Trials@uspto.gov
571-272-7822

Paper 36
Entered: September 18, 2024

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

XILINX, INC., ADVANCED MICRO DEVICES, INC., and
ATI TECHNOLOGIES ULC,
Petitioner,

v.

SENTIENT SENSORS LLC,
Patent Owner.

IPR2023-00195
Patent 6,938,177 B1

Before JON M. JURGOVAN, CHRISTA P. ZADO, and
SHEILA F. McSHANE, *Administrative Patent Judges.*

ZADO, *Administrative Patent Judge*.

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

I. INTRODUCTION

We have jurisdiction to hear this *inter partes* review under 35 U.S.C. § 6. This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a). For the reasons discussed herein, upon examination of the entirety of the record in this proceeding, we determine that Xilinx, Inc., Advanced Micro Devices, Inc., and ATI Technologies ULC (collectively, "Petitioner") has shown by a preponderance of the evidence that challenged claims 1–5, 10–15, and 17–20 ("the challenged claims") of U.S. Patent No. 6,938,177 B1 (Ex. 1001, "the '177 patent") are unpatentable.

A. Background

Petitioner filed a Petition (Paper 2, "Pet." or "Petition") for *inter partes* review of the challenged claims of the '177 patent, along with the Declaration of Dr. Stanley Shanfield (Ex. 1002). Sentient Sensors, LLC ("Patent Owner") timely filed a Preliminary Response. Paper 6 ("Prelim. Resp.").

On September 20, 2023 we instituted an *inter partes* review. Paper 12 ("Inst. Dec." or "Institution Decision").

Subsequent to institution, Patent Owner filed a Patent Owner Response (Paper 18, "PO Resp."), along with the Declaration of John Peck (Ex. 2020). Petitioner filed a Reply (Paper 21, "Pet. Reply") to the Patent Owner Response, along with a Supplemental Declaration of Dr. Stanley Shanfield (Ex. 1050). Patent Owner filed a Sur-Reply (Paper 24, "PO Sur-Reply").

An oral hearing was conducted on June 26, 2024. A transcript of the hearing is included in the record. Paper 35 ("Tr.").

B. Real Party-in-Interest

Petitioner identifies itself as the only real party in interest. Pet. 67. Patent Owner identifies itself and Management Sciences Inc. as the only real parties in interest. Paper 4, 1; Paper 27, 1.

C. Related Matters

The parties identify the following as related district court matters:

Sentient Sensors, LLC v. Xilinx, Inc., No. 22-cv-00173 (D. Del.) (filed Feb. 9, 2022) (active);

Blemel Technologies LLC v. National Instruments Corp., No. 15-cv-134 (E.D. Texas) (terminated by transfer to W.D. Texas);

Blemel Technologies LLC v. National Instruments Corp., No. 16-cv-1280 (W.D. Texas) (terminated by joint motion to dismiss with prejudice);

Sentient Sensors, LLC v. Microsemi Corp., No. 18-cv-121 (E.D. Texas) (terminated by joint motion to dismiss with prejudice);

Sentient Sensors, LLC v. Cypress Semiconductor Corp., No. 19-mn-1868 (D. Del.) (terminated by joint motion to dismiss with prejudice);
Pet. 67–68; Paper 4, 1; Paper 27, 1.

D. The '177 Patent (Ex. 1001)

The '177 patent is titled "Multi-Chip Module Smart Controller" and relates generally to a multi-chip module (MCM) "capable of running multiple [ ] processes and having non-volatile storage for numerous monitoring/controlling applications." Ex. 1001, code (54), 1:14–18. More specifically, the '177 patent relates to an Advanced Instrument Controller (AIC) (e.g., for space-based experimental test devices) that utilizes MCM design. *See generally id.* at 1:21–2:31. In particular, the '177 patent

purports to improve upon the AIC described in U.S. Patent No. 6,148,399 ("Lyke"). *Id.*

The '177 patent states that Lyke's AIC improved upon prior art by tightly coupling MCMs (which previously were not tightly coupled, according to the '177 patent). *Id.* at 1:44–2:8. "By tightly coupling the MCM more complex interactions between the components within the design are possible, thereby providing greater functional capability than would have been possible using a single integrated circuit." *Id.* at 1:61–64. In addition, tightly coupling MCM's "greatly reduced size, weight, and power consumption." *Id.* at 2:1–3.

Lyke's AIC design had drawbacks, however, according to the '177 patent. One stated drawback is the incapacity to perform "parallel independent processes":

> [T]here are several limitations to the design in [Lyke]. First, the AIC is not capable of performing independent parallel processes. Although the AIC does include a specialized [application-specific intended circuit] ASIC, it is controlled and clocked by the microprocessor. Accordingly, independent parallel processes are not possible.

Id. at 2:9–15.

Another stated drawback is that Lyke does not offer variable bit-depth when performing analog to digital ("A/D") conversion of signals:

> [T]he AIC in [Lyke] performs all A/D conversion at a 10 bit digitizing depth and does not offer variable conversion as may be required by different processes. For example, it may be preferred that some processes execute at a higher bit depth of 14 bits while some may be able to execute at a lower bit depth of 10 bits and still ensure accuracy.

Id. at 2:18–24.

The '177 patent attributes Lyke's alleged lack of independent processing to Lyke's "need for a resistive ASIC." *Id.* at 2:47–54. The '177 patent provides independent processing capability by, instead of relying on an ASIC, adding "a separately controllable FPGA [field programmable gate array] that acts as a parallel processor with internal or separate external clock" that "adds a freely re-configurable and separately programmable multi-purpose digital system which can run independent of the microprocessor." *Id.*; *see also id.* at 7:35–41 ("The design of the present invention eliminates the need for an analog application-specific intended circuit (ASIC). Instead, [the] present invention utilizes an independently operable and programmable FPGA which is used to implement many of the key instrumentation functions of an ASIC.").

In addition to describing the above-discussed type of independent processing—wherein an FPGA runs independently from the microprocessor (Ex. 1001, 2:51–54, 4:39–44, 7:41–45)—the '177 patent describes another type of "independent process" run by the FPGA—namely parallel processes within the FPGA that are independent of each other. *Id.* at 8:16–32, Fig. 2. Illustrated in Figure 2, and described in the patent, three independent processes A1, A2, and A3 may run in parallel within the FPGA. *Id.*

The FPGA in the '177 patent is not limited to independent operation, and may alternatively operate under control of the AIC's microprocessor. *Id.* at 4:39–44. Also, the FPGA may be clocked using: (1) the same voltage controlled oscillator (i.e., clock) as the microprocessor, (2) a separate, external clock, or (3) no clock (wherein the FPGA operates at gate speeds). *Id.* at 4:49–52, 7:64–8:1; *see also id.* at 6:65–7:34 (describing internal clocking).

The '177 patent provides a non-exhaustive, exemplary list of FPGA functions, which includes performing signal processing, acting as a cross-bar switching device to select analog or digital signals, performing signal conditioning and filtering, and monitoring functionality of the microprocessor, A/D and D/A conversions. *Id.* at 7:47–50, 7:59–63.

E. Challenged Claims

Petitioner challenges claims 1–5, 10–15, and 17–20 of the '177 patent. Pet. 1. Claims 1, 10, , 13, 15, and 20 are independent. Claim 1 is reproduced below with annotations in brackets as set forth in the Petition.

1. [1.pre] An instrument controller comprising:

[1.1] a non-volatile memory storage component for program and data storage;

[1.2] a large volatile memory storage component for additional program and data storage;

[1.3] a processor coupled to both the non-volatile memory storage component and the large volatile memory storage components, the processor capable of high-frequency and low-frequency operations and having an embedded memory for storing an initialization program that enables the processor to start up processing without first retrieving a program from the non-volatile memory; and

[1.4] at least two internal oscillators coupled to the processor, for providing clock signals for the low-frequency and high-frequency operations;

[1.5] a plurality of gates arranged in a field programmable gate array, the gate array coupled to the processor and configured to run independent processes in parallel with the processor;

[1.6] a plurality of analog-to-digital converters for receiving a plurality of analog inputs, digitizing the analog inputs at one of at least two possible bit depths, thereby generating digital inputs, and providing, the digital inputs to the processor;

> [1.7] wherein a first portion of the gates in the field programmable gate array is configured to perform signal processing; and
>
> [1.8] wherein a second portion of the gates in the field programmable gate array is configured to operate as a signal distribution matrix for rerouting signals within the instrument controller.

Ex. 1001, 9:55–10:20.

F. Asserted Prior Art References and Declarations

Reference or Declaration	Date	Exhibit No.
U.S. Patent No. 6,148,399 ("Lyke")	Nov. 14, 2000	Ex. 1003
Dehkordi, Peyman, et al., "Development of a DSP/MCM Subsystem Assessing Low-volume, Low-cost MCM Prototyping for Universities," *Proceedings: 1996 IEEE Multi-Chip Module Conference*[1] ("Dehkordi")	No later than Aug. 30, 2001[2]	Ex. 1004
Frantz, Gene A., et al., "The Texas Instruments TMS320C25 Digital Signal Microcomputer," *IEEE Micro*[3] ("Frantz")	No later than Dec. 1986[4]	Ex. 1005
Faura, Julio, "A Novel Mixed Signal Programmable Device with On-Chip Microprocessor," *Proceedings of 1997 IEEE Custom Integrated Circuits Conference*[5] ("Faura")	No later than Feb. 10, 1998[6]	Ex. 1006
U.S. Patent No. 5,663,734 ("Krasner")	Sept. 2, 1997	Ex. 1007
U.S. Patent No. 4,279,020 ("Christian")	July 14, 1981	Ex. 1008
U.S. Patent No. 5,675,824 ("Steele")	Oct. 7, 1997	Ex. 1009
Declaration of Dr. Stan Shanfield ("Shanfield Dec.")	Feb. 1, 2023	Ex. 1002
Declaration of June Ann Munford ("Munford Dec.")	Feb. 2, 2023	Ex. 1011

[1] *See* the Declaration of June Ann Munford (Ex. 1011) ¶¶ 6–12, App. DEHKORDI01–06.
[2] *See* Ex. 1011 ¶¶ 13–14, App. DEHKORDI02–06 (field 008).
[3] *See* Ex. 1011 ¶¶ 24–30, App. FRANTZ01–06.
[4] *See* Ex. 1011 ¶¶ 31–32, App. FRANTZ02–06 (field 008).
[5] *See* Ex. 1011 ¶¶ 15–20, App. FAURA01–05.
[6] *See* Ex. 1011 ¶¶ 21–23, App. FAURA01–05 (field 008).

