *systolic* implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops" (emphases added).

In the Decision on Institution, based on the parties' arguments and record at the time, we preliminarily interpreted the term "systolic" recited in claim 1 to mean "the characteristic of rhythmically computing and passing data directly between processing elements in a manner that is transport triggered (i.e., by the arrival of a data object) rather than a program counter or clock driving movement of the data." Dec. on Inst. 17–21. Petitioner agrees with that interpretation. Reply 26. Patent Owner argues that "systolic" has a "plain and ordinary meaning and need not be construed," but if the term is interpreted, it means "[a]n array of many interconnected

functional units that operates in a data flow sense and allows different data to flow in different directions." PO Resp. 42, 50.

Based on our review of the full trial record, we maintain our preliminary conclusion in the Decision on Institution that the applicants described the plain meaning of "systolic" during prosecution of the '324 patent. *See* Dec. on Inst. 17–21. Specifically, following an Office Action rejecting claim 1 under 35 U.S.C. § 112, first paragraph, and 35 U.S.C. § 103, the applicants conducted an interview with the examiner and proposed, among other amendments, adding the "systolic" language above to the claim. Ex. 1002, 186–212. The applicants then filed an Office Action response amending the claim, noting that the examiner had requested during the interview that the applicants "further define the term[s] instantiated and systolic" and arguing the following in response to the 35 U.S.C. § 112, first paragraph, written description rejection:

> *[T]he term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object.* Data flows across the array between functional units, usually with different data flowing in different directions. David J. Evans in his work, Systolic algorithms. Systolic algorithms, number 3 in Topics in Computer Mathematics. Gordon and Breach, 1991 define a Systolic system as a "network of processors which *rhythmically compute an[d] pass data* through the system[.]" *Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop.* In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of

> functional units. Thus, the process claimed by the Applicant therefore significantly increases the computing processes taking place in a reconfigurable processor.

Ex. 1002, 224–226 (emphases added); *see also id.* at 119 (stating that the term "'systolic,' coined by H.T. Kung of Carnegie-Mellon, refers to the rhythmic transfer of data through the pipeline, like blood flowing through the vascular system"). Importantly, after the applicants filed the Office Action response amending claim 1 to include the "systolic" language and making the above argument, the examiner allowed the claims. *Id.* at 234–238. During prosecution, "an applicant's amendment accompanied by explanatory remarks can define a claim term by demonstrating what the applicant meant by the amendment." *Personalized Media Comm'cns, LLC v. Apple Inc.*, 952 F.3d 1336, 1340 (Fed. Cir. 2020). Patent Owner acknowledges that the prosecution history argument quoted above was "an explanation of the plain and ordinary meaning of the term 'systolic.'" PO Resp. 47.

Our interpretation also is consistent with the Specification, in particular the description of Figures 8A and 8B showing an exemplary "systolic" operation where "data is continually passed directly from one computational loop to another for processing without intervening structures between the loops" and without a program counter or clock driving movement of the data. Ex. 1003 ¶ 70; *see* Ex. 1001, col. 8, ll. 27–45 (disclosing a "representative systolic wavefront operation" in which "systolic processing in the process 800 can pass previously computed data down within a column" and "to subsequent columns as well," with the computational loops operating concurrently); Ex. 1002, 227 (applicants

citing Figures 8A and 8B and the accompanying text as a "a systolic wave front operation" during prosecution of the '324 patent).

Other documentation from the time also shows that the applicants' description of the term "systolic" is how a person of ordinary skill in the art would have understood the term. H.T. Kung, *Why Systolic Architectures?*, IEEE Jan. 1982 (Ex. 1016, "Kung"), discloses:

> A systolic system consists of a set of interconnected cells, each capable of performing some simple operation. Because simple, regular communication and control structures have substantial advantages over complicated ones in design and implementation, cells in a systolic system are typically interconnected to form a systolic array or a systolic tree. *Information in a systolic system flows between cells in a pipelined fashion, and communication with the outside world occurs only at the "boundary cells."* For example, in a systolic array, only those cells on the array boundaries may be I/O ports for the system.
>
> . . .
>
> The basic principle of a systolic architecture, a systolic array in particular, is illustrated in Figure 1. By replacing a single processing element with an array of [processing elements], or cells in the terminology of this article, a higher computation throughput can be achieved without increasing memory bandwidth. *The function of the memory in the diagram is analogous to that of the heart; it "pulses" data (instead of blood) through the array of cells. The crux of this approach is to ensure that once a data item is brought out from the memory it can be used effectively at each cell it passes while being "pumped" from cell to cell along the array.* This is possible for a wide class of compute-bound computations where multiple operations are performed on each data item in a repetitive manner.

Id. at 39 (emphases added).

Figure 1 of Kung is reproduced below.

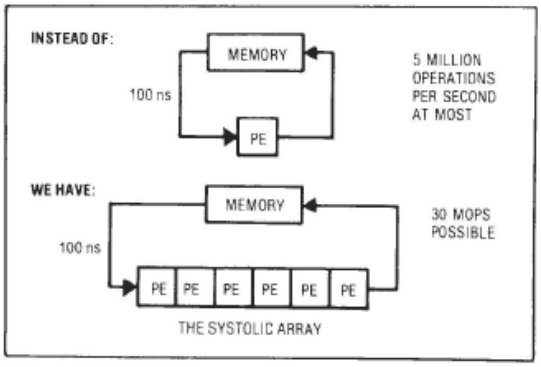


Figure 1. Basic principle of a systolic system.

Figure 1 depicts how data travels and interacts with memory using a systolic array of processing elements versus a single processing element. *Id.* at 38. Another reference by the same author states:

> A systolic system is a network of processors which rhythmically compute and pass data through the system. . . . In a systolic computing system, the function of a processor is analogous to that of the heart. Every processor regularly pumps data in and out, each time performing some short computation, so that a regular flow of data is kept up in the network.

Ex. 1015, 5. Including in the interpretation "rhythmically computing and passing data directly between processing elements" and operating in a "transport triggered" manner is consistent with Kung's description of each processing element processing data and "puls[ing]" or "pump[ing]" it to the next processing element in the array. *See* Ex. 1016, 39; Ex. 1003 ¶¶ 76–78.

Patent Owner in its Response relies on the same portions of the prosecution history and Kung discussed above. *See* PO Resp. 42–50. That evidence, however, supports our preliminary interpretation, not Patent Owner's proposed interpretation, for the reasons explained above. Patent Owner also argues that "systolic means an array of interconnected processing elements that only interact with memory at the array boundaries so that the data is processed by multiple processing elements before

returning to memory." PO Resp. 44–48 (citing Ex. 1001, col. 2, ll. 38–42, Fig. 2; Ex. 2046 ¶ 16; Ex. 2111 ¶¶ 125–131, 177).[10] We do not see how that undermines the preliminary interpretation, however. Logically, if processing elements interact with memory *only* at array boundaries, there can be no memory receiving data between the processing elements; data would be passed "directly" between them. *See* Reply 29. We further address the parties' arguments regarding the use of the word "directly" in interpreting "pass computed data seamlessly between said computational loops" below. *See infra* Section II.C.3.

Applying the broadest reasonable interpretation of the claims in light of the Specification, we interpret "systolic" to mean "the characteristic of rhythmically computing and passing data directly between processing elements in a manner that is transport triggered (i.e., by the arrival of a data object) rather than a program counter or clock driving movement of the data."

[10] Dr. Homayoun does not provide an opinion as to the proper interpretation of "systolic," but testifies that "[a] person or ordinary skill in the art would *not* consider the terms systolic, data driven, and seamless to have distinct meanings." Ex. 2111 ¶ 177 (emphasis added). We understand this to be a typographical error, given the preceding heading stating the opposite and Patent Owner's arguments in its Response. *See id.*; PO Resp. 34. We agree that "systolic" and "pass computed data seamlessly between said computational loops" in claim 1 have different meanings. Although both of our interpretations use the word "directly," we interpret the terms to mean different things. *See infra* Sections II.C.1, II.C.3. Nevertheless, the terms plainly are related, as claim 1 recites that the "systolic" implementation of the calculation is what "enables" the first and second computational loops to "pass computed data seamlessly between said computational loops."

2. "Computational Loop"

Claim 1 recites "utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a *first computational loop*" and "substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a *second computational loop*" (emphases added).

In the Decision on Institution, we preliminarily interpreted "computational loop" in claim 1 to mean "a set of computations that is executed repeatedly, either a fixed number of times or until some condition is true or false." Dec. on Inst. 21–23. Petitioner agrees with that interpretation. Reply 35. Patent Owner states that it "does not disagree with" the preliminary interpretation, but "proposes a small clarification to reflect the plain and ordinary meaning of a loop within the field of high-performance computing and in particular in the context of the '324 Patent." PO Resp. 70. Specifically, Patent Owner proposes the following interpretation: "a set of computations that is executed repeatedly *per datum*, either a fixed number of times or until some condition is true or false."[11] *Id.* at 69–70 (emphasis added).

Patent Owner in its Response cites three technical dictionary definitions of "loop," upon which we had based the preliminary interpretation as reflective of the plain meaning of the term. *See id.* at 70–71; Dec. on Inst. 21–23. None of those definitions, however, requires that a set of instructions or computations be executed repeatedly "per datum." *See* Ex. 2024, 4 ("[a] sequence of instructions that is repeated

[11] Patent Owner's proposed interpretation in its Preliminary Response—"a sequence of computations that is repeated until a prescribed condition is satisfied"—did not include a "per datum" requirement. Paper 15, 21–22.

until a prescribed condition, such as agreement with a data element or completion of a count, is satisfied"); Ex. 2025, 5 ("a series of instructions being carried out repeatedly until a terminal condition prevails" or "[a] sequence of computer instructions that repeats itself until a predetermined count or other test is satisfied, or until the process is interrupted by operator intervention"); Ex. 2026, 8 ("[a] set of statements in a program executed repeatedly, either a fixed number of times or until some condition is true or false"); *see also* Ex. 2038, 3 (defining "computation" as "an act, process, or method of computing"). Indeed, Patent Owner cites the definitions in asserting that "[a] 'computational loop' is an iterative sequence of computations that repeats until a prescribed condition is satisfied" (without mentioning any "per datum" requirement). PO Resp. 80.

The Specification of the '324 patent is consistent with the technical dictionary definitions, and does not require that the set of computations executed repeatedly as a "loop" be on a particular piece of data. *See* Ex. 1001, col. 7, ll. 3–5 (explaining that computational process 610 "loops over the depth slices" and "loops over the shots" of a seismic imaging application), Fig. 7A (depicting three loops each performing computations a particular number of times as "k = 1, nz"; "j = 1, ny"; and "i = 1, nx"), Fig. 8B (depicting two loops as "i = 1, l" and "k = 1, m"). Patent Owner does not cite—and we do not find—any specific support in the Specification (including a reference purportedly incorporated by reference in the '324 patent (Ex. 2037, "Caliga")) for imposing a "per datum" requirement. *See* PO Resp. 71–72 (citing Ex. 1001, col. 4, ll. 59–63, col. 6, ll. 1–30, col. 6, l. 47–col. 7, l. 48, col. 8, ll. 27–45, Figs. 4A–4B, 6B–6G, 7A, 8A–8B; Ex. 2037, 4, 5, 7, 12, 13, 16–19). Indeed, the Specification and Caliga do not use the term "datum." Caliga, in fact, describes the opposite of what

Patent Owner proposes, namely a "loop" of a set of computations executed repeatedly using *different* data. *See* Ex. 1076 ¶¶ 5–8; Ex. 2037, 5 (describing a "[l]oop over filter coefficients" where index j is incremented and different values for "Data", "R_j," and "R_{j+1}" are used at each iteration); Reply 36.[12]

Patent Owner further cites as support U.S. Patent No. 8,589,666 B2 (Ex. 2027, "the '666 patent"), which was originally assigned to the same original assignee as the '324 patent (SRC Computers, Inc.) but is unrelated to the '324 patent. PO Resp. 72–73. The cited portions of the '666 patent generally describe a "loop body" with "new data" (plural) "fed in on every clock tick," and do not describe a "loop" as requiring a set of computations to be executed repeatedly "per datum" (singular). *See* Ex. 2027, col. 2, l. 64–col. 3, l. 23, col. 6, ll. 6–28. Dr. Homayoun also testifies that "[a] computational loop evaluates each piece of data multiple times, 'a fixed number of times or until some condition is true or false,'" but does not explain the basis for that interpretation, in the Specification of the '324 patent or otherwise. Ex. 2111 ¶ 207.

Finally, in its Sur-Reply and at the oral hearing, Patent Owner relied on Figure 4B of the '324 patent as support for its proposed interpretation, arguing that the figure "unambiguously describes . . . two loops A and B operating a number of times (looping) on *each* dimension of data" and that

[12] We do not agree with Patent Owner that Petitioner's Reply and Dr. Stone's reply testimony are improper, as they respond directly to Patent Owner's arguments in the Response regarding claim interpretation, where Patent Owner argued a "per datum" requirement for the first time. *See* Sur-Reply 8–9; Reply 36; Ex. 1076 ¶¶ 5–8; 37 C.F.R. § 42.23(b) ("A reply may only respond to arguments raised in the corresponding . . . patent owner response.").

our preliminary interpretation "would exclude from the claims the '324 Patent's embodiments and figures." *See* Sur-Reply 15–17; Tr. 52:20–58:21, 82:8–84:2. Patent Owner also stated during the oral hearing that "per datum" in its proposed interpretation could be replaced with "per dimension," as Patent Owner did not intend there to be a "difference" between the terms. Tr. 53:10–11, 55:3–10, 82:8–17.

We disagree for two reasons. First, claim 1 already recites two functional units operating on two data dimensions forming two computational loops. A first functional unit "operate[s] upon a subsequent data *dimension* of said calculation forming a first computational loop" and a second functional unit concurrently "operate[s] upon a previous data *dimension* of said calculation forming a second computational loop" (emphases added). Dependent claims 2–5 specify different options for what those dimensions may comprise, namely "multiple vectors," "multiple planes," "multiple time steps," or "multiple grid points" in the calculation. To the extent Patent Owner's proposed interpretation would require that a functional unit operate on a particular data "dimension" of the calculation forming a computational loop, that is already encompassed in the claim language.

Second, the Specification indicates that a data "dimension" is not the same thing as an individual "datum."

Figure 4B of the ’324 patent is reproduced below.

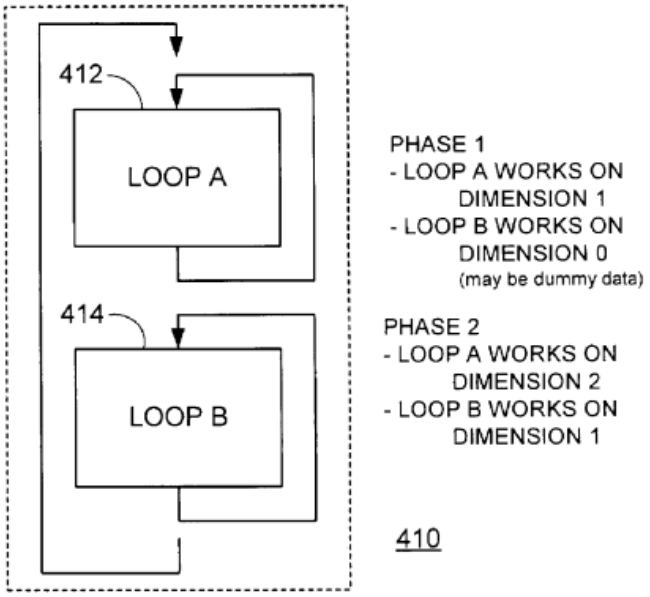


Fig. 4B

Figure 4B depicts “multi-dimensional process 410 . . . effectuated such that multiple dimensions of data are processed by both Loops A (first loop 412) and B (second loop 414) such that the computing system logic is operative on every clock cycle.” Ex. 1001, col. 6, ll. 13–20. An application

> will “pass” a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, *a “dimension” of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth*.

Id. at col. 6, ll. 24–30 (emphasis added). The Specification uses the phrasing “dimension” “of data” or “dimension” “of a given problem,” not “datum” individually, and provides examples of possible dimensions that mirror the

language of dependent claims 2–5. It does not describe loop calculations applied to the same individual "datum" repeatedly.[13]

Having reviewed all of the cited evidence, we do not find sufficient support for limiting the term "computational loop" to require that computations be executed repeatedly "per datum." Applying the broadest reasonable interpretation of the claims in light of the Specification, we interpret "computational loop" to mean "a set of computations that is executed repeatedly, either a fixed number of times or until some condition is true or false."

3. "Pass Computed Data Seamlessly Between Said Computational Loops"

Claim 1 recites that "said systolic implementation of said calculation enables said first computational loop and said second computational loop execute[14] concurrently and pass computed data seamlessly between said computational loops."

Petitioner argues that "pass computed data seamlessly between said computational loops" in claim 1 means "communicate computed data

---

[13] The parties do not refer to anything in the prosecution history as supporting their proposed interpretations of "computational loop." During prosecution, the applicants initially proposed (during the examiner interview discussed above in connection with the term "systolic") amending claim 1 to recite that "said first of said instantiated functional units and said second of said instantiated functional units are within a nested loop of said calculation." Ex. 1002, 208–209. The applicants then instead amended the claim to recite, rather than a "nested loop," first and second "computational loops" executing concurrently with computed data seamlessly passed between them. *Id.* at 214–215.

[14] We read the claim language to mean that the systolic implementation enables the loops "to" execute concurrently and pass computed data seamlessly between the computational loops.

directly between functional units that are calculating computational loops." Pet. 19; Reply 20. Based on the record at the time, we preliminarily agreed with Petitioner's proposed interpretation in the Decision on Institution. Dec. on Inst. 23–26. Patent Owner argues that the phrase instead should be interpreted to mean "communicating the computed data over the reconfigurable routing resources." PO Resp. 34–35.