G. Asserted Grounds of Unpatentability[7]

Claim(s) Challenged	35 U.S.C. §	Reference(s)/Basis
1, 4	103(a)	Lyke, Dehkordi, Frantz, Faura
2	103(a)	Lyke, Dehkordi, Frantz, Faura, Krasner
3	103(a)	Lyke, Dehkordi, Frantz, Faura, Christian
5, 10, 11, 13, 14, 15, 17, 19, 20	103(a)	Lyke, Dehkordi, Frantz, Faura, Steele
12, 18	103(a)	Lyke, Dehkordi, Frantz, Faura, Steele, Christian

II. ANALYSIS

A. Level of Ordinary Skill

In determining whether an invention would have been obvious at the time it was made, we consider the level of ordinary skill in the pertinent art at the time of the invention. *Graham v. John Deere Co.*, 383 U.S. 1, 17 (1966). "The importance of resolving the level of ordinary skill in the art lies in the necessity of maintaining objectivity in the obviousness inquiry." *Ryko Mfg. Co. v. Nu-Star, Inc.*, 950 F.2d 714, 718 (Fed. Cir. 1991).

[7] *See* the Leahy-Smith America Invents Act ("AIA"), Pub. L. No. 112-29, 125 Stat. 284, 287–88 (2011), amended 35 U.S.C. §§ 102, 103, effective March 16, 2013. Because the application for the '177 patent was filed before this date, the pre-AIA version of § 103 applies. *See* Ex. 1001, code (22).

In addition, as discussed below, *infra* Sec. II.B, in interpreting claim language we do so from the perspective of one with ordinary skill in the art. *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc).

In determining the level of skill in the art, factors that may be considered include the educational level of the inventor, type of problems encountered in the art, the prior art solutions to those problems, the rapidity with which innovations are made, the sophistication of the technology, and the educational level of active workers in the field. *Daiichi Sankyo Co. v. Apotex, Inc.*, 501 F.3d 1254, 1256 (Fed. Cir. 2007).

Petitioner contends

> The POSITA [person of ordinary skill in the art] of the '177 patent would have been a person with a Bachelor's degree in Electrical Engineering or Computer Engineering, or a related field, and at least two years of experience related to the design or development of multi-chip modules operating as instrument controllers.

Pet. 7 (citing Ex. 1002 ¶¶ 24–26).

Patent Owner contends

> An ordinary artisan would have been a person having, as of December 2001, a Bachelor's degree in Electrical or Computer Engineering or an equivalent degree with at least four years of experience in FPGA and processor systems design including but not limited to system on a chip "SoC" and electronic vehicle sensor systems or related technologies. Additional education may substitute for lesser work experience and vice-versa.

PO Resp. 24–25 (citing Ex. 2020 ¶ 22).

Despite the different levels set forth in the parties' papers, during the hearing the parties agreed that regardless of whose proposed level we adopt, it would not change the outcome in this proceeding:

> JUDGE ZADO: Counselor, I'm going [to] ask you to pause right there because this does touch upon an area where I did have a few questions, now that we're talking about the person of ordinary skill in the art.
>
> Petitioner has proposed in the petition one level, and Patent Owner proposes a different level. And my question is, does which level we adopt change how we should decide this case?
>
> MR. SMITH: We believe no, Your Honor.

Tr. 6:11–17; *see also id.* at 6:18–7:11 (Petitioner explaining why the adopted level of ordinary skill in the art does not impact the analysis by the parties).

> JUDGE ZADO: So, Counselor, before we run out of time, I wanted to briefly ask you a question that I also put to Petitioner, and this is about the level of ordinary skill in the art . . . [a]nd so, my question is, can you identify any argument in the record made by Patent Owner because I did not see it, any argument in the record by Patent Owner that either Petitioner or Dr. Shanfield's positions are incorrect as a result of applying a different level or ordinary skill in the art?
>
> MR. COCHRAN: Thank you, Your Honor. I think both definitions of POSA are very similar, and the arguments would be the same under either definition.

Tr. 35:22–36:12.

Neither party identifies any testimony or argument that it alleges is in error as result of the other party's proposed level or ordinary skill in the art, nor do we discern a difference in outcome that would result were we to adopt one level versus the other. Indeed, Petitioner's expert declarant states

that his opinions would not change if we were to adopt Patent Owner's proposed level of ordinary skill in the art. Ex. 1050 ¶ 16.

Based on the record presented, including our review of the '177 patent and the types of problems and solutions described in the patent and the cited prior art, we adopt Petitioner's assessment of the level of ordinary skill in the art, but without the qualifier "at least" before the words "two years," which we also did in the Institution Decision. Inst. Dec. 9–10. Patent Owner does not provide any explanation to justify requiring "at least four years of experience in FPGA and processor systems design including but not limited to system on a chip 'SoC' and electronic vehicle sensor systems or related technologies." PO Resp. 24–25; Pet. Reply 3. As Petitioner argues, "[t]he '177 patent is not about 'FPGA and processor systems design'" or "system on a chip 'SoC' and electronic vehicle sensor systems." Pet. Reply 2. Rather, the '177 patent is directed to multi-chip module ("MCM") design, and even distinguishes MCM design from SoC. *Id.* at 2–3 (citing Ex. 1050 ¶ 18 (quoting Ex. 1001, 3:55–60)).

However, as the parties acknowledge, whichever level we adopt does not alter the outcome of this Decision.

B. Claim Construction

Pursuant to 37 C.F.R. § 42.100(b), we apply the claim construction standard as set forth in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). Under *Phillips*, claim terms generally are given their ordinary and customary meaning as would be understood by one with ordinary skill in the art in the context of the specification, the prosecution history, other claims, and even extrinsic evidence including expert and inventor testimony, dictionaries, and learned treatises, although extrinsic

evidence is less significant than the intrinsic record. *Phillips*, 415 F.3d at 1312–17.

Only terms that are in controversy need to be construed, and then "only to the extent necessary to resolve the controversy." *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co. Matal*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (in the context of an *inter partes* review, applying *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999)).

Limitation [1.5] of claim 1 recites, "a plurality of gates arranged in a field programmable gate array, the gate array coupled to the processor and configured to run *independent processes* in parallel with the processor." Ex. 1001, 10:4–7 (emphasis added).

Claim 5 recites the instrument controller of claim 1, "wherein a third portion of the gates in the field programmable gate array is configured to operate as an internal embedded *power converter* capable of receiving an input voltage level and generating each operating and reference voltage needed within the instrument controller." Ex. 1001, 10:38–43 (emphasis added).

The parties propose interpretations of the terms "process," "independent process[es]," and "power converter." PO Resp. 28–29; Pet. Reply 4–12, 21; PO Sur-Reply 3–8, 16.

1. "process"

Neither lexicography nor claim disavowal dictates a departure from applying the ordinary and customary meaning of "process" here. *GE Lighting Solutions, LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1309 (Fed. Cir. 2014) (explaining that departure from the ordinary and customary meaning is appropriate only in two circumstances: lexicography and disavowal).

Therefore, the term "process" is accorded its ordinary and customary meaning. *See Phillips*, 415 F.3d at 1312–17.

"process"	
Party/Board	**Construction**
Patent Owner	"a computational process programmed into the FPGA gates" (PO Resp. 28)
Petitioner	"a segment of code that may be run independently" (Pet. Reply 5) OR "[a] program or part of a program; a coherent sequence of steps undertaken by a program" (Ex. 1050 ¶ 55)
Construction adopted herein	"a program or part of a program; a coherent sequence of steps undertaken by a program"

Patent Owner construes "process" as "a computational process programmed into the FPGA gates." PO Resp. 28. Patent Owner argues:

> Robust specification support for this definition exists in multiple areas including, *e.g.*, at EX. 1001 at 8:16–32; 9:19–35. A primary purpose of the FPGA is to act as a co-processor for various analog or digital functions that compute a result. EX. 1001 4:39–50. Power up sequences do not qualify at least because they do not produce a computational result and such features have no specification support as processes. Similarly, initialization sequences also do not qualify because they also do not produce a computational result, but rather merely prepare the FPGA to perform computations and also have no specification support. EX 2020 ¶¶ 49–51.

PO Resp. 28.

Petitioner agrees that the '177 patent does not provide a lexicographic definition of the term "process," and the ordinary and customary meaning

applies. Pet. Reply 5. Petitioner argues that the plain and ordinary meaning of "process" includes "the meanings noted by the Board [in the Institution Decision], including that a process includes 'a segment of code that may be run independently.'" *Id.* Petitioner argues this understanding is consistent with the '177 patent specification, which discusses "processing" on an FPGA in terms of being configured to run tasks. *Id.* (citing Ex. 1001, 4:45–49; Ex. 1050 ¶¶ 47–55).