According to the plain language of the claim, which recites "pass[ing] computed data seamlessly between said computational loops," "seamlessly" refers to how computed data is passed *between* the first and second computational loops (performed by the first and second functional units of the reconfigurable processor)—not, for example, how computed data is passed from either of the computational loops to any other component unrelated to the functional units, or vice versa.

The only other time "seamlessly" appears in the '324 patent is in independent claims 25 and 51, which mirror the language of claim 1, reciting that computed data is passed "seamlessly" between systolic walls or columns of a calculation. The written description does not use the terms "seam" or "seamlessly." Figure 2, though, shows functional units 204 interconnected without any intervening structures between them, which is consistent with Petitioner's proposed interpretation. *See* Ex. 1001, col. 5, ll. 41–47. Figures 7A–7B and 8A–8B similarly show direct communication of data from one computational loop to another. *See id.* at col. 7, l. 42–col. 8, l. 6 (loops 702, 704, and 706), col. 8, ll. 27–55 (loops 812 and 814). Although the description of the figures does not use the terms "seam" or "seamlessly," a person of ordinary skill in the art would have understood the claim language in light of that description, given that it describes and

shows the interaction between multiple "loops," as recited in claim 1. *See id.*

The prosecution history of the '324 patent also supports Petitioner's proposed interpretation. The applicants added the "seamlessly" language to claim 1 in the Office Action response discussed above in connection with the term "systolic." Ex. 1002, 214–215. In response to the § 112, first paragraph, written description rejection, the applicants indicated that data from one functional unit performing a computational loop is passed directly to another functional unit:

> Data flows across the array between functional units, usually with different data flowing in different directions. . . . [I]n the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units. Thus, the process claimed by the Applicant therefore significantly increases the computing processes taking place in a reconfigurable processor.

Ex. 1002, 226. Although the language above primarily relates to the term "[s]ystolic," we also find it significant for purposes of interpreting the "seamlessly" phrase because it refers to the limitation expressly in describing "Applicant's invention." *See id.*; Ex. 1003 ¶ 98; *Shire Dev., LLC v. Watson Pharms., Inc.*, 787 F.3d 1359, 1366 (Fed. Cir. 2015) (holding that even where "prosecution history statements do not rise to the level of unmistakable disavowal, they do inform the claim construction").

Finally, Petitioner's proposed interpretation is consistent with the testimony of Mr. Huppenthal, one of the named inventors of the '324 patent, who describes "interpret[ing] standard high level language program

constructs, such as the fact that the output variable from one loop is the input to another, and then implement[ing] this as a *storage free, seamless connection between the two loops* implemented on the FPGA," and states:

> By seamlessly I mean that the results of one loop streamed from that loop's output to the input of the next loop *without being placed in a circuit element that required explicit address based on read or write operations such as a data register or memory* or through a switch that requires additional non-data content for routing purposes.

Ex. 2100 ¶ 78 (emphases added).

Based on our review of the full record after trial, Patent Owner's arguments regarding the "seamlessly" language in claim 1 are not persuasive. *See* PO Resp. 34–42; Sur-Reply 20–21. First, Patent Owner relies on portions of the Specification describing interaction of cells "at the boundary," "the problem of passing data over numerous boundaries (or seams) between processing elements in typical multi-processor systems," and the '324 patent's solution to that problem of "staying on a single FPGA chip, effectively eliminating the associated boundaries or seams from chip-to-chip communication." PO Resp. 35–36, 39 (citing Ex. 1001, col. 2, ll. 25–48) (emphasis omitted); *see* Ex. 2111 ¶¶ 160–162. The cited portions of the Specification do not use the terms "seam" or "seamlessly" and, unlike the portions cited by Petitioner describing Figures 7A–7B and 8A–8B, do not describe interaction between "loops" in the disclosed method. We find them less relevant than the portions discussed above, particularly given that the claim language at issue recites passing computed data seamlessly "between" the computational loops (performed by the functional units of the reconfigurable processor).

Second, Patent Owner relies on certain statements made by the applicants during prosecution of the '324 patent. PO Resp. 36–39; *see* Ex. 2111 ¶¶ 163–166. Three of those statements mirror the Specification language cited by Patent Owner and, importantly, predate the addition of the "seamlessly" language to claim 1 (and likewise do not use the terms "seam" or "seamlessly"). *See* Ex. 1002, 117–118, 148–150, 174–175. In addition, Patent Owner cites the following statement by the applicants in the Office Action response discussed above:

> The instantiation of the at least one reconfigurable processor with at least two functional units enables each functional unit to communicate with each other. Certainly communication between other reconfigurable processors within the system would require [a] communication protocol but communication between functional units within an individual reconfigurable processor is free of such a requirement. To alleviate any confusion, the reference to the term "protocol" has been replaced with an "interconnection" between functional units that is established by reconfigurable routing resources inside each chip.

Id. at 224–25; *see* PO Resp. 37–39. The cited language pertains to the "instantiating" step recited earlier in the claim, which also was amended in the response, including replacing "communications . . . independent of external and internal communication protocols" with "interconnects . . . based on reconfigurable routing resources." *See* Ex. 1002, 214, 224–25. By contrast, the applicants added the "seamlessly" limitation to specify the interaction between the computational loops (i.e., that computed data is passed "seamlessly" between them).

Third, Patent Owner disputes the "directly" aspect of Petitioner's proposed interpretation because "it would exclude standard FPGAs . . . since standard FPGAs contain reconfigurable routing resources (comprising buffers and switches) between the configurable logic blocks," and thus

would "exclude the very embodiments of the '324 Patent." PO Resp. 40–41, 47–48 (citing Ex. 1035, 31; Ex. 2078, 19–29, 32–34, 37–41, 46–51, 59–65). We agree with Petitioner that "[j]ust because a standard FPGA may include memories does not mean that when functional units are instantiated within such an FPGA that the memories are necessarily placed *between* functional units." *See* Reply 25. Claim 1 recites two functional units being instantiated and concurrently performing two computational loops, with computed data passed seamlessly between the computational loops. Petitioner's proposed interpretation of the "seamlessly" phrase correctly pertains to how computed data is passed *between* the computational loops when so instantiated, and is not inconsistent with the mere existence of memory on an FPGA.

Fourth, Patent Owner argues that the word "directly" makes Petitioner's proposed interpretation unreasonable because it "introduce[s] ambiguity and confusion." PO Resp. 36, 40–41; *see* Ex. 2111 ¶¶ 169–176. According to Patent Owner, Dr. Stone on cross-examination could not identify what "intervening" structures would prevent direct communication, and contradicted himself by stating that "an intervening memory would not be a direct connection" and later that "if a register . . . were between the two processing elements then the connection would still be direct, but if the intervening structure were a buffer then the connection would not be direct." PO Resp. 41 (citing Ex. 2064, 86:13–91:24). Patent Owner argues that if Dr. Stone's view is adopted, "[t]he same circuit would be both direct and indirect, depending on where the boundaries of the 'processing element' are arbitrarily drawn with respect to intervening structures." Sur-Reply 20–21.

We disagree that the word "directly" in Petitioner's proposed interpretation is unclear or that Dr. Stone's testimony about it introduces ambiguity. Dr. Stone testified as follows:

> Q. Okay. And you mention the word "directly," it was passing data directly between processing elements. What does that phrase mean to you or what's the context? What are you trying to describe there?
>
> A. That the data goes from first to the second *without going to something intervening*. It directly go – is connected immediately. Indirectly we – you go through one or more intervening places to get there.
>
> Q. Okay. So would memory, if the data was going from one processing element to memory and then back to a processing element, is that something you would consider as an intervening thing?
>
> A. Well, that would not be a direct connection of the output of the cell to the next cell. It says, "Between processing elements you're directly connected." *If you're saying you have a processing element outputting to memory and then coming back to another processing element, that would not be direct*.

Ex. 2064, 85:14–86:12 (emphases added). When asked about a register in particular, Dr. Stone testified as follows:

> Q. Well, how about a – a register? Would that be an intervening structure?
>
> A. I – I'm puzzled because that – that register would be within – *within the processing element in my mind*.
>
> Q. Okay.
>
> A. If it's *within the processing element as a register*, yeah, I would put it there, then *the output of that register, if it's connected directly to the input of the next processing element, would be direct*.

Id. at 86:19–87:5 (emphases added); *see also id.* at 87:23–25 ("If the register is part of the processing element, then the connection would be direct.").

Data is not communicated "directly" between processing elements when it is communicated through an intervening structure *between* them. *See id.* at 85:14–91:24. Examples of such a structure are memory and

another processing element, but logically others are possible as well depending on how they are situated with respect to the processing elements. *See id.* at 86:13–18. When a register is *within* a processing element, however, data can still be communicated "directly" (provided there is a direct connection between the processing elements), as Dr. Stone explained. *See id.* at 85:14–91:24. We do not see any ambiguity in making this distinction. The boundaries of a processing element are not arbitrary as Patent Owner contends, but rather, in the context of a particular written document, would depend on how the reference describes the processing element and the communication of data to and from the processing element. *See, e.g.*, Ex. 1001, Fig. 2 (depicting functional units 204 of adaptive processor chip 202). Logically, also, in an arrangement with multiple processing elements, there must be some division between the processing elements, otherwise there would not be multiple elements. *See* Tr. 14:1–5.

Applying the broadest reasonable interpretation of the claims in light of the Specification, we interpret "pass computed data seamlessly between said computational loops" to mean "communicate computed data directly between functional units that are calculating computational loops."

4. "Establishing a Stream Communication Connection Between Functional Units"

Claim 15 depends from claim 1 and recites that "instantiating includes establishing a stream communication connection between functional units" (the "stream communication" limitation). Petitioner argues that "stream communication" should be interpreted to mean "communication of a data sequence." Pet. 19–21. Patent Owner argues that "stream communication" means "a data path that acts like a queue connecting via the reconfigurable

routing resources a producer and a consumer of data that operate concurrently." PO Resp. 50–69. We did not preliminarily interpret "stream communication" in the Decision on Institution.

We begin with the language of the claim. Claim 15 recites that "instantiating" includes establishing a stream communication connection "between functional units." "[I]nstantiating" in claim 15 refers to the "instantiating" step of parent claim 1 (emphasis added):

> *instantiating at least two of said functional units* at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are instantiated and wherein each instantiated functional unit at the at least one reconfigurable processor interconnects with each other instantiated functional unit at the at least one reconfigurable processor based on *reconfigurable routing resources within the at least one reconfigurable processor as established at instantiation* . . . .

Thus, according to the plain language of the claim, what is being "instantiat[ed]" is the at least two "functional units." The "reconfigurable routing resources," based on which the instantiated functional units are "interconnect[ed]," are established "at" the instantiation.

Turning to the language of claim 15, regardless of whether "stream communication" is used as an adjective (as Petitioner contends) or a noun (as Patent Owner contends), the term appears immediately before and modifies the phrase "connection between functional units." *See* Reply 30–31; Sur-Reply 10 & n.2. The claim, therefore, requires establishing a connection of a particular type, namely a "stream communication" type of connection. Nothing further in the language of the claim limits or sheds light on what that type of connection entails.

Nor does the Specification. The only other time "stream communication" appears in the '324 patent is in claim 40, which mirrors the language of claim 15. The written description never uses the term.[15] It also does not differentiate between different types of connections between functional units.

Both parties, however, refer to patents that the '324 patent purports to incorporate by reference. *See* Pet. 20–21; PO Resp. 62–66. The '324 patent states that it is "related to the subject matter of," and "incorporate[s] in [its] entirety by . . . reference," U.S. Patent No. 6,434,687 B1 (Ex. 1014, "the '687 patent").[16] Ex. 1001, col. 1, ll. 9–20. Petitioner points to the phrase "streams of operands" in the '687 patent as supporting its proposed interpretation. Pet. 20. The cited portion describes multi-adaptive (MAP) processor 112 with reconfigurable array 42 having one or more high performance FPGAs. Ex. 1014, col. 9, ll. 1–6. "After configuration, the user array 42 can perform whatever function it was programmed to do." *Id.* at col. 9, ll. 11–12. "In order to maximize its performance for vector processing, the array 42 should be able to access two streams of operands simultaneously" by using a chain port "connector allow[ing] the MAP element 112 to use data provided to it by a previous MAP element 112." *Id.* at col. 9, ll. 12–26.

---

[15] Petitioner argues that its proposed interpretation is consistent with a portion of the Specification describing systolic wall of computation 712 in Figure 7C and stating that "communication of values between adjacent rows . . . in the virtual wall can occur without storing values to memory." Ex. 1001, col. 8, ll. 7–12; *see* Pet. 20–21; Ex. 1003 ¶ 117. The cited portion, however, never uses the term "stream communication."

[16] Petitioner challenges the '687 patent in Case IPR2018-01594.

We find the cited portion to be of limited relevance. The '687 patent uses "streams of operands" rather than "stream communication" and does not describe the "streams of operands" other than the single sentence above. Also, given the length of the '687 patent disclosure, it is unclear exactly what aspects of the '687 patent are incorporated in the '324 patent or how they would apply in the context of the '324 patent invention. *See* Ex. 1001, col. 1, ll. 9–20 (merely stating that the '324 patent is "related to the subject matter of" the '687 patent). If anything, the cited portion of the '687 patent simply indicates that certain data (i.e., "operands") can be streamed from one MAP element to another.

Patent Owner similarly quotes other excerpts from the '687 patent describing the use of a chain port connection, input buffer 40, and output First-In-First-Out (FIFO) buffer 74 in support of its contention that "stream communication" requires a "data path" acting like a "queue." PO Resp. 63–66 (citing Ex. 1014, col. 7, ll. 43–57, col. 8, ll. 7–26, col. 9, ll. 2–67); *see* Ex. 1014, Figs. 6–7. We are not persuaded for the same reasons as Petitioner's citation. The cited portions merely describe an exemplary embodiment of the '687 patent that uses memory buffers. Other than the single use of "streams of operands," there is no connection between the '687 patent disclosure and anything pertaining to streaming. *See* Sur-Reply 12 (acknowledging that the '687 patent is "a different patent referring to a different invention pertaining to internet communications"). Certainly, the cited portions do not use or define what is meant by "establishing a stream communication connection between functional units" in the context of the data processing method recited in claim 15.

Patent Owner also relies on U.S. Patent No. 6,339,819 B1 (Ex. 2085, "the '819 patent"), of which the '687 patent is a continuation-in-part, as

incorporated by reference in the '324 patent. PO Resp. 62–64. The '324 patent describes "a representative systolic wavefront operation in the form of a reservoir simulation function" where

> the computation of fluid flow properties are communicated to neighboring cells 710 and, importantly, this computation can be scheduled to eliminate the need for data storage. In accordance with the technique of the present invention, a set of cells can reside in an adaptive processor and the pipeline of computation can extend across multiple adaptive processors. Communication overhead between multiple adaptive processors may be advantageously minimized through the use of MAP™ adaptive processor chain ports as disclosed in U.S. Pat. No. 6,339,819 issued on Jan. 15, 2002 for: "Multiprocessor With Each Processor Element Accessing Operands in Loaded Input Buffer and Forwarding Results to FIFO Output Buffer," assigned to SRC Computers, Inc., assignee of the present invention, the disclosure of which is herein specifically incorporated by this reference.

Ex. 1001, col. 7, ll. 42–46, col. 7, l. 59–col. 8, l. 6, Figs. 7A–7B. According to Patent Owner, the Specification thereby discloses "the concept of using chain ports and a FIFO buffer for chip to chip communications," such that a person of ordinary skill in the art "would recognize that this inter-chip communication concept can be adapted to intra-chip communications between functional units on the same chip—e.g., a FIFO within the chip." PO Resp. 62–63. The cited disclosure is of a "representative" (i.e., exemplary) embodiment where communication "may" occur "between multiple adaptive processors" using chain ports and buffers. *See* Ex. 1001, col. 7, ll. 42–46, col. 7, l. 59–col. 8, l. 6. Again, there is no connection to anything pertaining to streaming data between functional units, as the disclosure does not use the word "stream" at all. The incorporated '819 patent also includes much of the same disclosure as the '687 patent and

suffers from the same problems explained above. *Compare* Ex. 1014, col. 4, l. 15–col. 20, l. 35, Figs. 1–11B, *with* Ex. 2085, col. 4, l. 31–col. 21, l. 43, Figs. 1–11B. We are not persuaded that the incorporated patents support Patent Owner's contention that "stream communication" in claim 15 requires a "data path" acting like a "queue."

Turning next to the prosecution history, we find some guidance as to the scope of "stream communication." Claim 15 was amended to its current form in the applicants' first Office Action response. Ex. 1002, 110. The applicants, however, did not discuss the language of claim 15 in the response and instead attempted to distinguish the cited prior art as applied to parent claim 1. *Id.* at 116–122. Patent Owner points to various statements later in the prosecution history, only one of which addressed the "stream communication" language of claim 15 directly. *See* PO Resp. 66–69 (citing Ex. 1002, 128–129, 147–150, 174–175, 208, 224–225). Specifically, in the subsequent Office Action, the examiner found the "stream communication" limitation taught by a particular reference, noting that the reference

> taught minimiz[ing] interconnections of processing elements and the matrix and vector signal subsets are specifically formed so that they need to be inputted to only one row and one column[] and yet still [are] properly processing systolically along all dimensions within the array. . . . Consequently the stream of communication between functional units is established as the interconnections are made and data is transferred systolically in at least one stream between processors.

Ex. 1002, 128–129. This explanation indicates that establishing a stream communication connection allows data to be "transferred" as a "stream." Because the applicants and examiner do not appear to have ever mentioned a "data path" acting like a "queue" when addressing the "stream communication" limitation, though, we are not persuaded that the

prosecution history supports a narrower interpretation including those limitations, as Patent Owner contends.

Based on what information there is in the intrinsic record, in particular the claim language surrounding "stream communication" and the single reference to the "stream communication" limitation in the prosecution history, we conclude that the plain meaning of "establishing a stream communication connection between functional units" is establishing a connection over which data is streamed between functional units. The interpretation gives meaning to the "stream communication" term and results in claim 15 appropriately having a different scope than claim 1, as claim 15 requires data to be communicated over a particular type of connection that permits streaming, whereas claim 1 would encompass any type of connection. We do not find any support in the intrinsic record for Patent Owner's proposed interpretation requiring a "data path" acting like a "queue."