Petitioner's expert, Dr. Shanfield, testifies that in addition to Petitioner's proposed construction, a plain an ordinary meaning of "process" includes "[a] program or part of a program; a coherent sequence of steps undertaken by a program." Ex. 1050 ¶ 55.

For reasons discussed below, we construe "process" as "a program or part of a program; a coherent sequence of steps undertaken by a program."

Patent Owner's construction does not interpret the meaning of the term "process," but rather adds the word "computational" in front of the word "process"—a word that appears neither in the claim nor in the patent specification—and adds the words "programmed into the FPGA gates" after the word "process." In so doing, Patent Owner essentially re-writes the claim language to be narrower, by adding the requirement that the process be "computational." We do not discern, and Patent Owner has not provided, a basis for narrowing the term "process" in this manner. Patent Owner argues the word "computational" is not narrowing, but rather clarifies the type of process. PO Sur-Reply 4. However, Patent Owner does not explain why the clarification is needed, much less why it is accurate or justified. Rather, Patent Owner provides a conclusory statement that it has "provided ample support in the intrinsic evidence for its proposed claim construction." PO

Sur-Reply 5 (citing PO Resp. 28 (citing Ex. 1001, 2:9–18, 2:48–55, 4:39–50, 8:16–32, 9:19–35); Ex. 2020 ¶¶ 50–52).

It is insufficient to simply cite to evidence without explaining how the evidence supports an argument, especially here where the cited evidence does not use the term "computational" or appear to suggest that the "process" is "computational." Furthermore, Patent Owner never explains what the term "computational" means. Patent Owner's construction, therefore, itself requires interpretation.

Petitioner's proposed construction is based on a passage in *Computer Architecture: A Quantitative Approach* (Ex. 3004) that we quoted in the Institution Decision. The quote reads "a process is a segment of code that may be run independently." Ex. 3004, 530 (citied by Inst. Dec. 15). We provided this quote as an example of use of the term "process." Inst. Dec. 15. However, we also provided additional examples:

> By way of example, the *Microsoft Computing Dictionary*[11] defines "process" as "[a] program or part of a program; a coherent sequence of steps undertaken by a program." Ex. 3002, 423. *Computer Organization and Architecture Designing for Performance*[12] states that the concept of "process" first was used by designers of Multiplexed Information and Computing Services (Multics), and further states that once a program is admitted to a system for processing, it becomes a process. Ex. 3003, 234–235. *Computer Architecture: A Quantitative Approach*[13] similarly states that multiprogramming led to the concept of a process: "[t]he invention of multiprogramming, where a computer would be shared by several programs running concurrently, led to new demands for protection and sharing among programs . . . Multiprogramming leads to the concept of a *process*."

Id. at 14–15.

We do not necessarily find Petitioner's proposed definition to be an erroneous statement concerning the term "process." However, we find that the phrase "a segment of code" requires further clarification in the context of an FPGA. Petitioner's expert, Dr. Shanfield, explains that

> While a POSITA would typically speak of the code or program on the FPGA as the FPGA's "configuration" or "configuration bitstream" (see e.g., EX1035 ("configured via a string of bits")) the term "process" was also sometimes used and would have been understood as referring to the individual pieces of "program" or "code" (i.e., the relevant portion of the configuration bitstring), which in an FPGA are the partitions of gates configured to perform a task. As one example of a FPGA reference referring to the code or program configuration running on the FPGA as a process, *see, e.g.*, EX1066, 34:25-26 (referring to "a plurality of [FPGA gate] partitions, i.e. processes"); 34:55-65 (talking about the various tasks that are running on the FPGA as "processes"); *id.* at 37:60-63 (same)).

Ex. 1050 ¶ 54. We find credible Dr. Shanfield's testimony that "process" may refer to configurations of gates or partitions within the FPGA because the record supports it. For example, the '177 patent specification describes processes as "independent configurations of a subset of gates within the FPGA." Ex. 1001, 8:18–26.

Between Petitioner's construction (i.e., a segment of code that may be run independently) and Dr. Shanfield's additional interpretation (i.e., a program or part of a program; a coherent sequence of steps undertaken by a program), we adopt the latter to avoid the need to clarify that in the context of FPGA's a segment of code refers to partitions of gates that perform tasks. However, had we adopted Petitioner's proposed construction, with the clarification of the preceding sentence, our Decision would be the same.

In adopting our construction, we find that in addition to being consistent with the ordinary and customary meaning reflected in the *Microsoft Computing Dictionary*, it is consistent with the use of the term in the context of the '177 patent specification. Dr. Shanfield provides examples of use of the term "process" in the '177 patent specification in the context of the FPGA being configured to perform various tasks. Ex. 1050 ¶¶ 52–53 (citing Ex. 1001, 4:45–49, 8:18–26, Fig. 2). We note that although Patent Owner argues that our interpretation of the term "process" should include the clarifying word "computational," Patent Owner does not argue that the two interpretations acknowledged by Dr. Shanfield (*see* Ex. 1050 ¶ 55) are not ordinary and customary meanings in the context of the '177 patent.

For the above reasons, we construe "process" as "a program or part of a program; a coherent sequence of steps undertaken by a program."

2. "independent process[es]"

Neither lexicography nor claim disavowal dictates a departure from applying the ordinary and customary meaning of "independent processes" here. *GE Lighting Solutions, LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1309 (Fed. Cir. 2014) (explaining that departure from the ordinary and customary meaning is appropriate only in two circumstances: lexicography and disavowal). Therefore, we accord the phrase "independent processes" its ordinary and customary meaning. *See Phillips*, 415 F.3d at 1312–17.

The construction of the entire phrase "a plurality of gates arranged in a field programmable gate array, the gate array coupled to the processor and configured to run independent processes in parallel with the processor" raises potentially two questions: 1) what it means for processes to be

"independent," and 2) whether the processes run on the FPGA must be independent of each other, or whether they must be independent of processes run on the processor.

a) Meaning of "independent"

"independent process"/ "independent process"	
Party/Board	**Construction**
Patent Owner	"without computational assistance from another" (PO Resp. 28)
Petitioner	"a process that runs separately from another" (Pet. Reply 4, 6)
Construction adopted herein	"processes that run separately from one another"

We note that in the Institution Decision we encouraged the parties to propose a construction for the term "independent process" because the parties appeared to dispute the meaning of the term. Inst. Dec. 13. We stated that Petitioner's arguments suggest the term "indicates a process that runs separately from those that run on the processor." *Id.* (citing Pet. 28–29). We stated that Patent Owner's arguments "suggest the term 'independent process' requires the components running the processes (*i.e.*, processor and FPGA) to run on separate clocks" and that "when acting as a slave to the processor, FPGA processes are not independent of the processor." *Id.* (citing Prelim. Resp. 23, 24, 27) (footnote omitted). As discussed below, Patent Owner's construction, proposed in its Response, does not import the requirements of a separate clock and/or control bus. Patent Owner acknowledges in the Sur-Reply that its construction does not

require separate clocks, stating that its "proposed construction is agnostic to the type of clock." PO Sur-Reply 12 (citing Ex. 1001, 4:48–55). Despite omitting these requirements (i.e., separate clocks and/or control bus) from its construction, the Patent Owner Response argues the prior art does not teach or suggest independent processes because the FPGA and processor share a clock and share bus control lines. *See, e.g.*, PO Resp. 33, 38–39. However, Patent Owner backs down from this argument in the Sur-Reply. PO Sur-Reply 12.

Patent Owner construes "independent processes" as "without computational assistance from another." PO Resp. 28. Patent Owner asserts that its construction is consistent with its previous litigation position. *Id.* Such assertion is not evidence or argument concerning the correct construction of a claim term, and we accord it little weight. Patent Owner also asserts that its construction is consistent with the '177 patent specification (*id.* citing Ex. 1001, 2:9–18, 2:48–55), but does not provide any explanation as to how or why. *Id.* The cited portions of the specification state that prior art AIC designs are not capable of performing parallel independent processes, noting that the prior art's specialized ASIC is controlled and clocked by the microprocessor and cannot run independently of the microprocessor and the invention's FPGA is separately controllable and acts as a parallel processor with internal or separate clock. Ex. 1001, 2:9–18, 2:38–55. There is no discussion of computational assistance, much less whether parallel independent processes do or do not require or allow for computational assistance from another. *Id.*; Pet. Reply 8. We agree with Petitioner that the word "computational" does not appear anywhere in the

'177 patent, and we discern nothing in the intrinsic record that supports Patent Owner's construction. Pet. Reply 8.

Given the lack of evidence in the intrinsic record for Patent Owner's construction, we do not adopt it.

Petitioner construes "independent process" as "a process that runs separately from another," and explains that "independent processes" is the plural form of "independent process." Pet. Reply 4, 6.

Petitioner argues that its construction is consistent with the '177 patent specification and prosecution history. Pet. Reply 6–8. We agree.

Figure 2 of the '177 patent illustrates FPGA gates configured as "analog processes." Ex. 1001, Fig. 2 (cited by Pet. Reply 7). The '177 patent describes these processes as "independent configurations of a subset of gates within the FPGA." *Id.* at 8:18–26 (cited by Pet. Reply 7). Petitioner argues that this disclosure means the processes run separately. Pet. Reply 7. The context surrounding the cited passage supports Petitioner's understanding. The '177 patent describes three independent processes A1–A3 as independent configurations of a subset of gates within the FPGAs, and that outputs from these processes are output to a node. Ex. 1001, 8:16–26. The '177 patent suggests the processes run separately because when functions A1–A3 "differ," then "different analog processing functions are selectively applied" to the signal being monitored. *Id.* at 8:29–32. Furthermore, when functions A1–A3 are identical, redundancy is achieved—the fact that each process is redundant of the other suggests each function A1 through A3 is run separately. *Id.* at 8:26–29.