Patent Owner points to a number of sources of extrinsic evidence in support of its narrower interpretation. As an initial matter, none of those references is a technical dictionary, textbook, or similar source that typically are referenced to help determine a term's ordinary meaning. *See* Tr. 37:15–21, 38:10–17. Instead, Patent Owner refers to unrelated patents and other types of documents. *See* PO Resp. 53–62; Sur-Reply 12–14. For example, Patent Owner cites the '666 patent, which discloses:

> *A stream is a data path between a producer and consumer of data, where the producer and consumer run concurrently.* The path between the producer and consumer is made up of a data connection, a "valid" signal, and a reverse direction "stall" signal. FIG. 1 shows typical signals used in a *stream connection* as is well known and will be recognized by one skilled in the relevant art. The use of a First-In-First-Out buffer 110, or

> "FIFO" buffer, removes the need for tight synchronization between the producer 120 and consumer 130. The producer 120 will generate data values 125 at its own rate, allowing them to accumulate in the FIFO buffer 110. As the FIFO buffer 110 approaches becoming full, it will issue a stall signal 140 to the producer 120 so that it will suspend the generation of data values 125 until the stall signal is released. The consumer 130 will take 150 values 145 from the FIFO buffer at its own rate and as the values 145 are available.

Ex. 2027, col. 2, ll. 39–54 (emphases added); *see* PO Resp. 53–56; Sur-Reply 12. According to Patent Owner and Dr. Homayoun, this disclosure of using a FIFO buffer to account for different data rates and allow the producer and consumer to run concurrently is "entirely consistent with . . . instantiating reconfigurable [routing] resources to seamlessly communicate computed data between processing elements" in claim 1 and "provid[ing] stream communication between those processing elements" in claim 15. PO Resp. 55–56; *see* Ex. 2111 ¶¶ 155–156.

Although sharing an assignee, the '666 patent is unrelated to the '324 patent, has an effective filing date (July 10, 2006) nearly four years after that of the '324 patent (October 31, 2002), and has a different named inventor. *See* Reply 32; Ex. 1013, 2–3. We also do not see any relationship between the '666 patent's description of streaming in connection with a different invention and the '324 patent's systolic implementation. We are not persuaded, therefore, that the '666 patent supports reading "stream communication" in claim 15 to require a "data path" acting like a "queue" as Patent Owner contends, particularly when the intrinsic record provides some indication of a broader interpretation. "[E]xtrinsic evidence may be used only to assist in the proper understanding of the disputed limitation; it may not be used to vary, contradict, expand, or limit the claim language from

how it is defined, even by implication, in the specification or file history." *Bell Atlantic Network Servs., Inc. v. Covad Comm'cns Group, Inc.*, 262 F.3d 1258, 1269 (Fed. Cir. 2001).

Likewise, Patent Owner cites its own product documentation, dated 2002–2007, as "describ[ing] a stream as a data structure that allows flexible communication between concurrent producer and consumer loops" using an internal buffer, as well as a 1993 Argonne National Laboratory paper. PO Resp. 56–59 (citing Ex. 2107, 94–98; Ex. 2028, 31; Ex. 2100 ¶ 79; Ex. 2111 ¶ 157). We do not see—and Patent Owner does not explain sufficiently—why such product documentation shows how a person of ordinary skill in the art would have understood the language of the "stream communication" limitation in claim 15. Moreover, whereas Patent Owner's proposed interpretation is for "stream communication" to be a "data path," the product documentation and paper indicate that a "stream" is a "data structure." *See id.* at 50; Ex. 2107, 94; Ex. 2028, 31. They are different concepts. *See* Ex. 2065, 154–155 (defining "data path" as "[t]he route that a signal follows as it travels through a computer network" and "data structure" as "[a]n organizational scheme, such as a record or array, that can be applied to data to facilitate interpreting the data or performing operations on it"). Finally, Patent Owner quotes a number of other patents and patent application publications as allegedly showing that "stream communication" requires a "data path" acting like a "queue." PO Resp. 59–62. The references use the words "stream" or "streaming" (not "stream communication connection") in the context of describing embodiments of their own disclosed inventions. We find that they have little probative value as well.

We are not persuaded that the extrinsic evidence cited by Patent Owner shows that there was a uniform understanding of persons of ordinary skill in the art of the meaning of the term "stream communication," alone or in the context of a "stream communication connection" between functional units. Indeed, one technical dictionary, submitted but not cited by either party in connection with the term "stream communication," defines "stream" more broadly as "[a]ny data transmission, such as the movement of a file between disk and memory, that occurs in a continuous flow" (as a noun) or "[t]o transfer data continuously, beginning to end, in a steady flow" (as a verb). Ex. 2065, 509. This is consistent with Petitioner's view that the word "stream" in claim 15 "indicates some kind of movement, some kind of transfer of data," as opposed to a particular data structure as Patent Owner contends. *See* Tr. 71:5–15, 72:9–10. Notably, the same technical dictionary that defines "stream" as a "data *transmission*" also defines "queue" as "[a] multi-element data *structure* from which . . . elements can be removed only in the same order in which they were inserted." Ex. 2065, 443, 509 (emphases added). This indicates that a "stream" and a "queue" are distinct concepts, and contradicts Dr. Homayoun's testimony that "[a] person of ordinary skill in the art would understand that the term 'stream' refers to a specific type of structure called a queue." *See* Ex. 2111 ¶ 152.

Applying the broadest reasonable interpretation of the claims in light of the Specification, we give "establishing a stream communication connection between functional units" its plain meaning, namely "establishing a connection over which data is streamed between functional units." We are not persuaded that the extrinsic evidence relied upon by Patent Owner supports a different interpretation. No further interpretation is necessary to resolve the parties' disputes over the asserted grounds of

unpatentability in this proceeding.[17] *See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) ("Because we need only construe terms 'that are in controversy, and only to the extent necessary to resolve the controversy,' we need not construe [a particular claim limitation] where the construction is not 'material to the . . . dispute.'" (citations omitted)).

D. Legal Standards

"Anticipation requires that every limitation of the claim in issue be disclosed, either expressly or under principles of inherency, in a single prior art reference," *Corning Glass Works v. Sumitomo Elec. U.S.A., Inc.*,

---

[17] Although our interpretation differs from those proposed by the parties for "stream communication," we did not preliminarily interpret the term in the Decision on Institution, both parties had the opportunity to present arguments regarding the term in their papers and at the oral hearing, and the issue was discussed extensively during the oral hearing. *See* Tr. 20:21–22:18, 33:19–41:14, 68:19–73:17, 84:3–86:13; *TQ Delta, LLC v. DISH Network LLC*, 929 F.3d 1350, 1355 (Fed. Cir. 2019) (holding that the Board did not improperly "change course" under *SAS Institute, Inc. v. ComplementSoft, LLC*, 825 F.3d 1341 (Fed. Cir. 2016), *rev'd and remanded on other grounds*, *SAS Institute, Inc. v. Iancu*, 138 S. Ct. 1348 (2018), by "construing the limitation in the Final Written Decision because it did not construe the term in its Decision to Institute"); *WesternGeco LLC v. ION Geophysical Corp.*, 889 F.3d 1308, 1328–29 (Fed. Cir. 2018) (holding that "the Board is not bound to adopt either party's preferred articulated construction of a disputed claim term" and "was permitted to issue a new construction in the final written decision given that claim construction was a disputed issue during the proceedings"); *Intellectual Ventures II LLC v. Ericsson Inc.*, 686 F. App'x 900, 906 (Fed. Cir. 2017) (unpublished) ("The Board is not constrained by the parties' proposed constructions and is free to adopt its own construction," but "after the Board adopts a construction, it may not change theories without giving the parties an opportunity to respond.").

868 F.2d 1251, 1255–56 (Fed. Cir. 1989), and that the claim limitations be "arranged or combined in the same way as recited in the claim," *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1371 (Fed. Cir. 2008). However, "the reference need not satisfy an *ipsissimis verbis* test." *In re Gleave*, 560 F.3d 1331, 1334 (Fed. Cir. 2009). "In an anticipation analysis, the dispositive question is whether a skilled artisan would 'reasonably understand or infer' from a prior art reference that every claim limitation is disclosed in that single reference." *Acoustic Tech., Inc. v. Itron Networked Solutions, Inc.*, 949 F.3d 1366, 1373 (Fed. Cir. 2020) (citation omitted). "Expert testimony may shed light on what a skilled artisan would reasonably understand or infer from a prior art reference." *Id.*

A claim is unpatentable for obviousness if, to one of ordinary skill in the pertinent art, "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007) (quoting 35 U.S.C. § 103(a)). The question of obviousness is resolved on the basis of underlying factual determinations, including "the scope and content of the prior art"; "differences between the prior art and the claims at issue"; and "the level of ordinary skill in the pertinent art." *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966). Additionally, secondary considerations, such as "commercial success, long felt but unsolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented. As indicia of obviousness or nonobviousness, these inquiries may have relevancy." *Id.*

A patent claim "is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art." *KSR*,

550 U.S. at 418. An obviousness determination requires finding "both 'that a skilled artisan would have been motivated to combine the teachings of the prior art references to achieve the claimed invention, and that the skilled artisan would have had a reasonable expectation of success in doing so.'" *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1367–68 (Fed. Cir. 2016) (citation omitted); *see KSR*, 550 U.S. at 418 (for an obviousness analysis, "it can be important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does"). "Although the *KSR* test is flexible, the Board 'must still be careful not to allow hindsight reconstruction of references . . . without any explanation as to *how* or *why* the references would be combined to produce the claimed invention.'" *TriVascular, Inc. v. Samuels*, 812 F.3d 1056, 1066 (Fed. Cir. 2016) (citation omitted). Further, an assertion of obviousness "cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *KSR*, 550 U.S. at 418 (quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)); *accord In re NuVasive, Inc.*, 842 F.3d 1376, 1383 (Fed. Cir. 2016) (stating that "'conclusory statements'" amount to an "insufficient articulation[] of motivation to combine"; "instead, the finding must be supported by a 'reasoned explanation'" (citation omitted)); *In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) ("To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.").

E. Anticipation Ground Based on Splash2 (Claims 1, 15, 18, 21, and 22)

1. Splash2

Splash2[18] is a book describing the Splash 2 reconfigurable computer system created by the Supercomputing Research Center in the 1990s. Ex. 1007, xi. "Splash 2 is an attached processor system using Xilinx XC4010 FPGAs as its processing elements." *Id.* "[T]he XC4010 contains a 20 x 20 array of Configurable Logic Blocks (CLBs)." *Id.* at 11.

Figure 2.3 of Splash2 is reproduced below.

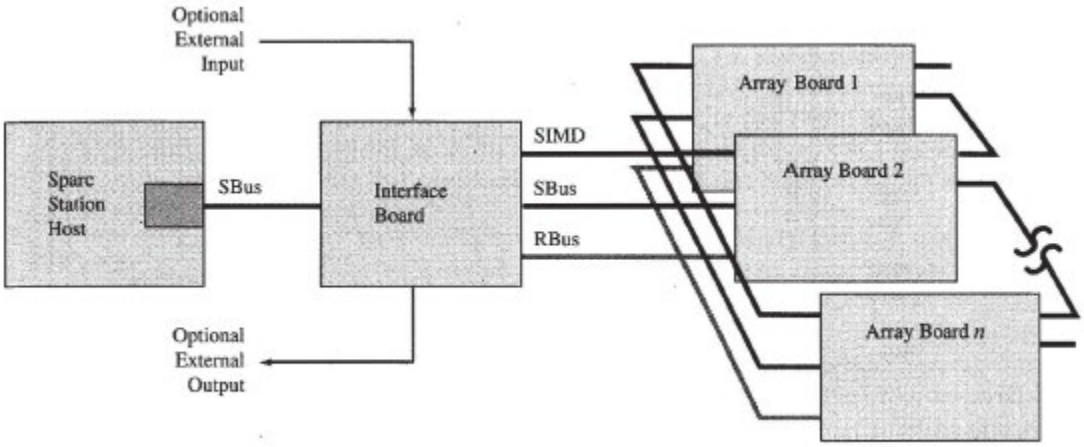


FIGURE 2.3 Splash 2 System Architecture

Figure 2.3 depicts the system architecture of the Splash 2 system, including a set of array boards connected to a SPARCstation 2 host via an interface board. *Id.* at 12–13.

[18] The first 11 pages of Splash2 and a four-page summary article about the Splash 2 system were made of record during prosecution of the '324 patent, but not the remainder of the book, including Chapter 8 on which Petitioner primarily relies. *See* Ex. 1001, code (56); Ex. 1002, 74; Ex. 1060; Pet. 1–2. RaPiD, Jeong, Chunky SLD, and Roccatano were not of record during prosecution of the '324 patent. *See* Ex. 1001, code (56).

Figure 2.4 of Splash2 is reproduced below.

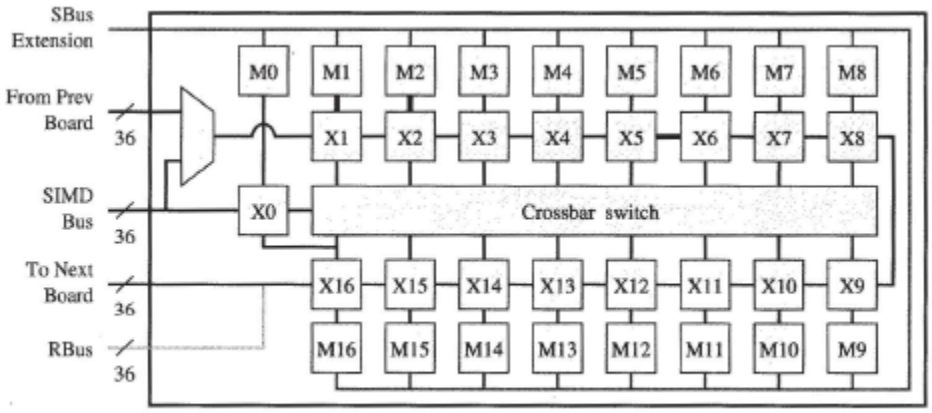


FIGURE 2.4 Array Board Architecture

Figure 2.4 depicts the architecture of one of the array boards. Each array board "contains 17 Xilinx XC40IO FPGA chips as its processing elements. Sixteen of these are connected in a linear array to create a linear data path and the seventeenth provides a broadcast capability to the other 16 chips." *Id.* at 13 (citations omitted). "Viewed as a machine with a linear data path, the [Single Instruction, Multiple Data (SIMD)] Bus can be used to transmit data from the Interface Board to the first FPGA on the first Array Board. The data can then be moved through the linear data path on that board, then to the first FPGA on the second Array Board, and so on." *Id.* at 14. The linear data path is bidirectional. *Id.*

Splash2 describes a number of applications programmed on the Splash 2 system. *Id.* at xi. Chapter 8 describes "two systolic array architectures for [genetic] sequence comparison and their implementations on the Splash 2 programmable logic array." *Id.* at 97. Splash2 discloses that "[i]n comparing two sequences, it is useful to quantify their similarity in terms of a distance measure," and one such measure is the "edit distance between two sequences," which is "the minimum cost of transforming one sequence to the other with a sequence of the following operations: deletion

of a character, insertion of a character, and substitution of one character for another." *Id.* at 98. Each operation has a cost, and the transformation cost is "the sum of the costs of the individual operations." *Id.* Splash2 discloses a "well-known dynamic programming algorithm" for computing edit distance, and describes how two example nucleotide sequences *TCTAGACC* and *GCATAAGC* would be compared using the equations for that algorithm. *Id.* at 98–99.

Splash2 discloses that the edit distance algorithm has "inherent parallelism" in that "each entry in the distance matrix" for comparing each character of the source sequence to each character of the target sequence "depends on adjacent entries," and such parallelism can be "exploited to produce systolic algorithms in which communication is limited to adjacent processors." *Id.* at 98–100, Figs. 8.2–8.3. Specifically, entries of the matrix on the same antidiagonal can be computed in parallel. *Id.* at 100, Fig. 8.4. Splash2 describes two architectures for doing so, one using a bidirectional systolic array and one using a unidirectional systolic array. *Id.* at 100–108. Splash2 discloses that "[b]oth the bidirectional and unidirectional systolic arrays have been implemented on the Splash 2 programmable logic array, with versions for DNA and protein sequences." *Id.* at 104.

a) Unidirectional Array Implementation

Figure 8.9 of Splash2 is reproduced below.

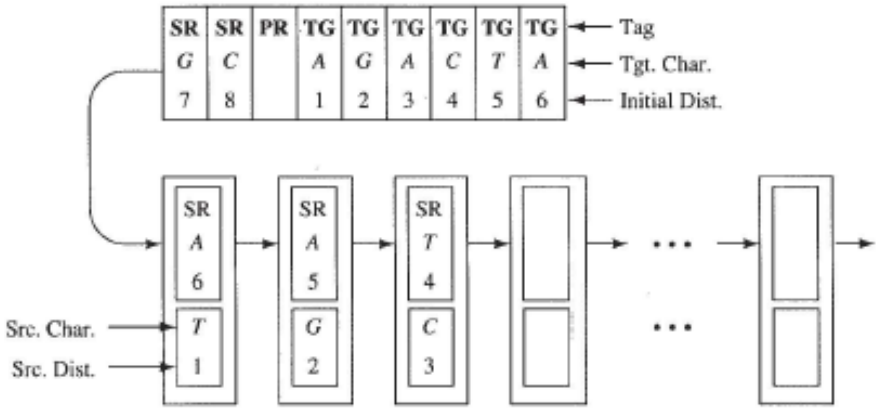


FIGURE 8.9 Data Flow through the Unidirectional Systolic Array. The source sequence is first loaded into the array. The target sequences are then streamed through the array. The tag acts as a simple instruction telling each PE how to process the incoming data. The SR tag instructs an empty PE to load the source character and distance from the input stream. The PR tag marks the end of the source stream. The TG tag signals a target character. Multiple source and target sequences can be carried on the input stream for uninterrupted pipelined processing.