As to the file history, during prosecution the Examiner treated "independent processes" as processes that run separately. As argued by

Petitioner, the Examiner and patent applicant treated an FPGA as a component that can be programmed in a variety of ways and functions as a parallel processor, each running a set of instructions. Pet. Reply 7–8 (citing Ex. 1010, 58 (citing Ex. 1040, 2:65–67); Ex. 1050 ¶¶ 37–43 (citing Ex. 1063, 354)). In determining that Cloutier teaches or suggests independent processes, the Examiner found that Cloutier teaches a multiprocessor, wherein the FPGA functions as a multiprocessor and functions as a matrix of processing elements each running a set of instructions. *Id.* (citing Ex. 1040, 2:65–67; Ex. 1050 ¶¶ 37–43). The description of each processing element running a set of instructions suggests running instructions separately is what is meant by independent processes.

The intrinsic record is consistent with extrinsic evidence. *Computer Architecture: A Quantitative Approach*[8] discusses running of independent processes in parallel in the context of multiprocessing and running separate sets of instructions. *Computer Architecture* states that multiprogramming led to the concept of a process: "[t]he invention of multiprogramming, where a computer would be shared by several programs running concurrently, led to new demands for protection and sharing among programs . . . Multiprogramming leads to the concept of a *process*." Ex. 3004, 469. *Computer Architecture* states further that

> a process is a segment of code that may be run independently, and that the state of the process contains all the information necessary to execute that program on a processor. In a multiprogrammed environment, where the processors may be

[8] WILLIAM STALLINGS, COMPUTER ORGANIZATION AND ARCHITECTURE DESIGNING FOR PERFORMANCE, (Prentice Hall 4th ed. 1996) (Ex. 3003, "*Computer Architecture*").

> running independent tasks, each process is typically independent of the processes on other processors.

Id. at 530. We included these excerpts from *Computer Architecture* in our Institution Decision and invited the parties to consider this evidence. Inst. Dec. 15. However, Patent Owner does not address this evidence, much less refute that it is consistent with the ordinary and customary meaning of "independent processes."

Having reviewed the evidence and argument of record, we determine that "independent processes" are "processes that run separately from one another."

b) What the "processes" must be "independent" of

The claim language states that "independent processes" run in parallel with the processor, but is silent as to what the processes are "independent" of. The claim's silence leaves open at least two possibilities: (1) the processes are independent of the processor, and (2) the processes are independent of each other. Both interpretations find support in the '177 patent.

As to (1), the '177 patent discloses, for example, "[t]he FPGA . . . adds a freely re-configurable and separately programmable multi-purpose digital system which can run independent of the microprocessor." Ex. 1001, 2:51–54. The '177 patent also discloses, e.g., "[t]he microprocessor is further coupled to a [FPGA] . . . [i]n a preferred embodiment, the FPGA . . . may operate either under control of the microprocessor (for example as a coprocessor) or, may be operated independently." *Id.* at 4:39–44.

As to (2), in some embodiments of the '177 patent the FPGA is not independent of the processor, but rather operates under its control. *Id.* at

4:40–44 (disclosing that processes run on the FPGA "may operate either under the control of the microprocessor (for example as a coprocessor) or, may be operated independently.").

Moreover, the '177 patent uses the term "independent" to refer to processes within the FPGA that are independent of each other. With regard to Figure 2, the '177 patent illustrates the FPGA running three processes A1, A2, and A3 independently of each other within the FPGA:

> The present invention can use the parallel processing of the FPGA to re-route a subset of its external interconnections …. FIG. 2 illustrates the concept . . . As shown, gates X1, X2 and X3 [within the FPGA] are configured to route the input at node A to anyone of three independent processes A1–A3. The processes are independent configurations of a subset of gates within the FPGA.

Ex. 1001, 8:16–32. That is, the "independent processes" the FPGA is "configured to run" are processes that are independent of each other.

Patent Owner asserts that:

> The plain meaning of "independent processes" are independent of each other and the processor. This is because grammatically speaking "independent" modifies "process" which is its plain meaning, and, as the Board points out—the specification supports both. Institution Decision at 16. EX 2020 ¶52.

PO Resp. 28.

Petitioner does not address whether the processes run on the FPGA must be independent of each other or independent of processes run on the processor. *See, e.g.*, Pet. Reply 8–12.

We determine that both (1) and (2) are grammatically consistent with the claim language and supported by the specification and figures of the '177 patent. The claim language does not use a conjunctive "and," and

nothing in the claim language indicates both conditions must be satisfied. Only one or the other need be satisfied to meet the claim. Accordingly, in order to show unpatentability, Petitioner may show either FPGA processes that are independent of each other or FPGA processes that are independent of the processor. Ultimately, Petitioner's arguments regarding "independent processes" applies to both interpretations, i.e., the relied-upon FPGA processes both run separately from each other and separately from those that run on the processor. *See, e.g.*, Pet. Reply 16 (arguing that the FPGA processes upon which Petitioner relies in the prior art are both processes that are independent of each other and independent of the processor).

3. "power converter"

Neither lexicography nor claim disavowal dictates a departure from applying the ordinary and customary meaning of "power converter" here. *GE Lighting Solutions, LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1309 (Fed. Cir. 2014) (explaining that departure from the ordinary and customary meaning is appropriate only in two circumstances: lexicography and disavowal). Therefore, the term "power converter" is accorded its ordinary and customary meaning. *See Phillips*, 415 F.3d at 1312–17.

"power converter"	
Party/Board	**Construction**
Patent Owner	"a converter which changes voltage level to another voltage level" (PO Resp. 29)
Petitioner	"a power converter converts power" (Pet. Reply 21)
Construction adopted herein	No express construction required

Patent Owner construes the term "power converter" as "a converter which changes voltage level to another voltage level." PO Resp. 29. Patent Owner's only support for this construction is that it "is consistent with Patent Owner's litigation definition (EX. 1012, 2023) and the specification. See, for example, EX. 1001 8:64–67. EX. 2020 ¶ 53)." Stating a construction is consistent with a litigation position is insufficient because it is not evidence and argument showing why a construction is correct. As to Patent Owner's assertion that its construction is consistent with the specification, Patent Owner's assertion is conclusory, without any explanation. The portion of the '177 patent that Patent Owner cites states "a first stage in the internal power converter converts an unregulated supply voltage to a fixed, regulated voltage (VREG)." Ex. 1001, 8:64–67. It is not evident on its face how this supports Patent Owner's construction. Although this describes an exemplary power converter, this does not appear to be intended to define the term "power converter." Moreover, it does not even describe changing a voltage *level* to another voltage *level*. Rather, it describes changing an *unregulated* supply voltage to a *regulated* (i.e., fixed or stable) voltage.

Patent Owner's citation to Mr. Peck's declaration (i.e., Ex. 2020 ¶ 53) does not cure the failure of the Patent Owner Response to provide an explanation to support Patent Owner's assertion, because it is an improper attempt to incorporate argument by reference. 37 C.F.R. § 42.6(a)(3) ("Arguments must not be incorporated by reference from one document into another document."). Moreover, Mr. Peck's testimony adds nothing, but merely repeats Patent Owner's Response, stating only that Patent Owner's construction "is consistent with Patent Owner's litigation definition (EX. 1012, 2023) and the specification. *See*, for example, EX. 1001 8:64–67)." Ex. 2020 ¶ 53.

Claim 5 already includes language that sufficiently describes the claimed "power converter." Specifically, claim 5 recites that the "power converter" is "capable of receiving an input voltage level and generating each operating and reference voltage needed." This claim language, in fact, is more consistent with the '177 patent specification (describing changing an unregulated supply voltage (e.g., claimed "input voltage") to a regulated voltage (e.g., claimed "operating and reference voltage needed")) than with Patent Owner's proposed construction.

Patent Owner has provided no basis to re-write the claim language by inserting an additional requirement regarding changing voltage level to another when neither the specification nor claim language requires, or even supports, it.

Petitioner argues the term "power converter" should be accorded its plain and ordinary meaning, "specifically, a power converter converts power." Pet. Reply 21. Petitioner states that this construction was adopted previously by a district court judge. *Id.* (citing Ex. 1012, 6–7; Ex. 1050

¶¶ 137–142). Petitioner argues further that the district court judge rejected the argument Patent Owner now proposes. *Id.* (citing Ex. 1012, 7).

Stating that a previous tribunal adopted a construction is not evidence and argument showing why the construction is correct. Also, it is improper to incorporate arguments by reference, such as by citing to Dr. Shanfield's declaration in lieu of providing an explanation in Petitioner's Reply. 37 C.F.R. § 42.6(a)(3).

For the reasons discussed above, we find that further interpretation of "power converter" is not necessary in view of other language in claim 5. Although we find no error in Petitioner's statement that a "power converter" "converts power," such construction merely switches the ordering of the two words and does not add anything that is not already present in the claim or assist in resolving any dispute in this proceeding. Accordingly, the term "power converter" is accorded its ordinary and customary meaning.

C. Patentability Challenges

1. Principles of Law: Obviousness

A claim is unpatentable as obvious under 35 U.S.C. § 103 if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious to a person having ordinary skill in the art to which said subject matter pertains. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations, including: (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of skill in the art; and (4) objective evidence of nonobviousness, i.e., secondary

considerations.[9] *See Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17–18 (1966).

2. Relevant Prior Art

a) Lyke (Ex. 1003)

Lyke is U.S. Patent No. 6,148,399, issued November 14, 2000, titled "Advanced Instrument Controller Statement of Government Interest." Ex. 1004, code (11), (45), (54).