As shown in Figure 8.9, "data flows through the unidirectional array in one direction. The source sequence is loaded once and stored in the array starting from the leftmost [processing element (PE)]. The target sequences are streamed through the array one at a time, separated by control characters." Ex. 1007, 103. "In this configuration, each PE computes the distances in one row of the distance matrix. At each time step, the PEs compute the distances along a single antidiagonal in the distance matrix . . . ." *Id.* at 104. Splash2 discloses that the "DNA version of the unidirectional array," for example, has 248 processing elements that would be capable of comparing millions of characters per second. *Id.* at 107.

Figure 8.12 of Splash2 is reproduced below.

```
loop
    if (TAGin = SR) then
        if (SRCch = Ø) then
            SRCch ← CHRin
            CHRout ← Ø
            DSTout ← PDSTin
        else
            CHRout ← CHRin
        endif
        PDSTout ← PDSTin
    else-if (TAGin = PR) then
        if (SRCch = Ø) then
            DSTout ← PDSTin
        endif
        PDSTout ← DSTin
        CHRout ← CHRin
    else-if (TAGin = TG) then
        if (SRCch ≠ Ø) and (CHRin ≠ Ø) then
                                 { PDSTout+ψ(SRCch,CHRin),
            DSTout ← min         { DSTin+ψ(SRCch,Ø),
                                 { DSTout+ψ(Ø,CHRin)
        else-if (SRCch = Ø) then
            DSTout ← DSTin
        endif
        PDSTout ← DSTin
        CHRout ← CHRin
    endif
    TAGout ← TAGin
endloop
```

FIGURE 8.12 Code executed by each PE in the unidirectional array

Figure 8.12 depicts "[t]he algorithm executed by each PE in the unidirectional array" that "compares one source sequence to a single target sequence." *Id.* at 104–105. As shown above, SRCch and CHRin represent the character of the source sequence and target sequence being processed, respectively; DSTin and PDSTin represent the distance stream being input; and DSTout and PDSTout represent the computed distance stream being output. *Id.* Splash2 discloses that "[a] unidirectional array of length n can compare a source sequence of length at most n to a target sequence of length m in $O(n + m)$ steps." *Id.* at 104.

b) Bidirectional Array Implementation

Figure 8.5 of Splash2 is reproduced below.

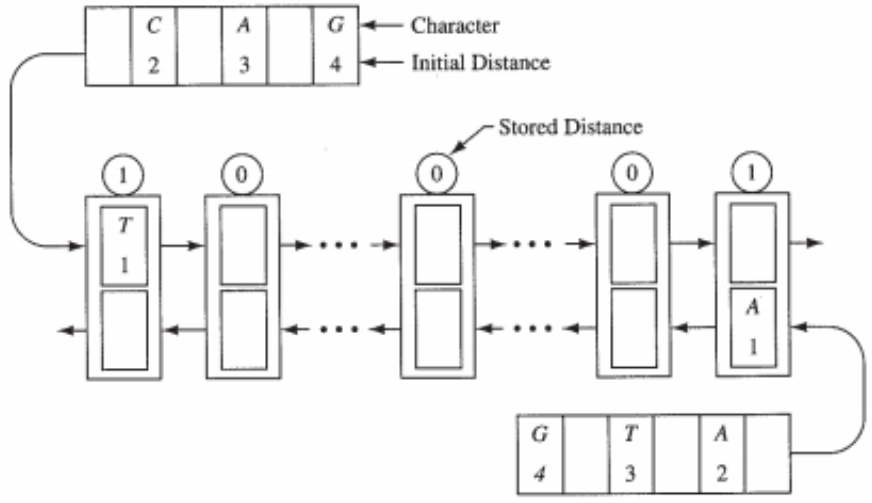


FIGURE 8.5 Data Flow through the Bidirectional Systolic Array. The source and target sequences are streamed through the array in opposite directions. A comparison is performed when a source character and a target character meet in a PE.

As shown in Figure 8.5, "[t]he source and target sequences enter the array on opposite ends and flow in opposite directions at the same speed." Ex. 1007, 102. Splash2 discloses that

> there is one distance stream associated with each character stream. At each step, the contents of the streams represent the characters to be compared and the distances along one of the antidiagonals of the distance matrix. At the end of the computation, the resulting edit distance is transported out of the array on the distance streams.

Id. (footnote omitted). In the bidirectional array implementation, each processing element "computes the distances along a particular diagonal of the distance matrix." *Id.* at 100. Splash2 discloses that the "DNA version of the bidirectional array," for example, has 384 processing elements that would be capable of comparing millions of characters per second. *Id.* at 107.

Figure 8.7 of Splash2 is reproduced below.

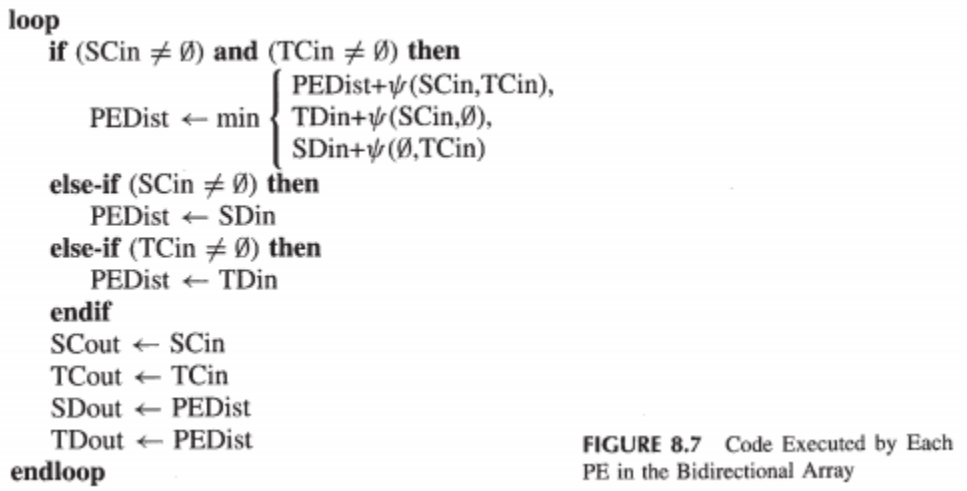


FIGURE 8.7 Code Executed by Each PE in the Bidirectional Array

Figure 8.7 depicts the algorithm executed by each processing element in the bidirectional array. *Id.* at 101. As shown above, SCin and TCin represent the characters of the source sequence and target sequence being processed, respectively; SCout and TCout represent respective characters being output; SDin and TDin represent the source and target distance stream being input, respectively; and SDout and TDout represent the computed source and target distance stream being output, respectively. *Id.*

Splash2 discloses that "[c]omparing sequences of lengths m and n requires at least $2\max(m + 1, n + 1)$ processors. The number of steps required to compute the edit distance and to transport it out of the array is proportional to the length of the array." *Id.* at 103 (footnote omitted). Splash2 explains that the bidirectional array implementation had certain inefficiencies for database search operations, such as the fact that "[a]t each computational step, at most half of the PEs are active" and the source and target sequences were limited to half of the array's length. *Id.* Those inefficiencies were remedied by the unidirectional array implementation according to Splash2. *Id.*

2. Claim 1

Petitioner argues that Splash2 is prior art under 35 U.S.C. §§ 102(a) and 102(b),[19] and explains in detail how the reference discloses every limitation of claim 1, relying on the testimony of Dr. Stone as support. *See* Pet. 21–46;[20] Ex. 1003 ¶¶ 118–179, 184–193, 203–208, 210–216, 220–233, 235–279. Petitioner argues that both the description of the unidirectional array and the description of the bidirectional array in Splash2 disclose certain limitations of claim 1. Pet. 21–46. We address both implementations below.

a) Unidirectional Array Implementation

(1) Undisputed Limitations

Petitioner argues that Splash2 discloses a "method for data processing," as recited in claim 1, where "the characters of a source sequence of genetic information are compared against a target sequence of genetic information and an edit distance is calculated based on that comparison." Pet. 29–30. Petitioner contends that Splash2 discloses

[19] Petitioner provides evidence supporting its contention that Splash2, as well as RaPiD, Jeong, Chunky SLD, and Roccatano, are prior art printed publications under 35 U.S.C. §§ 102(a) and 102(b). *See* Pet. 21–22, 56, 68 (citing Ex. 1065 ¶¶ 8–12; Ex. 1066 ¶¶ 37–46, 75, 77, 80–92; Ex. 1068 ¶¶ 4–6, 9–12); -1602 Pet. 63 (citing Ex. 1066 ¶¶ 47–55; Ex. 1068 ¶¶ 7–12); -1603 Pet. 67 (citing Ex. 1066 ¶¶ 65–73; Ex. 1067); Ex. 1007, Cover 6 (Library of Congress stamp for Splash2). Patent Owner does not assert otherwise in its Response, and we agree that the references are prior art for the reasons stated by Petitioner.

[20] Petitioner asserts many of the same arguments, such as those regarding claim 1, in all three of its Petitions. *See* Pet. 21–46; -1602 Pet. 22–52; -1603 Pet. 25–54. Where applicable, we cite the papers in Case IPR2018-01601 for convenience.

a reconfigurable computing system comprising "at least one reconfigurable processor" (i.e., the array boards with FPGAs) with "a plurality of functional units" (i.e., the matrix of CLBs configured as processing elements), and "transforming" an "algorithm" (i.e., the edit distance algorithm disclosed in Splash2) into a "calculation" (i.e., calculation of the edit distance for two sequences). *Id.* at 29–32. According to Petitioner, the calculation is "systolically implemented" in the unidirectional array implementation because "each Processing Element 'computes the distances in one row of the distance matrix,'" with "a target sequence streamed through the array, and distance information continually output to the downstream neighboring Processing Element (via DSTout and PDSTout)," and a character comparison is triggered by the arrival of data rather than a program counter or clock. *Id.* at 32–34. We agree, given our interpretation of the term "systolic" as "the characteristic of rhythmically computing and passing data directly between processing elements in a manner that is transport triggered (i.e., by the arrival of a data object) rather than a program counter or clock driving movement of the data." *See supra* Section II.C.1.

Petitioner further contends that Splash2 discloses "instantiating" at least two functional units to perform the calculation (i.e., loading information into the FPGAs to create the necessary processing elements) where "only functional units needed to solve the calculation are instantiated" because the FPGA components "clear all configuration memory before any configuration occurs" and the components instantiated for a particular calculation are tailored to that combination. Pet. 34–35. Petitioner argues that "each instantiated functional unit . . . interconnects with each other instantiated functional unit . . . based on reconfigurable routing resources within the at least one reconfigurable processor as established at

instantiation," as shown by the target sequence streaming through the processing elements in Splash2. *Id.* at 35–36. Finally, "systolically linked lines of code" of the calculation are "instantiated as clusters of functional units" (i.e., the processing elements on each FPGA) according to Petitioner because "each Processing Element continually computes distance information for each character comparison and passes it to a neighboring Processing Element" "without storage of that information in memory between processing elements, without the data being driven by a clock or program counter, and triggered by the arrival of tags in the data stream." *Id.* at 36–38.

Patent Owner in its Response does not dispute that Splash2 discloses the above limitations of claim 1. Petitioner's analysis for each of the limitations, supported by the testimony of Dr. Stone, which we credit, is persuasive. *See id.* at 29–38; Ex. 1003 ¶¶ 153–179, 184–193, 203–208, 210–216, 220–233, 235–242.

(2) Disputed Limitation: "Computational Loops"

Claim 1 recites the following "computational loop" limitations:

> utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and
>
> substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a second computational loop.

Petitioner argues that a person of ordinary skill in the art would have understood that the loop/endloop code, shown in Figure 8.12 above, "constitutes a time step of a 'data dimension' comprising multiple time steps because each such instance of code executes in a single time step, and the

systolic computation comprises multiple time steps." Pet. 38 (citing Ex. 1003 ¶¶ 245–246). According to Petitioner, a processing element executes the code "in a single time step," taking in certain inputs from the preceding processing element (e.g., TAGin, CHRin, distance information DSTin and PDSTin) and passing along outputs (e.g., TAGout, CHRout, distance information DSTout and PDSTout) to the downstream neighboring processing element for "each character comparison" performed. *Id.* at 26, 28, 33, 37–38, 46; *see* Ex. 1003 ¶¶ 135, 139, 167, 189, 192, 247–259.

Petitioner provides the following annotated version of Figure 8.13 of Splash2 (Pet. 39).

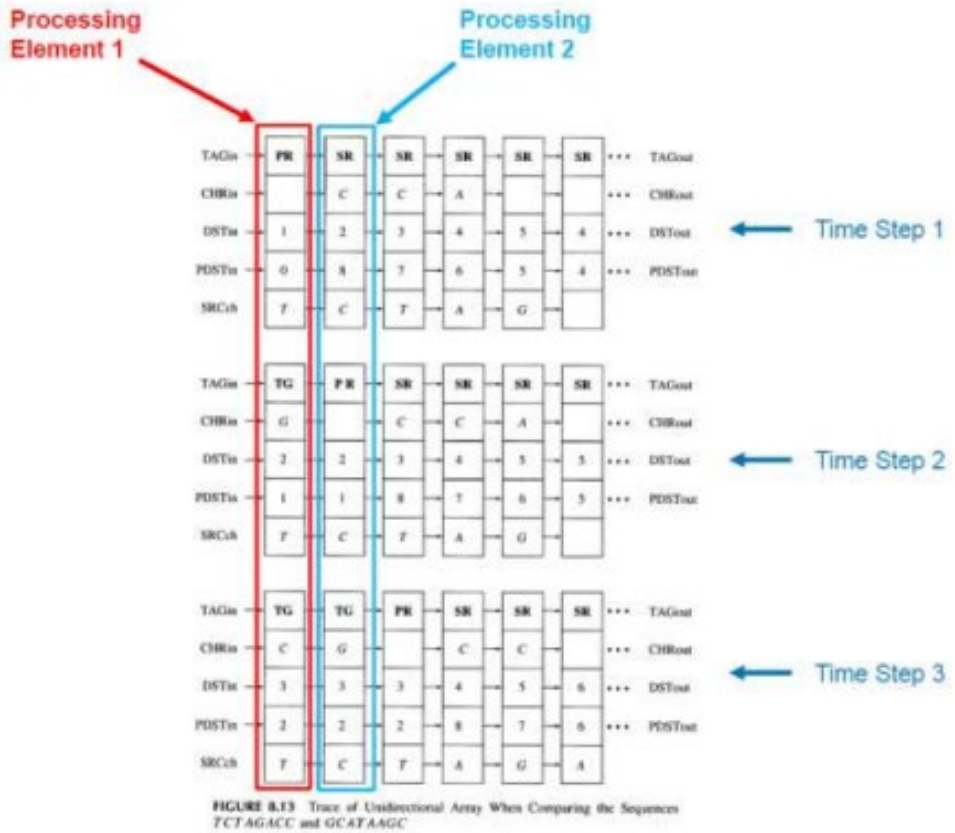


FIGURE 8.13 Trace of Unidirectional Array When Comparing the Sequences *TCTAGACC* and *GCATAAGC*

Annotated Figure 8.13 depicts the source sequence *TCTAGACC* pre-loaded in the processing elements (row SRCch), and the characters of the target sequence *GCATAAGC* streamed through the processing elements

systolically (row CHRin). *Id.* at 39–41 (citing Ex. 1003 ¶¶ 247–259). For example, at Time Step 2, Processing Element 1 compares the first target character *G* to the first source character *T*. *Id.* at 40. At Time Step 3, (1) Processing Element 1 compares the second target character *C* to the first source character *T*, and (2) Processing Element 2, which has received the first target character *G* streamed through the array, compares it to the second source character *C*. *Id.* Petitioner argues that, as shown in the example above, Processing Element 2 operates on a "subsequent data dimension" (i.e., comparing the second source character to the streamed target characters—the second row of the distance matrix) forming a "first computational loop" (i.e., "Processing Element 2 executing the loop instructions disclosed in Figure 8.12"); and Processing Element 1 operates on a "previous data dimension" (i.e., comparing the first source character to the streamed target characters—the first row of the distance matrix) forming a "second computational loop" (i.e., "Processing Element 1 executing the loop instructions disclosed in Figure 8.12"). *Id.* at 41 (emphases omitted). According to Petitioner, "[t]he same loop instructions are executed in each Processing Element, but they are still 'first' and 'second' computational loops because they are necessarily different instances of those loop instructions." *Id.* Petitioner's analysis for the "computational loop" limitations of claim 1, supported by the testimony of Dr. Stone, which we credit, is persuasive. *See* Ex. 1003 ¶¶ 243–259.

Patent Owner's arguments are largely premised on its proposed interpretation of "computational loop" as requiring repeated execution "per datum," which we do not adopt. *See* PO Resp. 80–87 (citing Ex. 2111 ¶¶ 127, 194–209; *supra* Section II.C.2. Patent Owner asserts that "[t]here is no disclosure [in Splash2] of looping or repeating of a computation multiple

times *for each data* until a condition is met or a number of repetitions has been satisfied," as allegedly represented in Figure 4B of the '324 patent. PO Resp. 80–81, 84–85 (emphasis added). We disagree that the claim requires such a process, and instead interpret "computational loop" to mean "a set of computations that is executed repeatedly, either a fixed number of times or until some condition is true or false." *See supra* Section II.C.2.

Patent Owner further contends that the code shown in Figure 8.12 of Splash2 does not illustrate a "computational loop." PO Resp. 81–87. Patent Owner first points to the if/else-if conditional statements in Figure 8.12, which select an execution path based on whether a condition is true or false. *Id.* at 82–83. Petitioner in its analysis, however, relies on the overall loop/endloop aspect of the code, not the if/else-if statements within that code. Pet. 26–28, 38–41. Petitioner first introduces the bidirectional array implementation and argues that Splash2 "discloses in Figure 8.7 the code executed by each processing element, . . . which *includes a loop for computing the edit distance* (i.e., the minimum cost of transforming one sequence to the other) between characters," reproducing the code of Figure 8.7 (which includes the notations "loop" and "endloop"), then does the same for the unidirectional array code in Figure 8.12 (which also includes "loop" and "endloop"). *Id.* at 26, 28 (emphasis added).

What we must determine is how a person of ordinary skill in the art would have understood the code disclosed in Splash2, in particular "loop" and "endloop." The parties and their respective declarants have different views. *See id.* at 26–28, 38–41; PO Resp. 83–87; Reply 38–40. Patent Owner relies on Dr. Homayoun, who testifies that "[t]o be defined as a loop, a loop exit condition must be specified. This is generally specified either in the first line of the code (begin loop and then condition) or the last line of

[the] loop (loop end and then condition)." Ex. 2111 ¶ 200. According to Dr. Homayoun, the code in Figure 8.12 lacks an "exit condition" at the beginning or end, so if the code was a loop as Petitioner contends, it "would run forever." *Id.* ¶¶ 201–202. "This means nothing would be calculated by the code because the first piece of data would enter each PE and then run forever. Nothing would be passed to the next PE." *Id.* ¶ 203. Therefore, "[t]he only reasonable interpretation of this pseudocode is to assume that something replaces the 'loop-endloop' syntax so that the pseudocode executes once then passes the data to the next PE." *Id.* ¶ 204. In that case, the processing element executing the code "would never evaluate the same data more than once" and would not perform a "computational loop" under Patent Owner and Dr. Homayoun's claim interpretation. *Id.* ¶¶ 205–208.

Dr. Stone testifies that the loop/endloop code in Figure 8.12 represents "a loop for computing the edit distance (i.e., the minimum cost of transforming one sequence to the other) between characters." Ex. 1003 ¶¶ 135, 139, 166–169. At a particular time step, the processing element "execut[es] the loop instructions disclosed in Figure 8.12" to compare the target character that was streamed in to the source character loaded in that processing element. *Id.* ¶¶ 139, 247–257. The processing element then repeatedly does the same for each input character of the target sequence (i.e., "multiple comparisons"). *Id.* Responding to Dr. Homayoun's testimony regarding the loop/endloop code,[21] Dr. Stone points out that the repeated

[21] Again, we do not agree with Patent Owner that Petitioner's Reply and Dr. Stone's reply testimony are improper, as they respond directly to Patent Owner's arguments in the Response and Dr. Homayoun's testimony regarding how a person of ordinary skill in the art would have understood the loop/endloop code in Splash2. *See* Sur-Reply 8–9; Reply 15–18, 38–40; Ex. 1076 ¶¶ 9–19; 37 C.F.R. § 42.23(b).

comparisons result in the code being executed a fixed number of times because the target and source sequences are of limited length (*m* and *n*, respectively) and each processing element executes the loop instructions until it completes computing distances along the diagonal matrix. Ex. 1076 ¶¶ 12–13.

Based on our review of the full trial record, we find that Dr. Stone's reading, explained in detail in his declarations, is most consistent with the disclosure of Splash2 and that a person of ordinary skill in the art would have read the reference in the manner he describes. Figure 8.12 is labeled as a "loop," indicating that the code within the "loop" and "endloop" designations is executed repeatedly by the processing element. Splash2 expressly describes the reason *why* it is executed repeatedly: the target sequence is "streamed through the array" so that a single incoming target character can be compared to the single source character loaded in that processing element. Ex. 1007, 103. The target sequence and source sequence both contain a large set of characters of "length *m*" and "length *n*," respectively. *Id.* at 104. The target sequence can be any "length," whereas the "maximum length" of the source sequence is limited by the "length" of the array. *Id.* Naturally, when the end of the *m*-length target sequence is reached, no more characters can be streamed in and the loop would complete for that processing element (i.e., after a fixed number of times executing the code). *See* Ex. 1076 ¶ 13. Thus, a person of ordinary skill in the art would have understood the code in Splash2 to loop and have an exit condition—it executes repeatedly (i.e., once for each target character streamed through the array) a fixed number of times (i.e., until the last target character in the target sequence is streamed through and compared).

Indeed, Splash2 expressly discloses limits on the number of times a processing element executes the code. In the unidirectional array implementation, "each PE computes the distances in one row of the distance matrix," and "[a]t each time step, the PEs compute the distances along a single antidiagonal in the distance matrix," indicating that processing would complete when there are no more distances to compute. *See* Ex. 1007, 104; *see also id.* at 100 (disclosing that each processing element "computes the distances along a particular diagonal of the distance matrix" in the bidirectional array implementation), 102 ("At the end of the computation, the resulting edit distance is transported out of the array on the distance streams."). Dr. Stone's understanding of the loop/endloop code is consistent with the text of Splash2 and the "loop" and "endloop" designations shown in the code itself. *See* Ex. 1076 ¶¶ 9–13.

Importantly, we also note that Splash2's description of the unidirectional array implementation would disclose the "computational loop" limitations even under Patent Owner's proposed interpretation. Because the loop/endloop code is executed repeatedly for the same source character (e.g., source character *T* for Processing Element 1 and source character *C* for Processing Element 2 in annotated Figure 8.13 above), comparing it to each target character as it is streamed in, computations are executed repeatedly "per datum" (i.e., for the same source character "datum") a fixed number of times. *See id.* ¶¶ 9–11, 14.

Patent Owner's next argument is that Splash2 does not disclose "utilizing" the two "instantiated functional units" on the reconfigurable processor to operate on data dimensions forming "computational loops," as recited in claim 1, because "even if Splash2 could be read to disclose two computational loops, it does not disclose instantiating those loops in the

FPGAs of Splash2." PO Resp. 87–90; Sur-Reply 17–19. According to Patent Owner, the external Sun workstation, not the FPGAs, would "handle any looping." PO Resp. 88 (emphasis omitted). As support, Patent Owner relies on testimony from its declarants (Ex. 2111 ¶ 209; Ex. 2164 ¶¶ 42–43) and one paragraph from the thesis of Richard Peyton Halverson, Jr., Ph.D., "The Functional Memory Approach to the Design of Custom Computing Machines," Ph.D. diss., University of Hawaii, 1994 (Ex. 2167, "the Halverson thesis"). PO Resp. 88.

We disagree. Splash2 expressly discloses that "[b]oth the bidirectional and unidirectional systolic arrays have been implemented *on the Splash 2 programmable logic array*, with versions for DNA and protein sequences." Ex. 1007, 104 (emphasis added). "In the DNA version of the unidirectional array, each of the 16 array FPGAs (X1 to X16) holds 14 PEs." *Id.* at 107. Splash2's discussion of the two implementations for the edit distance algorithm repeatedly refers to the functionality of an individual processing element (PE). *See, e.g.*, *id.* at 100 ("There are several ways to map the edit distance computation onto a linear systolic array. We describe two such mappings."), 104 ("At each time step, the PEs compute the distances along a single antidiagonal in the distance matrix . . . . The algorithm executed by each PE in the unidirectional array is listed in Figure 8.12."). Thus, the processing elements that perform the looping are instantiated within the FPGAs in Splash2. *See* Ex. 1003 ¶¶ 238, 247–259; Ex. 1076 ¶¶ 18–19.

The Halverson thesis does not support a different reading. It discloses:

> Splash 2 contains one or more boards each with an array of 16 well connected XILINX 4010 chips [Gokhale and Minnich,

> 1993]. The architecture does an excellent job supporting pipelined and SIMD processor configurations. Splash 2, for example, can be programmed in dbC, which is a superset of C used on other SIMD computers. The dbC preprocessor produces C that runs on the Sun and VHDL which define SIMD processors with an instruction set tailored to the application, one or more of which fit into each XILINX chip. When the actual program executes, looping is still handled in the Sun, which transmits SIMD instructions to the Splash 2 board(s).

Ex. 2167, 37–38. This isolated reference is to the Splash 2 system and how it "can be programmed in [Data-parallel Bit-serial C (dbC)]." *See id.* It does not mention or relate to the particular systolic array architectures and edit distance calculations described in Chapter 8 of Splash2. Further, whereas the paragraph above references the Splash 2 system being programmed in dbC, Splash2 describes programming the edit distance calculations in VHDL. *See id.* at 37–38, 208; Ex. 1007, 70, 106; Ex. 1074, 94–95, 97 ("Gokhale and Minnich" paper referenced in the Halverson thesis describing dbC and VHDL); Ex. 1003 ¶ 172; Ex. 1076 ¶¶ 15–17.

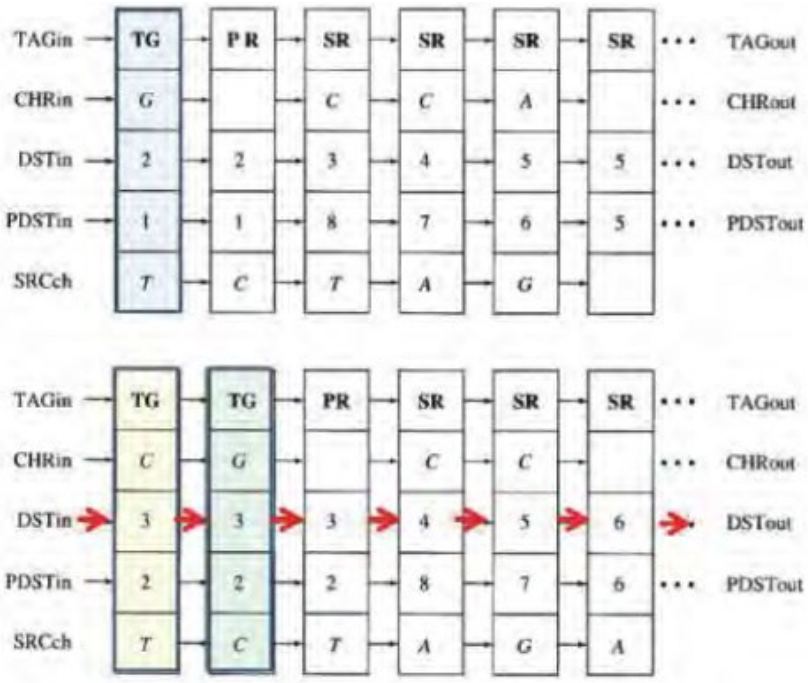
Dr. Stone conducted a detailed analysis of Splash2, citing specific figures and discussion in the reference, and explained why a person of ordinary skill in the art would have understood Splash2 to disclose utilizing two functional units to operate on different data dimensions of a calculation forming two "computational loops," as recited in claim 1. After reviewing the full trial record, we credit that testimony. For the foregoing reasons, we find persuasive the arguments and evidence presented by Petitioner and Dr. Stone and, therefore, find that Splash2, in its description of the unidirectional array implementation, discloses the "computational loop" limitations of claim 1.

(3) Disputed Limitation: "Seamlessly"

Claim 1 recites that "said systolic implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops" (the "seamlessly" limitation).

Petitioner contends that the computational loops in Splash2 "execute concurrently" and "pass computed data seamlessly between" the computational loops because each processing element "execute[s] an instance of" the loop instructions, shown in Figure 8.12 above, and "each Processing Element directly communicates computed distance information to a neighbor." Pet. 46 (citing Ex. 1003 ¶¶ 274–279). Dr. Stone explains that "[t]he output of a functional unit *passes directly* to the input of the next functional unit with *no intervening gaps or additional interfaces*." Ex. 1003 ¶ 276 (emphases added); *see also id.* ¶¶ 186, 239 (stating that "distance information for each character comparison" is passed to "a neighboring Processing Element . . . without storage of that information in memory *between* processing elements" (emphasis added)). Each functional unit takes in "DSTin from the previous functional unit in the series" and outputs "DSTout for the next functional unit in the series." *Id.* ¶ 276 (citing Ex. 1007, 104, Fig. 8.10).

Dr. Stone provides the following annotated version of a portion of Figure 8.13 of Splash2 (*id.* ¶ 277).



The annotated portion above of a trace of the unidirectional array depicts "the seamless communication of the DST data in the systolic calculation as a sequence of red arrows." *Id.* Dr. Stone explains that

> [t]he red arrows indicate . . . that the computed output DSTout from a functional unit is *directly connected* to the next functional unit input DSTin. Similarly, all output data from one functional unit are seamlessly connected to the corresponding inputs of the functional unit in the sequence. There are *no intermediate interfaces between modules* to translate the output of one module into a form where it can be used as input data to the next module. Such interfaces, if they were to exist, would be "seams" in the communication links between adjacent functional units.

Id. (emphases added). As explained above, we interpret "pass computed data seamlessly between said computational loops" to mean "communicate computed data directly between functional units that are calculating computational loops." *See supra* Section II.C.3. Petitioner's analysis for the

"seamlessly" limitation of claim 1, consistent with that interpretation and supported by the testimony of Dr. Stone, which we credit, is persuasive. *See* Ex. 1003 ¶¶ 272–277, 279.

Patent Owner makes four arguments in its Response. First, Patent Owner contends that in passing computed data between the computational loops operating on data dimensions of the calculation, claim 1 "specifically require[s] a sequence of computations *per datum*," but the loop/endloop code in Splash2 "merely selects an execution path for the processor to execute once per datum." PO Resp. 96–97. We disagree for the reasons stated above. *See supra* Sections II.C.2, II.E.2.a.2.

Second, Patent Owner argues that "the FPGAs [in Splash2] must communicate with the Sun workstation (which is handling any looping) through the Sbus" and the "boundary between the FPGAs and the workstation . . . clearly constitutes a 'seam,'" citing the same declarant testimony and Halverson thesis discussed above in connection with the "computational loop" limitations. PO Resp. 96–97, 100. Again, we disagree that the Sun workstation handles the looping discussed in Chapter 8 for the edit distance calculations. *See supra* Section II.E.2.a.2.

Third, Patent Owner asserts that "Splash2 is, at best, ambiguous on whether memory is used to store the results from each processing element after each time step to preserve it for output and later use." PO Resp. 97–100 (citing Ex. 2111 ¶¶ 210–219). Patent Owner contends that "storage is likely necessary to preserve the values calculated at each timestep," where the computed data is overwritten at each time step and the overall edit distance is based on the repeated calculations and overwriting (rather than just the final time step). *Id.* at 98; *see* Ex. 2111 ¶ 210 (identifying, for example, SDout, TDout, and PEDist from the bidirectional array

implementation). Patent Owner points out that Splash2 has "local memory at each FPGA for storage purposes," which "can be used for storage of results." PO Resp. 98–99 (citing Ex. 1007, 88, 95, 102 n.3; Ex. 1035, 5; Ex. 2156, 205–206). According to Patent Owner, "it is equally (if not more) plausible for a [person of ordinary skill in the art] to interpret Splash2 to use the local memory due to the known timing problems in systolic systems." *Id.* at 99–100.

We are not persuaded that Splash2 is ambiguous in the manner Patent Owner contends. The disclosure and figures discussed above indicate that computed distance information is communicated directly between processing elements. In the context of the unidirectional array implementation, for example, a processing element takes in DSTin, executes the loop/endloop code to compute DSTout, and outputs DSTout to the next processing element in the series. *See* Ex. 1007, 104 (Fig. 8.10 depicting the inputs and outputs for the unidirectional array PE), 105 (Fig. 8.12 showing how DSTout is calculated), 106 (Fig. 8.13 depicting direct connections between processing elements); Ex. 1003 ¶¶ 275–279; *see also* Ex. 1007, 101–102 (Figs. 8.6–8.8 showing inputs of SDin and TDin and output of the computed PEDist as SDout and TDout in either direction for the bidirectional array implementation). Claim 1 recites that the systolic implementation enables the computational loops to "pass computed data seamlessly between said computational loops." There is no indication in Splash2 itself that there is any memory or other intervening structure *between* the disclosed processing elements. That data may be stored temporarily *within* a processing element (e.g., in a storage register) does not automatically mean that memory between the processing element and another processing element is used for communicating that data. *See*

Ex. 1007, 104 (stating that the unidirectional array PE "stores two distances, DST and PDST," but never stating that memory between the PE and another PE is used to do so).

Fourth, Patent Owner argues that Splash2 does not disclose the "seamlessly" limitation even under Petitioner's proposed interpretation because "the Xilinx FPGAs . . . clearly contain structure (such as the buffered switch matrix) within the internal routing resources to connect processing elements." PO Resp. 100–105 (citing Ex. 1035, 28–31; Ex. 2078, 19–29, 32–34, 37–41, 46–51, 59–65, 70). Again, just because an FPGA has memory does not mean that when functional units are instantiated within the FPGA, memory is necessarily placed *between* functional units or used to transfer data from one functional unit to another. *See* Reply 25, 47; *supra* Section II.C.3.

For the foregoing reasons, we find persuasive the arguments and evidence presented by Petitioner and Dr. Stone and, therefore, find that Splash2, in its description of the unidirectional array implementation, discloses the "seamlessly" limitation of claim 1.

b) Bidirectional Array Implementation

Petitioner's explanation for why a person of ordinary skill in the art would have understood Splash2's description of the bidirectional array implementation to disclose the limitations of claim 1 is similar to its explanation for the unidirectional array implementation. Petitioner argues that the calculation is "systolically implemented" in the bidirectional array implementation because "each Processing Element 'computes the distances along a particular diagonal of the distance matrix,'" where the calculation is based on the source and target sequences streamed in from the processing

element's "upstream and downstream neighbors" and "[t]he resulting distance information for each character comparison is continually passed to the upstream and downstream neighboring Processing Elements, without storage of that information in memory between processing elements." Pet. 32.

With respect to the "computational loop" limitations, Petitioner argues that a person of ordinary skill in the art would have understood that the loop/endloop code shown in Figure 8.7 above "constitutes a time step of a 'data dimension' comprising multiple time steps because each such instance of code executes in a single time step, and the systolic computation comprises multiple time steps." *Id.* at 38 (citing Ex. 1003 ¶¶ 245–246). Petitioner provides the following annotated version of Figure 8.8 of Splash2 (*id.* at 43).