Lyke "generally relates to multi-chip module (MCM) microcircuits, and more specifically to a compact (few-chip) MCM electronics control system that exploits the tight coupling of components from non-similar processes and non-volatile storage for numerous monitoring/controlling applications under harsh conditions." *Id.* at 1:14–19.

With regard to, e.g., space-based environments, Lyke seeks to improve upon previous MCM designs for providing controller and data acquisition features, which generally were "bulky, power hungry, and expensive." *Id.* at 1:21–30. Lyke achieves this end by "combin[ing] the controller and acquisition functions [of previous designs] into a single, tightly coupled MCM design." *Id.* at 1:24–26. Lyke explains that tightly coupled MCMs refer to MCMs whose components possess one or more of the following features:

> (1) more input/output (I/O) terminal contacts than is normally consistent with a discrete implementation, (2) lower capacitive drive in output circuits than is normally consistent with a discrete implementation, (3) I/O terminals in locations inconsistent with

[9] The record does not present or address any objective evidence of nonobviousness.

> standard integrated circuits (IC)s, (4) I/O circuitry with reduced or eliminated electrostatic discharge protection structures.

Id. at 1:29–38.

According to Lyke, "[b]y tightly coupling the MCM, more complex interactions between the components within are possible, introducing a design with similar physical appearance and size of a packaged integrated circuit, but with greater functional capability than possible with a single integrated circuit." *Id.* at 1:38–43.

Lyke implements its tightly coupled MCM design within an Advanced Instrument Controller (AIC). *Id.* at 1:18–66. "The functional innovations within the [AIC], being a tightly coupled MCM, are based on the ensemble core and I/O functions and the way they are used." *Id.* at 2:6–9. "The AIC combines a central processing unit, an analog application-specific integrated circuit (ASIC), a resistor ASIC, and volatile and non-volatile memory storage systems on a single tightly coupled MCM" in order to achieve "ultra-low power requirements, extremely small size and weight, versatile functionality, and the ability to operate in extreme environments." *Id.* at 3:4–11. Figure 1, reproduced below, illustrates an AIC using tightly coupled MCM design.

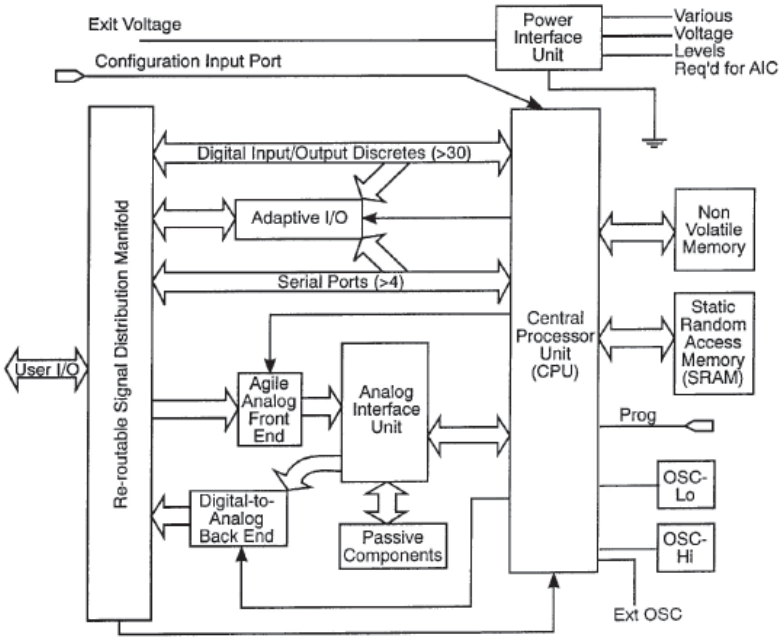


Fig. 1

Figure 1 of Lyke is a block diagram schematic of an AIC. Ex. 1003, 2:36.

Figure 1 of Lyke is a block diagram schematic of an AIC. *Id.* at 3:12–13. As shown in Figure 1 and described in Lyke, the AIC includes, *inter alia*, non-volatile memory for program and data storage (*id.* at 3:16, 4:14–15), large volatile memory (i.e., SRAM) (*id.* at 3:17, 3:66–67, 4:34–41), central processing unit (CPU) coupled to both the non-volatile and volatile memories (*id.* Fig. 1), two internal oscillators operating at two different frequencies (Hi and Lo) coupled to the CPU (*id.* at 3:19–20, 4:42–46), and multiple analog to digital converters (*see, e.g.*, 16 internal A/D channels with 12-bit resolution (*id.* at 4:64), description of analog to digital converters (*id.* at 5:18–24)).

Lyke employs a reprogrammable analog ASIC "to implement many of the key instrumentation functions," including 32 external analog inputs, 8

interpedently programmable digital to analog converter channels with 10-bit resolution, and 16 internal analog to digital converter channels with 12-bit resolution. *Id.* at 4:58–65, 5:6–7. Lyke's AIC also includes a resistive ASIC "to eliminate over 50 individual resistors." *Id.* at 5:48–51.

Lyke states that the matter set forth in the description and drawings "is offered by way of illustration only and not as a limitation," and "[o]ther variations to the current design include" "RAM-based field programmable gate array(s) interfaced internally within the AIC and configured automatically through a downloading mechanism involving AIC's internal CPU and access of the non-volatile memory." *Id.* at 8:48–59.

b) Dehkordi (Ex. 1004)

Dehkordi is a paper titled, "Development of a DSP/MCM Subsystem Assessing Low-volume, Low-cost MCM Prototyping for Universities," from *Proceedings: 1996 IEEE Multi-Chip Module Conference*.[10]

Dehkordi "discusses the design and development of a general-purpose programmable DSP subsystem packaged in a multichip module." Ex. 1004, 1.[11] Dehkordi states that IBM had utilized MCM packaging for high volume, high cost products for years, but that low cost, low volume services historically had been limited. *Id.* However, according to Dehkordi, a tremendous push from Advanced Project Agency (ARPA) for widespread usage of MCM technology led to increases in services. *Id.* Dehkordi states that, as part of an ARPA project, the authors exercised the availability of

[10] *See* Munford Dec. (Ex. 1011) ¶¶ 6–14, App. DEHKORDI01–06.
[11] For consistency with the parties, we use the pagination provided on the document by Petitioner rather than the IEEE's original pagination.

MCM technology for universities by designing and implementing an MCM subsystem. *Id.* Their implementation is a "design representing a typical application of a MCM including a processor, memory, and gate-array." *Id.* The design includes a "complete DSP [digital signal processing] subsystem based on the Motorola 96002 32-bit floating point DSP processor." *Id.* The MCM also includes SRAM, EEPROM, and a Xilinx 4010 field programmable gate array (FPGA). *Id.* According to Dehkordi, the MCM is "designed to exploit the multi-processing capability of the 96002." *Id.* The FPGA "is primarily a hardware pre-processor of incoming data but also provides other generic functions required for implementation of the entire subsystem." *Id.* Figure 1 of Dehkordi, reproduced below, is illustrative.

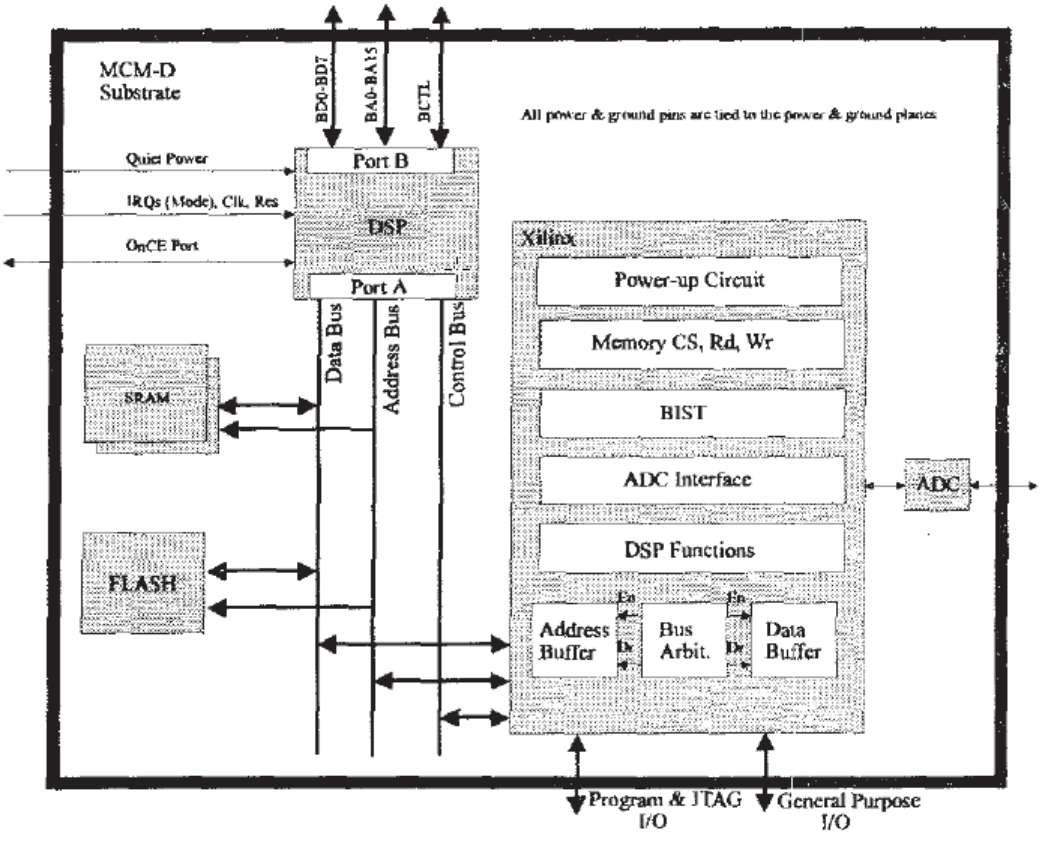


Figure 1 of Dehkordi is a block diagram of the DSP subsystem. Ex. 1004, 1.

Figure 1 of Dehkordi illustrates a MCM substrate that includes a DSP coupled to SRAM and FLASH memories, and a Xilinx FPGA. *Id.* The DSP, SRAM, FLASH, and FPGA are shown as communicating over the same bus. *Id.* In addition, the FPGA includes means to communicate externally via JTAG I/O and general purpose I/O ports. *Id.*

Dehkordi also addresses the issue of testing their MCM design. Dehkordi states that testing and fault isolation are challenging due to the small feature sizes of MCM. *Id.* at 3. Dehkordi states another challenge is that the DSP, unlike the FPGA, does not support JTAG testing. However, according to Dehkordi, TAG testing of both the DSP and FPGA is possible because "[t]he DSP supports a serial emulation port," which when "connected to the module pins along with the FPGA TAP port," supports TAG. *Id.* Dehkordi states further that:

> The FPGA has a RAM-based configuration where the configuration may come from outside the module. A portion of the FPGA is configured for multi-processing bus arbitration and is connected to the internal data and address bus on the module. The FPGA can easily be reconfigured to route these lines to the module pins for possible probing and debugging during testing. The DSP has a dual data and address port architecture in which the second port was also connected to the module pins for enhanced multi-processing and testing. Testing procedures are being constructed at the time of this publication.

Id.

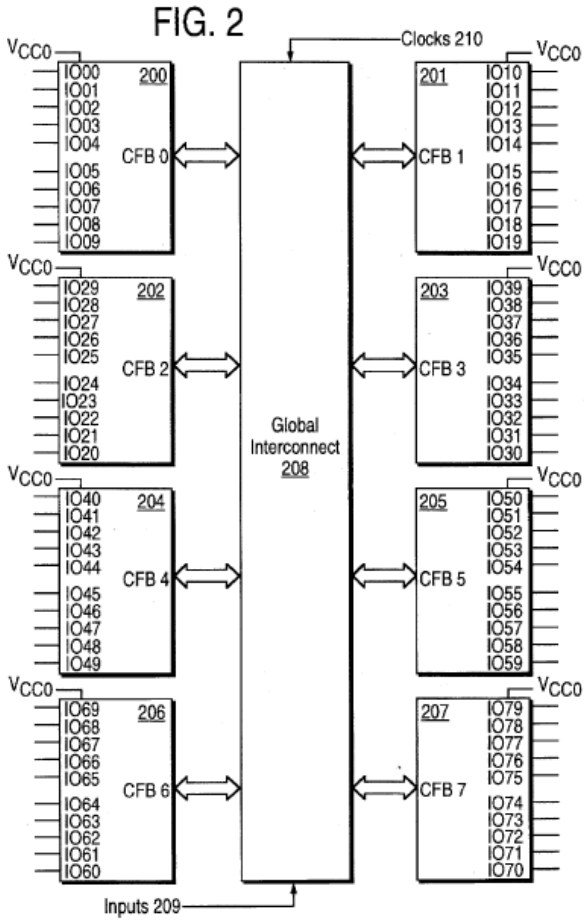
c) Steele (Ex. 1009)

Steele is U.S. Patent No. 5,675,824, issued October 7, 1997, titled "Programmable Logic Having Selectable Output Voltages." Ex. 1009, code (11), (45), (54).

Steele "pertains to the field of programmable logic devices," and more particularly, to "an apparatus and method for selecting a desired output voltage for a programmable logic device." *Id.* at 1:10–14.

Steele explains that at the time of manufacture, it is unknown whether a programmable logic device (PLD) (e.g., an FPGA) will operate at 3V or 5V. *Id.* at 1:55–59. Steele states that what is needed is a PLD that has the flexibility of processing and outputting both 3V and 5V applications. *Id.* at 2:22–23.

Figure 2 of Steele, reproduced below, illustrates such a PLD:



Ex. 1009, Fig. 2. Figure 2 of Steele is a block diagram illustrating an FPGA comprising eight configurable function blocks (CFBs) 200–207 coupled by

global interconnect 208. *Id.* at 4:6–10. The FPGA is powered by a 5 V supply voltage on the V_{CC} input pin, and can be configured to output a voltage on the CFBs' eight V_{CCO} output pins. *Id.* at 4:7–43. Specifically, the FPGA can be configured so that different CFBs output different voltages on their respective V_{CCO} pins. *Id.* at 2:35–47, 2:54–65, 4:7–43. In an exemplary embodiment, either 3 V or 5 V is output on the various V_{CCO} pins. *Id.* at 2:66–67.

d) Additional References

Petitioner's grounds of unpatentability rely on additional references. Pet. 1, 8–63. However, their descriptions are not pertinent to the issues raised by the parties, and we do not summarize them here.

D. Overview

Petitioner relies on the combination of Lyke, Dehkordi, Frantz, and Faura in challenging the patentability of claims 1–5, 10–15, and 17–20—all claims challenged in this proceeding. Pet. 1. For claims 2, 3, 5, 10–15, and 17–20, Petitioner relies on additional references. *Id.*

E. The '177 Patent and Lyke

Petitioner relies on Lyke as a primary reference. As discussed above (*supra* Sec. I.D), the '177 patent states that it improves upon the AIC disclosed in Lyke. *See, e.g.*, Ex. 1001, 1:21–2:31. Patent Owner does not dispute that Lyke discloses several of the features recited in claim 1. *See* PO Resp. 29–59. In addition, during the prosecution of the '177 patent, the Examiner relied on Lyke as a primary reference in rejecting a claim

containing all but the last two limitations of claim 1.[12] Ex. 1010, 56–57.[13] The Examiner found that Lyke teaches limitations [1.1] through [1.6], except for the embedded memory recited in limitation [1.3] and the gate array recited in limitation [1.5]. *Id.* at 56–58. For the teaching of these recitations, the Examiner relied on DaCosta[14] and Cloutier,[15] respectively. *Id.* at 57–58. As a result, the Examiner found that claim 1, without limitations [1.7] and [1.8], would have been obvious. *Id.* at 56–58.

During prosecution, the patent applicant did not challenge the Examiner's findings. *See id.* at 42–46. As such, the applicant did not dispute that claim 1, without limitations [1.7] and [1.8], is unpatentable as obvious over Lyke combined with two additional references.

In the instant proceeding, Petitioner relies on Lyke for teaching most of the same limitations as the Examiner, with a few exceptions. For the embedded memory recited in limitation [1.3], Petitioner relies on Frantz, whereas the Examiner relied on DaCosta. Pet. 24–26; Ex. 1010, 57–58. For the gate array recited in limitation [1.5], Petitioner relies on Lyke and Dehkordi, whereas the Examiner relied on Cloutier. Pet. 27–29; Ex. 1010,

---

[12] Issued claim 1 essentially is originally-filed dependent claim 3, rewritten in independent form. *See* Ex. 1010, 60 (office action objecting to originally filed claim 3, indicating it would be allowable if rewritten in independent form); *see also id.* at 42 (applicant remarking, *inter alia*, that originally filed claim 3 is amended to be independent); *see also id.* at 33 (amending originally filed claim 3).

[13] For the file history (Exhibit 1010), which is a multi-document exhibit, our citations refer to the pagination at the bottom center of each page.

[14] U.S. Patent Publication No. 2002/0129191 A1 ("DaCosta") (Exhibit 1041).

[15] U.S. Patent No. 5,892,962 ("Cloutier") (Exhibit 1040).

58. Finally, as to limitation [1.6], Petitioner relies on Lyke, as did the Examiner, but Petitioner provides an alternative argument that Faura additionally teaches this limitation. Pet. 29–31; Ex. 1010, 57.

As to limitations [1.7] and [1.8], the Examiner did not find these to be taught in the prior art. Ex. 1010, 60. For these limitations, Petitioner relies on Dehkordi. Pet. 33–34. The table below summarizes the Examiner's and Petitioner's respective positions.

Claim Limitation	**Reference(s) relied on by Examiner**	**Reference(s) relied on by Petitioner**
[1.pre] An instrument controller comprising	Lyke	Lyke
[1.1] a non-volatile memory storage component for program and data storage	Lyke	Lyke
[1.2] a large volatile memory storage component for additional program and data storage	Lyke	Lyke
[1.3] a processor coupled to both the non-volatile memory storage component and the large volatile memory storage components, the processor capable of high-frequency and low-frequency operations and having an embedded memory for storing an initialization program that enables the processor to start up processing without first retrieving a program from the non-volatile memory	DaCosta (for "embedded memory" as claimed) Lyke for remainder of limitation	Frantz (for "embedded memory" as claimed) Lyke for remainder of limitation
[1.4] at least two internal oscillators coupled to the processor, for providing clock signals for the low-frequency and high-frequency operations	Lyke	Lyke

Claim Limitation	Reference(s) relied on by Examiner	Reference(s) relied on by Petitioner
[1.5] a plurality of gates arranged in a field programmable gate array, the gate array coupled to the processor and configured to run independent processes in parallel with the processor	Cloutier	Dehkordi Lyke (each reference separately teaches limitation)
[1.6] a plurality of analog-to-digital converters for receiving a plurality of analog inputs, digitizing the analog inputs at one of at least two possible bit depths, thereby generating digital inputs, and providing, the digital inputs to the processor	Lyke	Lyke Faura (separately teaches "plurality of [A/D] converters")
[1.7] wherein a first portion of the gates in the field programmable gate array is configured to perform signal processing	Did not assert prior art	Dehkordi
[1.8] wherein a second portion of the gates in the field programmable gate array is configured to operate as a signal distribution matrix for rerouting signals within the instrument controller	Did not assert prior art	Dehkordi

As to claim 1, Patent Owner disputes only whether the combination of Lyke and Dehkordi teaches the gate array recited in limitation [1.5]. PO Resp. 29–52.