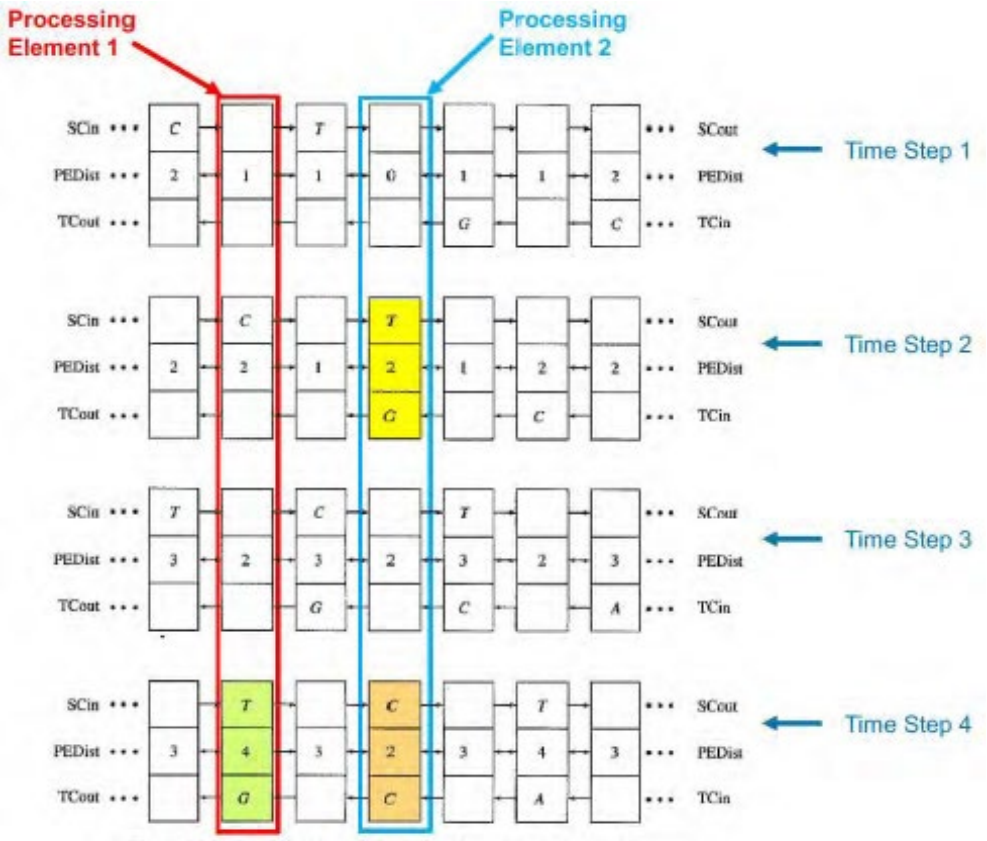


FIGURE 8.8 Trace of Bidirectional Array When Comparing the Sequences *TCTAGACC* and *GCATAAGC*

Annotated Figure 8.8 depicts the characters of the source sequence *TCTAGACC* streamed left to right through the processing elements (row SCin), and the characters of the target sequence *GCATAAGC* streamed right to left through the processing elements (row TCin). *Id.* at 42–46 (citing Ex. 1003 ¶¶ 260–271). For example, at Time Step 2, Processing Element 2 compares the first target character *G* to the first source character *T* (shown in yellow). *Id.* at 43. At Time Step 4, (1) Processing Element 2 compares the second target character *C* to the second source character *C* (shown in orange), and (2) Processing Element 1, which has received the first target character *G* streamed through the array, compares it to the third source character *T* (shown in green). *Id.*

Petitioner argues that, as shown in the example above, Processing Element 2 operates on a "subsequent data dimension" (i.e., comparing streamed target and source characters on a particular diagonal of the distance matrix) forming a "first computational loop" (i.e., "Processing Element 2 executing the loop instructions disclosed in Figure 8.7"); and Processing Element 1 operates on a "previous data dimension" (i.e., comparing streamed target and source characters on a particular diagonal of the distance matrix) forming a "second computational loop" (i.e., "Processing Element 1 executing the loop instructions disclosed in Figure 8.7"). *Id.* at 44–45 (emphases omitted). As with the unidirectional array implementation, "the loop instructions executed in each Processing Element are first and second computational loops because they are necessarily different instances of those loop instructions." *Id.* at 45.

With respect to the "seamlessly" limitation, Petitioner argues that each processing element "execute[s] an instance of" the loop instructions shown in Figure 8.7 above and "each Processing Element directly communicates

computed distance information to a neighbor." *Id.* at 46 (citing Ex. 1003 ¶¶ 274–279). Dr. Stone explains that the same passing of data occurs in the unidirectional and bidirectional array implementations, where "[t]he output of a functional unit *passes directly* to the input of the next functional unit with *no intervening gaps or additional interfaces*." Ex. 1003 ¶¶ 276, 278 (emphases added). Dr. Stone provides the following annotated version of a portion of Figure 8.8 of Splash2 (*id.* ¶ 278).

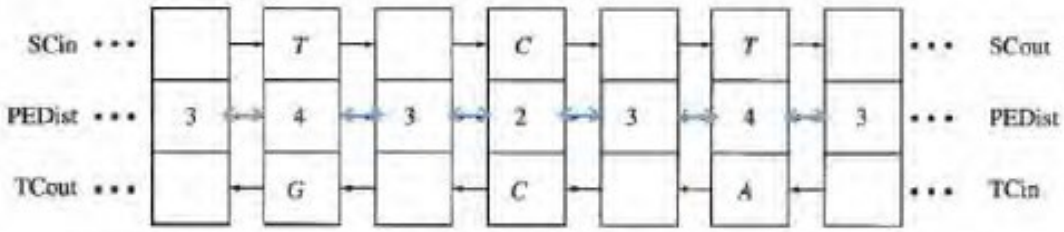


FIGURE 8.8 Trace of Bidirectional Array When Comparing the Sequences *TCTAGACC* and *GCATAAGC*

The annotated portion above of a trace of the bidirectional array depicts the seamless communication of distance information. *Id.* Dr. Stone explains that

> [t]he blue arrows indicate that the computed output PEDist from a functional unit is directly connected to the next functional unit input PEDist in both directions. The code for the Bidirectional implementation discloses that the computed PEDist is conveyed to the left on output TDout, and to the right on output SDout.

Id. (citing Ex. 1007, 101, Fig. 8.6).

Patent Owner argues in its Response that Splash2 does not disclose the "computational loop" and "seamlessly" limitations of claim 1. PO Resp. 80–90, 96–105. Patent Owner's arguments apply to both the unidirectional and bidirectional array implementations described in Splash2,

and we disagree with those arguments for the reasons explained above.[22] *See id.*; *supra* Section II.E.2.a. Petitioner's analysis of how Splash2's description of the bidirectional array implementation discloses the limitations of claim 1, supported by the testimony of Dr. Stone, which we credit, is persuasive. *See* Ex. 1003 ¶¶ 153–179, 184–193, 203–208, 210–216, 220–233, 235–246, 260–276, 278–279. For the reasons stated above, we find persuasive the arguments and evidence presented by Petitioner and Dr. Stone and, therefore, find that Splash2, in its description of the bidirectional array implementation, discloses the limitations of claim 1.

c) Conclusion

We are persuaded by Petitioner's explanations and supporting evidence regarding both the unidirectional and bidirectional implementations of Splash2, and find that Splash2 discloses every limitation of claim 1 under both of Petitioner's theories. Petitioner has proven, by a preponderance of the evidence, that claim 1 is anticipated by Splash2 under 35 U.S.C. §§ 102(a) and 102(b).

[22] The loop/endloop code executed by a processing element operates similarly in both implementations. We note that although the unidirectional array implementation would meet the "computational loop" limitation even under Patent Owner's proposed interpretation (because the code is executed repeatedly for the same source character), the bidirectional array implementation would not (because the target and source sequences are both streamed through).

3. Claims 18, 21, and 22

Dependent claims 18, 21, and 22 limit the "calculation" of parent claim 1 to a particular type. Claim 18 recites that the calculation comprises a "search algorithm for data mining," claim 21 recites that it comprises a "genetic pattern matching function," and claim 22 recites that it comprises a "protein folding function." Petitioner argues that a person of ordinary skill in the art would have understood Splash2's edit distance calculation to be each type of calculation. -1603 Pet. 54–57. For example, Petitioner argues that the edit distance calculation is a "genetic pattern matching function" because it "compares characters of two genetic sequences." *Id.* at 55; *see* Ex. 1007, 100–104; Ex. 1003 ¶¶ 126–140, 462–465. Patent Owner does not argue separately dependent claims 18, 21, and 22 in its Response, only disputing Petitioner's contentions with respect to parent claim 1. PO Resp. 77–90, 96–105. We have reviewed Petitioner's contentions regarding claims 18, 21, and 22, which are consistent with the disclosure of Splash2 and supported by the testimony of Dr. Stone, and are persuaded that Petitioner has proven, by a preponderance of the evidence, that dependent claims 18, 21, and 22 are anticipated by Splash2 under 35 U.S.C. §§ 102(a) and 102(b). *See* -1603 Pet. 54–57.

4. Claim 15

Claim 15 recites that the "instantiating" step of parent claim 1 includes "establishing a stream communication connection between functional units." Petitioner argues that Splash2 discloses establishing such a connection "between adjacent Processing Elements in both the unidirectional and bidirectional edit distance calculation because it discloses streaming target and source characters between processing elements both

unidirectionally and bidirectionally." -1602 Pet. 52–53. Patent Owner responds that Splash2 does not disclose the limitation because it does not disclose a "queue" between processing elements or "signaling" for the processing elements to interact with such a queue. PO Resp. 77–80. Patent Owner's arguments are premised on its proposed interpretation of "stream communication" as "a data path that acts like a queue connecting via the reconfigurable routing resources a producer and a consumer of data that operate concurrently." *Id.* at 50, 77–80. As explained above, we disagree and instead interpret "establishing a stream communication connection between functional units" to mean "establishing a connection over which data is streamed between functional units." *See supra* Section II.C.4.

Given our interpretation, Petitioner has made a sufficient showing that Splash2 discloses the limitation. In the unidirectional array implementation, "[t]he source sequence is loaded once" and "target sequences are streamed through the array one at a time, separated by control characters." Ex. 1007, 103. "[D]ata flows through the unidirectional array in one direction." *Id.* In the bidirectional array implementation, "[t]he source and target sequences are streamed through the array in opposite directions." *Id.* at 101–102. "[T]here is one distance stream associated with each character stream. At each step, the contents of the streams represent the characters to be compared and the distances along one of the antidiagonals of the distance matrix." *Id.* at 102. Thus, Splash2 discloses in the various implementations establishing a connection over which data (e.g., characters of the source and target sequences, computed distance information) is streamed between functional units. Petitioner has proven, by a preponderance of the evidence, that dependent claim 15 is anticipated by Splash2 under 35 U.S.C. §§ 102(a) and 102(b).

F. Obviousness Ground Based on Splash2 (Claims 1, 15, 18, 21, and 22)

1. Obviousness Contentions

Petitioner argues, in the alternative, that claims 1, 15, 18, 21, and 22 would have been obvious over Splash2. Pet. 47–52; -1602 Pet. 53–59; -1603 Pet. 57–63. Petitioner argues that "[t]o the extent one might argue" that the chapters of Splash2 cannot be considered together for purposes of anticipation, a person of ordinary skill in the art would have found claim 1 to be obvious based on the full disclosure of Splash2. Pet. 47–48. Petitioner further contends that "[t]o the extent one might argue" that Splash2 does not disclose four limitations of claim 1 (i.e., the "transforming" step, portions of the "instantiating" step, the limitation that "only functional units needed to solve the calculation are instantiated," and the limitation that "each instantiated functional unit at the at least one reconfigurable processor interconnects with each other instantiated functional unit"), a person of ordinary skill in the art would have been motivated to modify Splash2 to include them. *Id.* at 48–52. For example, Petitioner argues that if Splash2 does not disclose the limitation that "only functional units needed to solve the calculation are instantiated," doing so would have been obvious because the FPGAs in Splash2 are "cleared of any configuration data before a new configuration can be entered," the arrays are customized for a particular application, and a person of ordinary skill in the art would have been motivated to instantiate only the processing elements necessary to solve a calculation to "avoid the additional work" of writing code for processing elements that "performed no function [and] served no purpose." *Id.* at 50–51. Splash2 is a single reference, and Patent Owner does not dispute that the chapters of Splash2 can be considered together. *See* Ex. 1007, 104 (Chapter 8 indicating that "the bidirectional and

unidirectional systolic arrays have been implemented on the Splash 2 programmable logic array," which was described in earlier chapters). Nor does Patent Owner dispute that Splash2 discloses the four identified limitations. We find that the four limitations are disclosed by Splash2. *See supra* Section II.E.2.a.1. Thus, we need not address the alternative arguments made by Petitioner.

Patent Owner argues that Splash2 fails to disclose certain other limitations of claims 1 and 15. *See supra* Sections II.E.2, II.E.4. We disagree, find that Splash2 expressly teaches all limitations of claims 1, 15, 18, 21, and 22, and conclude that Petitioner has proven anticipation by a preponderance of the evidence. *See id.* Accordingly, absent a persuasive showing of secondary considerations, which we discuss below, the claims also would have been obvious based on Splash2. *See Realtime Data, LLC v. Iancu*, 912 F.3d 1368, 1373 (Fed. Cir. 2019) ("[I]t is well settled that a disclosure that anticipates under § 102 also renders the claim invalid under § 103, for anticipation is the epitome of obviousness." (citations and internal quotation marks omitted)).

2. Secondary Considerations of Nonobviousness

Patent Owner also argues that secondary considerations of nonobviousness demonstrate that claims 1–5, 7–9, 15, 17, 18, and 20–24 would not have been obvious to a person of ordinary skill in the art. PO Resp. 121–125. Although some of the challenged claims are anticipated by—and, therefore, would have been obvious based on—Splash2, because the parties refer to the challenged claims collectively in their arguments regarding secondary considerations, we do so as well and now address those

arguments as applied to all of the claims challenged as obvious. *See id.*; Reply 52–55.

"In order to accord substantial weight to secondary considerations in an obviousness analysis, the evidence of secondary considerations must have a nexus to the claims, *i.e.*, there must be a legally and factually sufficient connection between the evidence and the patented invention." *Fox Factory, Inc. v. SRAM, LLC,* 944 F.3d 1366, 1373 (Fed. Cir. 2019) (citations and internal quotation marks omitted). "The patentee bears the burden of showing that a nexus exists." *Id.* "To determine whether the patentee has met that burden, we consider the correspondence between the objective evidence and the claim scope." *Id.* A patentee is entitled to a rebuttable presumption of nexus "when the patentee shows that the asserted objective evidence is tied to a specific product and that product 'embodies the claimed features, and is coextensive with them.'" *Id.* (citation omitted). However, "[a] finding that a presumption of nexus is inappropriate does not end the inquiry into secondary considerations." *Id.* "To the contrary, the patent owner is still afforded an opportunity to prove nexus by showing that the evidence of secondary considerations is the 'direct result of the unique characteristics of the claimed invention.'" *Id.* at 1373–1374 (citation omitted).

"Where the offered secondary consideration actually results from something other than what is both claimed and *novel* in the claim, there is no nexus to the merits of the claimed invention," meaning that "there must be a nexus to some aspect of the claim not already in the prior art." *In re Kao*, 639 F.3d 1057, 1068–69 (Fed. Cir. 2011). On the other hand, there is no requirement that "objective evidence must be tied exclusively to claim elements that are not disclosed in a particular prior art reference in order for

that evidence to carry substantial weight." *WBIP, LLC v. Kohler Co.*, 829 F.3d 1317, 1331 (Fed. Cir. 2016). A patent owner may show, for example, "that it is the claimed combination as a whole that serves as a nexus for the objective evidence; proof of nexus is not limited to only when objective evidence is tied to the supposedly 'new' feature(s)." *Id.* at 1330. Ultimately, the fact finder must weigh the secondary considerations evidence presented in the context of whether the claimed invention as a whole would have been obvious to an ordinarily skilled artisan. *Id.* at 1331–32.

As objective evidence of nonobviousness, Patent Owner cites the declarations of Dr. Homayoun (Ex. 2111), Mr. Huppenthal (Ex. 2100), and Dr. El-Ghazawi (Ex. 2164); the deposition transcript of Stephen M. Trimberger, Ph.D. in Cases IPR2018-01599 and IPR2018-01600 (Ex. 2076); the deposition transcript of Dr. Stone in Case IPR2018-01594 (Ex. 2066); and two publications authored by, among others, Dr. El-Ghazawi and one of the authors of Splash2, Duncan Buell, Ph.D.: Duncan Buell *et al.*, "High-Performance Reconfigurable Computing," *IEEE Computer Society*, pp. 23–27 (Mar. 2007) (Ex. 2166), and Tarek El-Ghazawi *et al.*, "The Promise of High-Performance Reconfigurable Computing," *IEEE Computer Society*, pp. 69–76 (Feb. 2008) (Ex. 2165). PO Resp. 121–125.

Patent Owner asserts that there was commercial success and praise by others for "DirectStream's patented products (SRC-6, SRC-6e)," which involved "instantiating the reconfigurable resources necessary for an algorithm so as to maximize the speed data can be passed between different looping portions of the program" running concurrently "to achieve a particular systolic system that processes data without the associated delays of communications protocols or the Von Neumann bottleneck." *Id.* at

14–15, 125. Patent Owner, however, does not provide any explanation or analysis demonstrating that its "SRC-6" or "SRC-6e" products were used to perform the methods recited in any of the challenged claims, or are coextensive with the claimed methods. Mr. Huppenthal discusses "SRC-6 products and the SRC-7 . . . under development" and testifies that "[p]roduction systems would incorporate . . . AI and Machine Learning as described in [the '324 patent]" and "Army and Air Force applications would make extensive use of program loops and streams as defined in [the '324 patent]." Ex. 2100 ¶¶ 80, 83. Importantly, though, Mr. Huppenthal never refers to the claims in his testimony and acknowledged that he did not perform "any kind of comparison of the claims [of the '324 patent] to any particular system." *See id.* ¶¶ 80–87; Ex. 1073, 106:14–107:14, 108:17–109:5; PO Opp. 2 (acknowledging that Mr. Huppenthal only provides "high-level, general testimony regarding . . . specific hardware features of interest" to "customers who purchased SRC-6 computers" and "does not specifically discuss any applications any customers ran on the SRC-6"). Therefore, we find that a presumption of nexus is inappropriate.