F. Obviousness of Claim 1

Petitioner asserts claim 1 is unpatentable under 35 U.S.C. § 103 as obvious over the combination of Lyke, Dehkordi, Frantz, and Faura. Pet. 12–34.

Discussed further below, Patent Owner argues that Lyke and Dehkordi do not teach the field programmable gate array recited in limitation [1.5]. PO Resp. 29–52. Patent Owner does not otherwise provide arguments or evidence regarding this ground of unpatentability. *See id.*

We have considered Petitioner's evidence and arguments, including the relevant testimony of Dr. Shanfield, as well as the evidence and arguments presented by Patent Owner, including the relevant testimony of Mr. Peck. Based on this record, we determine Petitioner has demonstrated unpatentability by a preponderance of the evidence.

We first address limitation [1.5]. In so doing, we highlight the following; however, these highlights should not be interpreted as limiting our findings and conclusions.

1. Lyke

Petitioner provides evidence and arguments that limitation [1.5] of claim 1, reproduced below, would have been obvious in view of Lyke:

> [1.5] a plurality of gates arranged in a field programmable gate array, the gate array coupled to the processor and configured to run independent processes in parallel with the processor.

Pet. 27–28.

Petitioner relies on Lyke's teaching that the disclosed MCM may include, in some variations, field programmable gate array(s). *Id.* Specifically, Lyke discloses:

> The matter set forth in the foregoing description and accompanying drawings is offered by way of illustration only and not as a limitation. Other variations to the current design include . . . RAM-based ***field programmable gate array(s) [FPGAs]*** interfaced internally within the AIC and configured

> automatically through a downloading mechanism involving the AIC's internal CPU and access of the non-volatile memory.

Ex. 1003, 8:48–59 (emphasis added).

As to the recitation that the field programmable gate array is coupled to the processor, Dr. Shanfield testifies that it would have been understood that the FPGA taught in Lyke is coupled to Lyke's processor because Lyke's FPGA is configured by downloading configuration information from the processor. Ex. 1002 ¶ 82; Ex. 1003, 8:55–58 (stating the FPGA is configured automatically through a downloading mechanism involving AIC's internal CPU). Dr. Shanfield further testifies that coupling an FPGA to a processor was well known in applications such as those disclosed in Lyke. Ex. 1002 ¶ 82. In support of this testimony, Dr. Shanfield relies on Halverson[16] to demonstrate his knowledge of one of ordinary skill in the art. *Id.* (citing Ex. 1025, 2:50–52, 27:11–14). Patent Owner does not contest that Lyke's FPGA is coupled to the processor.

As to the recitation that the field programmable gate array is configured to run independent processes in parallel with the processor, Petitioner asserts that Lyke's disclosure of using an FPGA (i.e., Ex. 1003, 8:55–59) "was sufficient to teach the challenged element to the POSITA, who would have come to Lyke with the common knowledge of a POSITA as reflected in exemplary prior art." Pet. Reply 14–15 (citing Ex. 1050 ¶¶ 105–109; Ex. 1061, 301; Ex. 1062, 167). According to Petitioner, the '177 patent did not invent an FPGA that runs independent parallel processes, and this

[16] U.S. Patent No. 5,574,930. (Exhibit 1025, "Halverson").

would have been a well-known feature of an FPGA. Pet. 27–28; Ex. 1002 ¶ 83; Pet. Reply 15 (citing Ex. 1002 ¶ 29).

Petitioner presents persuasive evidence in support of its argument. Dr. Shanfield explains that configuring an FPGA to run processes that run independently in parallel with a processor was typical in the relevant time frame. Ex. 1002 ¶ 83. According to Dr. Shanfield, in the relevant time frame the ability to run separate processes simultaneously (i.e., in parallel) was one of the fundamental features of FPGAs. Ex. 1050 ¶ 105 (citing Ex. 1058, 2; Ex. 1060, 9).

To corroborate his testimony, Dr. Shanfield relies on both Kostarnov[17] and Halverson. Ex. 1002 ¶ 83. For example, regarding independence between a CPU and FPGA, Kostarnov discloses "[t]he CPU and FPGA execute as two independent execution units." Ex. 1036, 4 (*quoted in* Ex. 1002 ¶ 83). As to parallel processes, Halverson states "the concept of parallel processing and the use of field programmable gate arrays is not new." Ex. 1025, 2:27–32 (*quoted in* Ex. 1002 ¶ 83). According to Dr. Shanfield, an ordinarily skilled artisan would have "readily recognized and been prompted to pursue the known benefits of using an FPGA to run independent processes in parallel with the processor . . . and would have had a reasonable expectation of success in doing so." Ex. 1002 ¶ 83.

Kostarnov supports Petitioner and Dr. Shanfield. Kostarnov relates to providing systems for processing high performance media algorithms such as video decompression. Ex. 1036, 1. Kostarnov explains that media

[17] Kostarnov, Igor, et al., "A Reconfigurable Approach to Low Cost Media Processing," Appliance Computing Dept., Hewlett-Packard Laboratories. (Exhibit 1036, "Kostarnov").

algorithm processing by a CPU may be accelerated by using, e.g., application-specific integrated circuits (ASICs), digital signal processors (DSPs), and application-specific instruction processors (ASIPs). *Id.* However, these acceleration techniques suffered from various drawbacks including the inability to program and/or reprogram the accelerators, according to Kostarnov. *Id.* at 1–2. Kostarnov, therefore, proposes an FPGA accelerator as an improvement, explaining that unlike previous accelerators, an FPGA can be reprogrammed. *Id.* Kostarnov discloses three modes of interaction between the CPU, FPGA, and memory.

Pertinent here, Kostarnov discloses a datapath mode (relied on by Petitioner and Dr. Shanfield, (Pet. 28 (citing Ex. 1002 ¶ 83)) in which the CPU and FPGA operate as two independent execution units. Ex. 1036, 4. Kostarnov explains that datapath mode provides the best potential performance, and is "particularly effective when the FPGA can process a large independent part of the algorithm without the need for complex control and synchroni[z]ation with the CPU." *Id.* at 11. We are persuaded that Kostarnov's data path mode describes an FPGA that is configured to run independent processes in parallel with the processor because in this mode the CPU and FPGA run as independent execution units, and because the FPGA can process portions of the algorithm without need for complex control and synchronization with the CPU. *Id.* at 4, 11. Kostarnov, therefore, corroborates Dr. Shanfield's testimony that configuring an FPGA to run processes that run independently in parallel with a processor was typical in the relevant time frame and known to one of ordinary skill in the art. Ex. 1002 ¶ 83.

Dr. Shanfield also relies on Diessel,[18] Fornaciari,[19] and Levinson[20] to corroborate his testimony that in the relevant time frame the ability to run separate processes simultaneously (i.e., in parallel) was one of the fundamental features of FPGAs. Ex. 1050 ¶ 105 (citing Ex. 1058, 2[21]; Ex. 1060, 9[22]; Ex. 1061, 301). The cited references support Dr. Shanfield's testimony. Diessel supports Dr. Shanfield's testimony that FPGAs were known in the art and used for their ability to perform independent processes in parallel, stating that FPGAs "that allow partial reconfiguration at run-time can be shared among multiple *independent* tasks." Ex. 1058, 2 (emphasis added). Fornaciari supports Dr. Shanfield's testimony that FPGAs were known and used in the art for their ability to perform independent processes in parallel as follows. Fornaciari discloses that FPGAs have been developed

---

[18] Deissel, Oliver, et al., "Dynamic Scheduling of Tasks on Partially Reconfigurable FPGAs," IEEE Proceedings—Computers and Digital Techniques, vol. 147, no. 3, pp. 181–188, ISSN 1350-2387 (2000) (Ex. 1058, "Diessel").

[19] W. Fornaciari, V. Piuri, "Virtual FPGAs: Some steps behind the physical barriers," In: Romlin J. (eds) Parallel and Distributed Processing, IPPS 1998, Lecture Notes in Computer Science, vol 1388, Springer, Berline, Heidelberg (1998). (Ex. 1060, "Fornaciari").

[20] L. Levinson, R. Manner, et al., "*Preemptive multitasking on FPGAs*," Prov. 2000 IEEE Symposium on Filed-Programmable Custom Computing Machines, Napa Valley, CA, USA, pp. 301–302 (2000). (Ex. 1061, "Levinson").

[21] Petitioner cites to document production numbers added to the bottom of each page rather than the original pagination of the reference. To avoid confusion, we adopt Petitioner's citation convention.

[22] Dr. Shanfield quotes Ex. 1060 and cites to page 9, but this appears to be in error. Ex. 1060 does not have a page 9. In addition, the quote appears at page 3 of Ex. 1060. We assume, therefore, that Dr. Shanfield is referring to page 3.

"to provide an efficient computational support for highly demanding applications whenever use of general purpose microprocessors cannot satisfy performance requirements or the ASIC solution is either far too expensive, or the time-to-market must be short." Ex. 1060, 1. In pertinent part, Forniciari describes multitasking systems, stating that "concurrent tasks may need to use the FPGA to perform specific (usually, *independent and unrelated*) algorithms in hardware so as to achieve the performance required by the corresponding applications." Ex. 1060, 3 (emphasis added).