Nor does the other evidence cited by Patent Owner establish a nexus between DirectStream's products and any of the challenged claims. Patent Owner cites one of the publications and Dr. Homayoun's declaration mentioning the "SRC-6," "SRC-6E," and "SRC-7" products, but does not point to anything in either document explaining how the products embody the claims. *See* PO Resp. 125 (citing Ex. 2165, 2–7, Fig. 4; Ex. 2111 ¶¶ 108–109, 118–119). Absent some explanation of how the cited products worked, there is no basis to say that they were used to practice the methods recited in the challenged claims. Accordingly, we find that Patent Owner

has not established a sufficient nexus between the claims and the alleged commercial success and industry praise.

Patent Owner further asserts that there was a long-felt but unmet need for "very fast processing of large volumes of data, and improvements in conventional systems focused on reducing the delays associated with chip-to-chip communications protocols and also the Von Neumann processing bottleneck." *Id.* at 123 (citing Ex. 2164 ¶¶ 17–25; Ex. 2166, 3–5; Ex. 2100 ¶¶ 36–41; Ex. 2111 ¶¶ 40–106). According to Patent Owner, there was skepticism in the industry that "reconfigurable processors could adequately satisfy these needs compared to other well-established solutions in the same technology space." *Id.* (citing Ex. 2076, 129:24–130:20; Ex. 2066, 168:9–169:4, 179:6–13, 197:8–11).

To support a conclusion of nonobviousness, an alleged long-felt need must have been a persistent one that was recognized by those of ordinary skill in the art, must not have been satisfied by another before the challenged patent, and must have been satisfied by the claimed invention. *Perfect Web Techs., Inc. v. InfoUSA, Inc.*, 587 F.3d 1324, 1332–33 (Fed. Cir. 2009). Similarly, evidence showing that "skilled artisans were initially skeptical about the [claimed] invention" is relevant to nonobviousness. *Metabolite Labs., Inc. v. Laboratory Corp. of Am. Holdings*, 370 F.3d 1354, 1368 (Fed. Cir. 2004). We have reviewed all of the cited evidence and do not find it persuasive, as Patent Owner does not provide any explanation establishing a nexus to the challenged claims. Patent Owner does not explain in its Response, for example, how exactly the claimed methods solved the identified problems of speed and reducing communication delays. Nor does Patent Owner explain how the cited testimony from Dr. Trimberger and Dr. Stone allegedly shows skepticism in the industry that the reconfigurable

processor approach *of the challenged claims* would reduce such issues, as opposed to mere views on reconfigurable processors in general. *See* PO Resp. 123.

Patent Owner also contends that others in the industry failed to solve the problems allegedly solved by the '324 patent. PO Resp. 124–125 (citing Ex. 2167, 12–20; Ex. 2174, 4; Ex. 2164 ¶¶ 32–43; Ex. 2100 ¶¶ 30–31, 36–41; Ex. 2111 ¶¶ 179–184; Ex. 2066, 168:9–169:4, 179:6–13, 197:8–11; Ex. 2076, 129:24–130:20). We are not persuaded for similar reasons. Patent Owner does not explain in sufficient detail how the cited evidence demonstrates a "failure of others to do *that which the patent claims*." *See id.* at 124 (quoting *Para-Ordnance Mfg., Inc. v. SGS Importers Int'l, Inc.*, 73 F.3d 1085, 1094 (Fed. Cir. 1995)) (emphasis added). Without further explanation, we are not persuaded that Patent Owner's evidence of the failure of others supports the nonobviousness of the challenged claims.

Finally, Patent Owner points to the publications authored by Dr. Buell and Dr. El-Ghazawi in 2007 and 2008 (well after the effective filing date of the '324 patent, October 31, 2002) that "survey[ed] . . . the state of the art" and discussed "DirectStream's SRC-6 and SRC-6e systems" but "did not include [the Splash 2 system] as an example of a viable [High-Performance Computing Technology (HPRC)] system." *Id.* at 124–125 (citing Exs. 2165, 2166). Patent Owner does not point to any authority—and we are not aware of any—for the proposition that a reference's *silence* as to a particular implementation indicates a failure of that implementation to solve a particular problem. *See* Tr. 66:1–67:11. Because the cited documents do not mention the Splash 2 system, we find that they do not support Patent Owner's contention that the system represented a failure of others.

For the reasons explained above, we conclude that Patent Owner's evidence purportedly showing commercial success, industry praise, long-felt need, skepticism in the industry, and failure of others does not weigh in favor of nonobviousness of the challenged claims.

3. Conclusion

Based on all of the evidence of record, including evidence of secondary considerations of nonobviousness submitted by Patent Owner, we determine that claims 1, 15, 18, 21, and 22 would have been obvious based on Splash2 under 35 U.S.C. § 103(a).

G. Obviousness Ground Based on Splash2 and RaPiD (Claims 8 and 9)

Petitioner asserts that dependent claims 8 and 9 are unpatentable over Splash2 and RaPiD. Pet. 55–67; *see* Ex. 1003 ¶¶ 330–369. Claim 8 recites that the calculation comprises "a JPEG image compression calculation" and claim 9 recites that the calculation comprises "an MPEG image compression calculation." With respect to claim 8, for example, Petitioner argues that RaPiD teaches a discrete cosine transform (DCT) "implemented on a systolic array of a reconfigurable computing system called the Reconfigurable Pipelined Datapath, or 'RaPiD.'" Pet. 56. RaPiD discloses that "[t]he datapaths constructed in RaPiD are linear arrays of functional units communicating in mostly nearest-neighbor fashion" and "[s]ystolic algorithms, for example, map very well into RaPiD datapaths." *Id.* (quoting Ex. 1009, 106). RaPiD explains that one application implemented on the disclosed system was a "2-D DCT . . . used in JPEG/MPEG data compression to convert an image from the spatial domain to the frequency domain." *Id.* at 56–57 (quoting Ex. 1009, 110).

Petitioner asserts that a person of ordinary skill in the art would have combined the teachings of Splash2 and RaPiD such that "the functionality of the cell of RaPiD's Figure 10 would be instantiated into Processing Elements of the Splash2 Unidirectional Systolic Array," where "[t]he Processing Elements would perform the matrix multiply calculations described in RaPiD, and then pass the resulting data to the next Processing Element in a systolic fashion." *Id.* at 58–59, 64–65. Petitioner provides numerous reasons why a person of ordinary skill in the art would have been motivated to do so. *Id.* at 64–67.

Patent Owner does not dispute that RaPiD teaches JPEG and MPEG image compression calculations, as recited in claims 8 and 9, but makes other arguments. First, Patent Owner argues that the cited calculations in RaPiD do not constitute a "computational loop," as recited in parent claim 1. PO Resp. 93–95.

Petitioner asserts in the Petition that each processing element ("cell") in the RaPiD system carries out a "computational loop" in performing the 2-D DCT calculation, citing the following annotated version of Figure 10 of RaPiD provided by Dr. Stone (Pet. 63; Ex. 1003 ¶ 357).

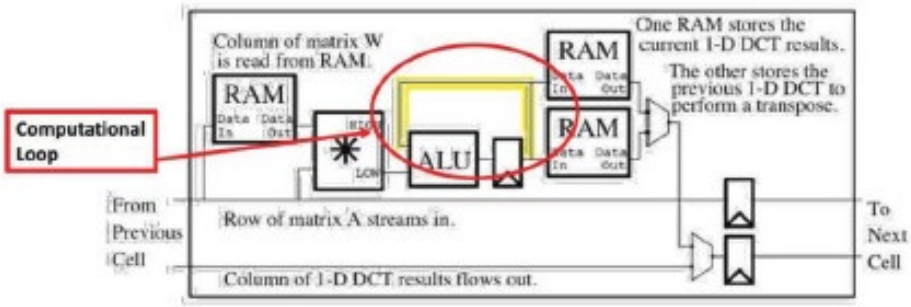


Figure 10: *Netlist for one cell of 2-D DCT. The top pipelined bus streams in the* **A** *matrix while the bottom bus streams out resulting 1-D DCT, transposed. The top bus also streams the* **W** *columns into the local memories prior to the computation.*

Annotated Figure 10 depicts a netlist for one cell of the 2-D DCT. Dr. Stone provides a detailed explanation of how RaPiD performs its calculations. Ex. 1003 ¶¶ 333–360. In relevant part for purposes of addressing Figure 10, "[a] 2-D DCT can be decomposed into two sequential 1-D DCTs." *Id.* ¶ 333 (quoting Ex. 1009, 110). The 1-D DCT can be carried out in "cells" of an 8-cell pipeline by performing matrix multiply calculations on two matrices: input vector A and weight W. *Id.* ¶¶ 334–336. RaPiD discloses that the 2-D DCT can be reduced to the following dot product equations (Ex. 1009, 111).

$$z_{mj} = \sum_{n=0}^{N-1} a_{mn} w_{nj}, \qquad (5)$$

and thus

$$y_{ji} = \sum_{m=0}^{N-1} z_{mj} w_{mi} \qquad (6)$$

As illustrated in the equations above, "both z_{mj} and y_{ji} are equivalent to . . . matrix multiplies," but because "the z_{mj} values are produced in row-major order but required in column-major order, the results from the z_{mj} DCT must be transposed prior to computing y_{ji},." *Id.* Figure 8 of RaPiD is reproduced below.

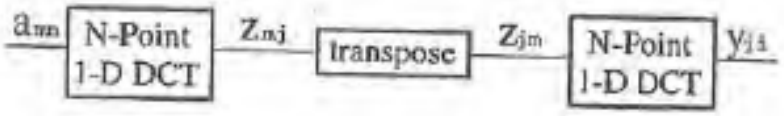


Figure 8: 2-D $N \times N$ DCT

Figure 8 depicts the 2-D DCT as two 1-D DCT calculations with a transpose in between. Dr. Stone explains in detail how the calculation of the dot product equations above requires multiple iterations of (1) multiplications between elements of a first matrix (*a* or *z*) with elements of a second matrix (*w*), and (2) addition of the result of that multiplication to a running sum

from previous iterations. Ex. 1003 ¶¶ 351–359; Ex. 1076 ¶¶ 21–22.[23] According to Dr. Stone, the portion of annotated Figure 10 above highlighted in yellow shows "the output [of] the [arithmetic logic unit (ALU) being] looped back to the ALU input." Ex. 1003 ¶ 357. That path is what "loops the intermediate, or running, sum from the output of the ALU to its input so that it may be used in the next iteration of the DCT calculation." Ex. 1076 ¶ 25.

With respect to Figure 10, the 2-D DCT partitions an M x N image into 8 x 8 sub-images and performs two 1-D DCTs for each, using an 8 x 8 weight matrix, with the transpose in between. Ex. 1003 ¶¶ 337–341 (citing Ex. 1009, 111, Figs. 8–10). RaPiD discloses that

> [s]ince a 2-D DCT performs two multiplies by the same weight matrix, *W* is loaded only once: one column per cell in both the first 8 cells and last 8 cells. The transpose in between matrix multiplies is performed with two local memories per cell: one to store products of the current sub-image and the other to store the products of the *previous* sub-image. During the computation of

[23] Again, we do not agree with Patent Owner that Petitioner's Reply and Dr. Stone's reply testimony are improper, as they respond directly to Patent Owner's argument in the Response that RaPiD teaches a "bypass or forwarding path" rather than a "computational loop." *See* Sur-Reply 8–9; Reply 40–44; Ex. 1076 ¶¶ 20–27; 37 C.F.R. § 42.23(b). Petitioner did not change its theory as to how RaPiD teaches a "computational loop," relying in both its Petition and in its Reply on the portion of Figure 10 highlighted in yellow above. *See Apple Inc. v. Andrea Elecs. Corp.*, 949 F.3d 697, 706–707 (Fed. Cir. 2020) (concluding that a reply was proper where it did not change the "legal ground" asserted in the petition and "relie[d] on the same [teaching] from the same prior art reference to support the same legal argument"); *Chamberlain Grp., Inc. v. One World Techs., Inc.*, 944 F.3d 919, 925 (Fed. Cir. 2019) ("Parties are not barred from elaborating on their arguments on issues previously raised."). Patent Owner also had the opportunity to cross-examine Dr. Stone about his reply declaration. *See* Ex. 2176.

> the current sub-image, the transpose of the previous sub-image computation is passed to the next 8 cells. The datapath for one RaPiD cell of a 2-D DCT is shown in Figure 10.

Ex. 1009, 111. Thus, as shown in Figure 10, "each cell in the array stores one column of weight data and receives both a stream of row image data and a stream of 1-D DCT data. The cell calculates the DCT for the current sub-image and passes on the DCT data for the previous sub-image along with row image data directly to the next cell." Ex. 1003 ¶ 343; *see* Ex. 1009, 111, Fig. 10 (depicting the "[c]olumn of matrix W" stored in RAM and the "[r]ow of matrix A" and "[c]olumn of 1-D DCT results" received from the previous cell).

Patent Owner responds that the portion of annotated Figure 10 above highlighted in yellow is not a "computational loop" but rather a "bypass or forwarding path," which "support[s] back to back execution of operations without stall, by forwarding (or bypassing) the output of an ALU to an input of the same or other ALU." PO Resp. 94–95 (citing Ex. 2029 ¶¶ 41–48; Ex. 2111 ¶¶ 234–241; Ex. 2043, 301; Ex. 2044, 2, 6–7). According to Patent Owner, Dr. Stone also conceded during cross-examination that Figure 10 shows a bypass or forwarding path. *Id.* at 95 (citing Ex. 2064, 201:21–202:1).

Based on our review of the full record after trial, we find Patent Owner's arguments unavailing. Dr. Stone's explanation for how the calculation of the dot product equations requires adding products to a running sum is consistent with the figures and text of RaPiD, and we see no structure in Figure 10 other than the portion highlighted in yellow that would be capable of performing such functionality. Nor does Patent Owner point to any. RaPiD teaches, via the looping of the ALU output in Figure 10, a set

of computations that is executed repeatedly a fixed number of times (i.e., until the end of the matrices is reached). *See supra* Section II.C.2.

We also are not persuaded that Dr. Stone conceded otherwise during cross-examination. He was asked simply whether "Figure 10 show[s] *a* bypass path or a forwarding path" and responded "Yes." Ex. 2064, 201:21–202:2 (emphasis added). He was not asked to—and did not—identify any particular structure shown in Figure 10. Nor was he asked specifically whether the yellow highlighted portion is a bypass or forwarding path. Moreover, there is a bypass or forwarding path shown in Figure 10 that is not the yellow highlighted portion, namely the bottom wire labeled "Column of 1-D DCT results flows out" where DCT data from the previous cell is passed along to the next cell, bypassing the ALU functionality and memory shown in the top portion of the figure. Ex. 1003 ¶ 343; Ex. 1076 ¶ 27. Dr. Stone states that that wire was what he was referring to during cross-examination. Ex. 1076 ¶ 27. We see no contradiction between Dr. Stone's declaration testimony and his statement that a bypass or forwarding path exists in Figure 10.

Second, Patent Owner argues that RaPiD does not teach passing computed data "seamlessly" between computational loops because Figure 10 "shows storage of results in memory (RAM) before being passed onto the next cell." PO Resp. 105–106. As an initial matter, we note that Petitioner's position, supported by the testimony of Dr. Stone, is that a person of ordinary skill in the art would have had reason to combine the references' teachings such that "each Processing Element of the Splash2 Unidirectional Systolic Array would carry out the calculations of RaPiD Figure 10" and computed data would be "communicated directly from one Processing Element to the next." Pet. 64 (emphases omitted); *see* Ex. 1003 ¶¶ 344–346,

349, 351, 360, 363. We find, for the reasons explained above, that in the unidirectional array implementation of Splash2, computed data is passed "seamlessly" (i.e., communicated directly, without intervening memory or other structures) between processing elements performing the computational loops. *See supra* Section II.E.2.a.3. Thus, Patent Owner's argument attacking RaPiD individually with respect to the "seamlessly" limitation is not persuasive. *See In re Mouttet*, 686 F.3d 1322, 1332–33 (Fed. Cir. 2012) (holding that the test for obviousness is "what the combined teachings of the references would have suggested to those having ordinary skill in the art"); *Medichem, S.A. v. Rolabo, S.L.*, 437 F.3d 1157, 1166 (Fed. Cir. 2006) (explaining that in an obviousness analysis, "the prior art must be considered *as a whole* for what it teaches"); *In re Merck & Co., Inc.*, 800 F.2d 1091, 1097 (Fed. Cir. 1986) ("Non-obviousness cannot be established by attacking references individually where the rejection is based upon the teachings of a combination of references."). Regardless, though, the RAM that Patent Owner points to in Figure 10 is part of the cell (i.e., functional unit) itself, not between cells.[24] *See* Pet. 60–64; Reply 48; Ex. 1009, 106 ("RaPiD is a linear array of functional units which is configured to form a mostly linear computational pipeline. This array of functional units is divided into identical cells which are replicated to form a complete array."); Ex. 2064, 201:11–18.

[24] As explained above, for purposes of determining whether memory is within a processing element or between processing elements, the boundaries of a processing element are not arbitrary, but rather depend on the disclosure of the particular reference and how the reference describes the processing element and the communication of data to and from the processing element. *See supra* Section II.C.3. In this case, Figure 10 of RaPiD shows the "[n]etlist for one *cell* of [the] 2-D DCT." Ex. 1009, 111 (emphasis added).

Third, Patent Owner raises various challenges to all of Petitioner's obviousness grounds collectively, in particular to Petitioner's reliance on the testimony of Dr. Stone. Patent Owner argues, for example, that Dr. Stone's testimony is conclusory, grounded in hindsight bias, fails to disclose the underlying facts or data on which his opinions are based under 37 C.F.R. § 42.65(a), fails to "articulate reasons why or how a [person of ordinary skill in the art] would combine the references" or consider whether the asserted combinations were "feasible," and "assumes only the benefits of his combinations and ignores any problems or drawbacks." PO Resp. 17–29, 110–115, 119–121 (emphasis omitted). Many of Patent Owner's arguments are premised on Petitioner failing to "acknowledge" purported "shortcomings" of Splash2, such as the "need for looping to be performed by [the] Sun workstation" and Splash2's lack of seamless passing of data and stream communication. *See, e.g.*, *id.* at 27, 112–115, 120. Because we are not persuaded that Splash2 has any of those alleged deficiencies, those arguments are not persuasive. *See supra* Section II.E.2. Other arguments as to how Dr. Stone allegedly failed to understand the '324 patent, prior art, and problems and issues in the art are premised on his testimony in a different proceeding challenging the '687 patent and belied by the detailed explanations provided in his declarations, as explained herein. *See, e.g.*, PO Resp. 17–19, 21, 23–26, 113–114, 119–120 (citing Ex. 2066). We have reviewed Patent Owner's arguments and cited evidence in the record and disagree with the remainder of Patent Owner's assertions as well. We address the combination of Splash2 and RaPiD as representative.

Petitioner explains in detail what teachings of the two references it is relying on as teaching the various limitations of claims 8 and 9 and, importantly, explains exactly how a person of ordinary skill in the art would

have combined those teachings, i.e., instantiating the cell functionality of Figure 10 in the processing elements of the Splash2 unidirectional array implementation. Pet. 58–59, 64–65. Petitioner asserts that the references are combinable in an obviousness combination because they are both analogous art to the '324 patent and provides multiple reasons a person of ordinary skill in the art would have had for making the combination. *Id.* at 65–66. For example, RaPiD expressly cites the Splash 2 system as a "very successful example[]" of a field-programmable custom computing machine. Ex. 1009, 106 (citing a reference describing the Splash 2 system), 115; *see* Pet. 64–67. Splash2 also expressly states that the Splash 2 system would be advantageous for image processing applications (of which JPEG and MPEG image compression are examples), suggesting that implementing the RaPiD calculations in the Splash 2 system "would have been successful and efficient." *See* Ex. 1007, 141–142, 162 ("Even though Splash 2 was not designed specifically for image processing, this platform possesses architectural properties that make it well suited for the computation and data transfer rates that are characteristic of this class of problems. Furthermore, the price/performance of this system makes it a competitive alternative to conventional real-time image processing systems."); Pet. 66–67. Other documentation indicates that image compression techniques using DCT were being used at the time, such that an ordinarily skilled artisan would have been motivated to find "more efficient and powerful solutions" to do so (e.g., by implementing the RaPiD DCT calculation in the Splash 2 system). *See* Ex. 1049, 73; Pet. 66.

Petitioner's assertions are supported by the testimony of Dr. Stone. *See* Ex. 1003 ¶¶ 330–369. That testimony is not based on hindsight bias, but rather the content of both asserted references, which Dr. Stone cites

extensively in his analysis. *See id.* He explains the exact combination of teachings and the reasons for making the combination. *See id.* The reasons are not conclusory; they are explained in sufficient detail and premised on express statements in Splash2 and RaPiD connecting the two disclosures and suggesting that their combination would be advantageous for multiple reasons. Patent Owner never addresses those identified reasons to combine in its papers or explains in any way why they are factually incorrect. Finally, we note that Dr. Stone stated during cross-examination that in addition to advantages of the various obviousness combinations, he considered potential disadvantages where relevant to the combination. *See, e.g.*, Ex. 2066, 145:9–146:5. We do not find any evidence in the record that combining RaPiD's teaching of the DCT calculation with Splash2's teaching of the unidirectional array implementation would have had any disadvantages that would have outweighed Petitioner's stated advantages.

Based on all of the evidence of record, including evidence of secondary considerations of nonobviousness submitted by Patent Owner, *see supra* Section II.F.2, we determine that claims 8 and 9 would have been obvious based on Splash2 and RaPiD under 35 U.S.C. § 103(a).

H. Obviousness Ground Based on Splash2 and Jeong (Claim 20)

Petitioner asserts that dependent claim 20 is unpatentable over Splash2 and Jeong. Pet. 67–77; *see* Ex. 1003 ¶¶ 425–461. Claim 20 recites that the calculation comprises "an encryption algorithm." Petitioner relies on Jeong for this limitation, arguing that Jeong teaches "a systolic implementation of a modular multiplication algorithm for encryption suitable for implementation in FGPAs." Pet. 68–69 (citing Ex. 1061, 214, Fig. 2(a)). Petitioner provides numerous reasons why a person of ordinary

skill in the art would have been motivated to combine the teachings of Jeong and Splash2, including, for example, that Jeong "expressly cites the systolic structures as the platform to which its computing algorithms are mapped, and discloses the intent to make use of FPGAs as a possible means for implementation." *Id.* at 75–77 (citing Ex. 1061, 211).

Patent Owner does not dispute that Jeong teaches "an encryption algorithm." Instead, Patent Owner argues that Jeong does not teach two "computational loops," as recited in parent claim 1, because the reference does not teach repeating a computation multiple times "for each piece of data." PO Resp. 92–93. Patent Owner's argument is premised on its proposed interpretation requiring repeated execution "per datum," which we do not adopt. *See supra* Section II.C.2. Also, as explained above, we find that Splash2 teaches two "computational loops." *See supra* Section II.E.2.

Likewise, with respect to the "seamlessly" limitation of claim 1, Patent Owner argues that similar to Splash2, "Jeong is ambiguous" regarding where results and inputs are stored. PO Resp. 109. Patent Owner further disputes the combination of references because Splash2 discloses a "linear" system and Jeong discloses a "non-linear" system. *Id.* at 118–119. Petitioner's position, though, is that a person of ordinary skill in the art would have had reason to implement the "modular multiplication algorithm" of Jeong using the unidirectional systolic array described in Splash2—not the entirety of Jeong's disclosed system. Pet. 70–71, 75–76; *see* Ex. 1003 ¶¶ 443, 444, 455, 458. That assertion is supported sufficiently by the testimony of Dr. Stone, which is not conclusory or based on hindsight bias as Patent Owner contends. *See* Ex. 1003 ¶¶ 456–461; *supra* Section II.G.

Based on all of the evidence of record, including evidence of secondary considerations of nonobviousness submitted by Patent Owner,

see supra Section II.F.2, we determine that claim 20 would have been obvious based on Splash2 and Jeong under 35 U.S.C. § 103(a).

I. Obviousness Ground Based on Splash2 and Chunky SLD (Claims 7, 17, and 24)

Petitioner asserts that dependent claims 7, 17, and 24 are unpatentable over Splash2 and Chunky SLD. -1602 Pet. 63–79; *see* Ex. 1003 ¶¶ 373–417, 523–531. Claims 7, 17, and 24 limit the "calculation" in claim 1 to a particular type of calculation. For example, claim 17 recites that the calculation comprises "a search algorithm for an image search." Petitioner relies on Chunky SLD for the limitation, arguing that the reference teaches an "automatic target resolution algorithm . . . used in the search for partially obscured images in synthetic aperture radar data" and "implemented on a linear systolic array instantiated in a Splash 2 system." -1602 Pet. 64 (citing Ex. 1011, 192). Petitioner provides numerous reasons why a person of ordinary skill in the art would have been motivated to combine the teachings of Chunky SLD with Splash2's teachings regarding the unidirectional array implementation. *Id.* at 74–76. For example, Petitioner notes that Chunky SLD "expressly cites the Splash 2 system as the platform to which its computing algorithms are mapped." *Id.* at 75 (citing Ex. 1011, 192).

Patent Owner does not dispute that Chunky SLD teaches the limitations of dependent claims 7, 17, and 24 or dispute Petitioner's asserted reasons to combine Splash2 and Chunky SLD (other than the general arguments regarding motivation addressed above). *See supra* Section II.G. Instead, Patent Owner argues that Chunky SLD does not teach two "computational loops," as recited in parent claim 1, because the reference

does not teach repeating a computation multiple times "for each piece of data." PO Resp. 91. Patent Owner's argument is premised on its proposed interpretation requiring repeated execution "per datum," which we do not adopt. *See supra* Section II.C.2. Also, as explained above, we find that Splash2 teaches two "computational loops." *See supra* Section II.E.2.

Likewise, with respect to the "seamlessly" limitation of claim 1, Patent Owner argues that "Chunky SLD is simply an algorithm deployed on Splash2, and therefore the same ambiguity present in Splash2 [regarding where data is stored] is also present." PO Resp. 108–109. We disagree that Splash2 is ambiguous. *See supra* Section II.E.2.a.3. Further, Petitioner's arguments regarding the combined teachings of Splash2 and Chunky SLD are supported sufficiently by the testimony of Dr. Stone, which is not conclusory or based on hindsight bias as Patent Owner contends. *See* -1602 Pet. 69–70, 73–76; Ex. 1003 ¶¶ 409–417; *supra* Section II.G.

Based on all of the evidence of record, including evidence of secondary considerations of nonobviousness submitted by Patent Owner, *see supra* Section II.F.2, we determine that claims 7, 17, and 24 would have been obvious based on Splash2 and Chunky SLD under 35 U.S.C. § 103(a).

J. Obviousness Ground Based on Splash2 and Roccatano (Claims 2–5, 22, and 23)

Petitioner asserts that dependent claims 2–5, 22, and 23 are unpatentable over Splash2 and Roccatano. -1603 Pet. 67–86; *see* Ex. 1003 ¶¶ 280–329, 466–522. Claims 22 and 23 limit the "calculation" in claim 1 to a particular type of calculation, and claims 2–5 further limit the "subsequent and previous data dimensions" of the calculation in claim 1. For example, claim 2 recites that "said subsequent and previous data dimensions of said

calculation comprise multiple vectors in said calculation." Petitioner relies on Roccatano for the limitation, arguing that Roccatano teaches "a program for simulating molecular dynamics interactions of molecules and compounds" using "a systolic loop of processors as laid out on a 3-D grid" shown in Figure 2 of Roccatano. -1603 Pet. 67–69. Petitioner provides numerous reasons why a person of ordinary skill in the art would have been motivated to combine the teachings of Roccatano with Splash2's teaching of the unidirectional array implementation, including, for example, that "Splash2 discloses that it is one of two reconfigurable systems that have achieved 'supercomputer performance' on applications that include molecular biology, which is the underlying application for Roccatano." *Id.* at 77–79 (citing Ex. 1007, 6).

Patent Owner does not dispute that Roccatano teaches the limitations of dependent claims 2–5, 22,[25] and 23 or dispute Petitioner's asserted reasons to combine Splash2 and Roccatano (other than the general arguments regarding motivation addressed above). *See supra* Section II.G. Instead, Patent Owner argues that Roccatano does not teach two "computational loops," as recited in parent claim 1, because the reference does not teach repeating a computation multiple times "for each piece of data." PO Resp. 91–92. Patent Owner's argument is premised on its proposed interpretation requiring repeated execution "per datum," which we do not adopt. *See supra* Section II.C.2. Also, as explained above, we find that Splash2 teaches two "computational loops." *See supra* Section II.E.2.

[25] We also conclude that Petitioner has proven that claim 22 is anticipated by Splash2. *See supra* Section II.E.3.

Likewise, with respect to the "seamlessly" limitation of claim 1, Patent Owner argues that Roccatano does not teach passing computed data "seamlessly" because it uses multiple processors and, even if not, it is "ambiguous" where the processors store intermediate results, similar to the alleged teachings of Splash2. PO Resp. 106–107. Petitioner's position, though, is that a person of ordinary skill in the art would have had reason to implement the calculations of Roccatano using the unidirectional systolic array described in Splash2—not using the multiple processors of Roccatano. -1603 Pet. 70–72, 76–81, 85–86; *see* Ex. 1003 ¶¶ 291, 294, 315, 493, 495–497, 501, 504, 517. Those assertions are supported sufficiently by the testimony of Dr. Stone, which is not conclusory or based on hindsight bias as Patent Owner contends. *See* Ex. 1003 ¶¶ 299–318, 514–520; *supra* Section II.G. Patent Owner's arguments attacking Roccatano individually are not persuasive.

Based on all of the evidence of record, including evidence of secondary considerations of nonobviousness submitted by Patent Owner, *see supra* Section II.F.2, we determine that claims 2–5, 22, and 23 would have been obvious based on Splash2 and Roccatano under 35 U.S.C. § 103(a).

K. Obviousness Grounds Based on Combinations with Gaudiot (Claims 1–5, 7–9, 15, 17, 18, and 20–24)

Petitioner contends that claim 1 is unpatentable over Splash2 and Gaudiot, relying on Gaudiot in the alternative for one limitation of claim 1. *See* Pet. 52–55; -1602 Pet. 60–63; -1603 Pet. 63–67. Specifically, Petitioner states that "[t]o the extent one might argue that Splash2 does not sufficiently disclose that the edit distance calculation 'is systolically implemented by

said reconfigurable computing system at the at least one reconfigurable processor,'" as recited in claim 1, it would have been obvious to modify Splash2 to do so based on the teachings of Gaudiot. Pet. 52–55 (emphasis omitted). Petitioner does not rely on Gaudiot for any of the limitations of claim 1 that are disputed by Patent Owner, namely "computational loops" and passing computed data "seamlessly" between the computational loops. *See id.*; Tr. 51:4–18. As explained above, we conclude that Splash2 discloses the systolic implementation limitation of claim 1 (which is not disputed by Patent Owner in its Response) and that Splash2 anticipates claim 1. *See supra* Section II.E.2. As such, we need not address Petitioner's alternative ground of unpatentability of claim 1 based on the combination of Splash2 and Gaudiot. Likewise, for all of Petitioner's other asserted obviousness grounds, Petitioner relies on certain references "with or without" Gaudiot. *See* Pet. 5; -1602 Pet. 6; -1603 Pet. 5. We address the grounds "without" Gaudiot above and need not address Petitioner's alternative grounds based on combinations "with" Gaudiot. *See supra* Sections II.F–II.J.

L. Constitutionality Argument

Patent Owner argues that this proceeding should be dismissed because the instant panel "was unconstitutionally appointed in violation of the Appointments Clause of the U.S. Constitution, Article II, Section 2, Clause 2," and "lacks the constitutional authority to enter a final decision." Sur-Reply 21–23. We decline to consider Patent Owner's constitutional challenge, as the issue has been addressed in *Arthrex, Inc. v. Smith & Nephew, Inc.*, 941 F.3d 1320 (Fed. Cir. 2019), *reh'g denied*, 933 F.3d 760 (Fed. Cir. 2020).

III. CONCLUSION[26]

Petitioner has demonstrated, by a preponderance of the evidence, that claims 1–5, 7–9, 15, 17, 18, and 20–24 are unpatentable. In summary:

Claim(s)	35 U.S.C. §	Reference(s)/ Basis	Claims Shown Unpatentable	Claims Not shown Unpatentable
1, 15, 18, 21, 22	102(a), 102(b)	Splash2	1, 15, 18, 21, 22	
1, 15, 18, 21, 22	103(a)	Splash2	1, 15, 18, 21, 22	
1, 15, 18, 21, 22	103(a)	Splash2, Gaudiot[27]		
8, 9	103(a)	Splash2, RaPiD	8, 9	
8, 9	103(a)	Splash2, RaPiD, Gaudiot		
20	103(a)	Splash2, Jeong	20	
20	103(a)	Splash2, Jeong, Gaudiot		
7, 17, 24	103(a)	Splash2, Chunky SLD	7, 17, 24	

[26] Should Patent Owner wish to pursue amendment of the challenged claims in a reissue or reexamination proceeding subsequent to the issuance of this Decision, we draw Patent Owner's attention to the April 2019 *Notice Regarding Options for Amendments by Patent Owner Through Reissue or Reexamination During a Pending AIA Trial Proceeding*. *See* 84 Fed. Reg. 16,654 (Apr. 22, 2019). If Patent Owner chooses to file a reissue application or a request for reexamination of the challenged patent, we remind Patent Owner of its continuing obligation to notify the Board of any such related matters in updated mandatory notices. *See* 37 C.F.R. §§ 42.8(a)(3), 42.8(b)(2).

[27] As explained above, we do not reach Petitioner's alternative grounds based on combinations with Gaudiot. *See supra* Section II.K.

7, 17, 24	103(a)	Splash2, Chunky SLD, Gaudiot		
2–5, 22, 23	103(a)	Splash2, Roccatano	2–5, 22, 23	
2–5, 22, 23	103(a)	Splash2, Roccatano, Gaudiot		
Overall Outcome			1–5, 7–9, 15, 17, 18, 20–24	

IV. ORDER

In consideration of the foregoing, it is hereby:

ORDERED that claims 1–5, 7–9, 15, 17, 18, and 20–24 of the '324 patent have been shown to be unpatentable;

FURTHER ORDERED that Petitioner's Motion to Exclude (Paper 60) is *granted-in-part*, *denied-in-part*, and *dismissed-in-part*;

FURTHER ORDERED that Patent Owner's Motion to Exclude (Paper 61) is *denied-in-part* and *dismissed-in-part*; and

FURTHER ORDERED that Exhibit 2168 is expunged from the record of this proceeding.

This is a final decision. Parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

FOR PETITIONER:

Joseph A. Micallef
Jason Greenhut
Scott Border
SIDLEY AUSTIN LLP
jmicallef@sidley.com
jgreenhut@sidley.com
sborder@sidley.com

FOR PATENT OWNER:

Alfonso Chan
Joseph DePumpo
SHORE CHAN DePUMPO LLP
achan@shorechan.com
jdepumpo@shorechan.com