Levinson supports Dr. Shanfield's testimony that FPGAs were known and used for their ability to perform processes in parallel, stating that FPGAs are suited to parallel execution of tasks in a multitasking environment. Ex. 1061, 301.

Accordingly, we credit Dr. Shanfield's testimony that, in the relevant time frame, configuring an FPGA to run processes that run independently in parallel with a processor was typical (Ex. 1002 ¶ 83) and that the ability to run separate processes simultaneously (i.e., in parallel) was one of the fundamental features of FPGAs (Ex. 1050 ¶ 105).

Petitioner further argues that as taught in Lyke, configuring an FPGA from non-volatile memory—i.e., retrieving program code from something that is not the processor—would have indicated that such programs run independently of the program code running separately on the processor. Pet. 27–28.

Patent Owner contends that Petitioner's arguments are entitled to little to no weight on the grounds that during prosecution the Examiner found Lyke neither teaches nor suggests limitation [1.5]. PO Resp. 29–32.

Patent Owner asserts that the '177 patent discloses details of Lyke's AIC, disclosing that the AIC "suffers from several limitations, such as its *inability to run parallel and independent processes*." *Id.* at 30. According to Patent Owner, "[t]his theme is repeated throughout the '177 Patent's specification." *Id.* at 31 (citing Ex. 1001, 2:9–18; Ex. 2020 ¶ 58). Patent Owner further points out the Examiner's finding that "Lyke fails to disclose . . . a field programmable gate array coupled to the processor for parallel processing." *Id.* at 32 (quoting Ex. 2006, 4, but failing to mention that on the next page the Examiner finds nonetheless that this feature would have been obvious in view of the combination of Lyke with Cloutier); Ex. 1010, 57.

We disagree that Petitioner's evidence and arguments concerning Lyke should be given little to no weight.

As an initial matter, the Examiner found limitation [1.5] would have been obvious in view of Lyke combined with another reference, i.e., Cloutier, and patent applicant did not dispute this finding. Ex. 1010, 58 (Examiner finding that limitation [1.5] would have been obvious in view of the combination of Lyke with Cloutier); *id.* at 14 (patent applicant canceling claim containing limitation [1.5] without disputing the Examiner's finding that this limitation would have been obvious in view of the combination of Lyke with Cloutier).

Moreover, to the extent the Examiner considered Lyke, there is no indication in either the '177 patent or the prosecution history of the patent that the Examiner considered the portion of Lyke upon which Petitioner relies. Pet. Reply 14 (citing Ex. 1003, 8:55–59; Pet. 27–28, 32) ("neither the applicant nor the Examiner noted Lyke's FPGA findings").

In particular, in asserting Lyke cannot run independent processes in parallel with the processor, the '177 patent discusses Lyke's ASIC. In particular, Lyke discloses:

> [T]here are several limitations to the design in [Lyke]. First, the AIC is not capable of performing parallel independent processes. Although the AIC does include a specialized ASIC, it is controlled and clocked by the microprocessor and cannot run independently of the microprocessor. Accordingly, independent parallel processes are not possible.

Ex. 1001, 2:9–15;

> [T]he circuitry design of the present invention eliminates the need for a resistive ASIC. Instead, the present invention adds a separately controllable FPGA that acts as a parallel processor with internal or separate external clock. The FPGA . . . adds a freely re-configurable and separately programmable multi-purpose digital system which can run independent of the microprocessor.

Id. at 2:47–55; and

> The design of the present invention eliminates the need for an analog application-specific intended circuit (ASIC). Instead, present invention utilizes an independently operable and programmable FPGA which is used to implement many of the key instrumentation functions of an ASIC.

Id. at 7:35–41.

As these excerpts evince, the '177 patent's statements regarding inability to perform parallel independent processes refer to ASICs in Lyke's AIC.

Neither the '177 patent, nor the Examiner, mentions, or otherwise acknowledges, Lyke's disclosure of an FPGA. Specifically, Lyke discloses that

> The matter set forth in the foregoing description and accompanying drawings [i.e., involving ASICs] is offered *by way of illustration only and not as a limitation. Other variations to the current design include* . . . [a] RAM based *field programable gate array [] [FPGA]* interfaced internally within the AIC and configured automatically through a downloading mechanism involving AIC's internal CPU and access of the non-volatile memory might also be included.

Ex. 1003, 8:48–59 (emphasis added).

There is no mention in either the '177 patent or the prosecution history of Lyke's disclosure of utilizing an FPGA in "other variations" of the AIC, much less discussion of whether an FPGA implemented in Lyke's AIC would have been capable of performing parallel independent processes. In view of the above, we find no evidence that the Examiner considered Lyke's disclosure of an FPGA.

Having addressed Patent Owner's contention that we should accord little to no weight to Petitioner's evidence and arguments regarding Lyke, we now turn to Patent Owner's substantive arguments concerning Lyke.

First, Patent Owner argues that mere mention of an FPGA in Lyke is not sufficient. PO Sur-Reply 9. Patent Owner argues that it is not necessarily the case that Lyke's FPGA runs independently in parallel with the processor simply by virtue of having logic configured from non-volatile memory. PO Resp. 32–33; *see* Ex. 1003, 8:55–59 (stating Lyke's FPGA may be RAM based "and configured automatically through a downloading mechanism involving AIC's internal CPU and access of the non-volatile memory"). Patent Owner explains, "[a] set of unspecified 'program instructions' does not confer on any FPGA an unstated ability to run 'independent parallel operations.'" *Id.* at 32. This argument is unavailing

because Petitioner's showing does not depend exclusively on Lyke's disclosure of an automatic downloading mechanism for obtaining program instructions for the FPGA.

As we discussed above, Petitioner's showing relies also on the knowledge of an ordinarily skilled artisan, as evinced by, e.g., Kostarnov, Diessel, Fornaciari, and Levinson. Pet. 28 (citing Ex. 1036, 4; Ex. 1025, 2:1–31); *see also, e.g.*, Ex. 1002 ¶ 83 (quoting Ex. 1036, 4); Pet. Reply 14–15 (citing Ex. 1050 ¶ 105 (citing Ex. 1058, 2; Ex. 1060, 9; Ex. 1061, 301)).

Patent Owner does not address Petitioner's showing of the knowledge of a person of ordinary skill in the art at the time of the invention (e.g., as evinced in Diessel, Fornaciari, or Levinson), and does not provide evidence and argument disputing or otherwise undermining Petitioner's showing that FPGAs were well known for and used in the relevant time frame for performing independent processes in parallel with each other or with processes run on another processor. To the extent Patent Owner argues, nonetheless, that Lyke does not teach or suggest claim limitation [1.5], Patent Owner's arguments are based on its construction of "independent processes," and Patent Owner does not address whether the art would have taught or suggested the limitation at issue under the construction adopted here—i.e., Patent Owner does not undermine Petitioner's showing that a skilled artisan would have known or understood FPGAs in Lyke would have run separate processes.

Patent Owner does not provide any substantive argument regarding the knowledge of skill in the art reflected in Kostarnov, except with reference to a mode of operation in Kostarnov that Petitioner does not rely on, which we address below. PO Resp. 39–42. Rather, its argument

essentially is that Petitioner improperly relies on Kostarnov because this reference is not asserted in the ground of unpatentability. PO Resp. 50–52. This argument is unavailing. Petitioner does not combine Kostarnov with other references, but rather this reference is used as corroboration of Dr. Shanfield's knowledge of an ordinarily skilled artisan when reading Lyke. Pet. 28 (citing Ex. 1002 ¶ 83; Ex. 1036) (stating that running independent processes in parallel was a "standard way" to deploy FPGAs with processors, and citing Kostanov to support this statement); Ex. 1002 ¶ 83 (citing Ex. 1036, 4) (Dr. Shanfield opining that running independent processes in parallel is how FPGAs were typically deployed in processors, and citing Kostarnov to support his testimony). We find no error in Dr. Shanfield's disclosure of references as evidence to corroborate testimony. *Cf.* 37 C.F.R. § 42.65(a) ("Expert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little to no weight.").

Patent Owner asks us to disregard Lyke's express disclosure that FPGAs may be used, arguing there would have been no pragmatic use for an FPGA in Lyke. PO Resp. 33 (citing Ex. 2020 ¶ 62). Patent Owner relies on a strawman argument to support its contention. Specifically, Patent Owner argues that neither the resistive nor analog ASIC in Lyke would have been implemented using an FPGA at the time of Lyke's filing. PO Resp. 34–35. This argument is unavailing because the relevant question is what a skilled artisan would have understood at the time of the invention of the '177 patent (filed December 19, 2001, Ex. 1001 (22)), not at the time of Lyke's filing (filed October 26, 1998, Ex. 1003 (22)). In addition, neither Petitioner nor Lyke argue or otherwise suggest Lyke's FPGA is intended to implement

Lyke's resistive ASIC or analog ASIC, so Patent Owner's point is inapposite.

Expanding upon its inapposite arguments concerning Lyke's ASICs, Patent Owner also argues that neither the resistive nor analog ASIC in Lyke is capable of autonomous, independent parallel operation. PO Resp. 33–36. We find this argument unavailing because it fails to address the actual combination upon which Petitioner relies. Petitioner does not rely on Lyke's ASICs, but rather on Lyke's FPGA. Pet. 27–28.

Patent Owner also argues that, even if Lyke's system were to include an FPGA that incorporates a feature that replaces one implemented in Lyke's ASICs, such FPGA necessarily would be governed by the same processor as the ASICs, and therefore would "suffer from the deficiencies that inspired the '177 [patent]." PO Resp. 36 (citing Ex. 2020 ¶ 67). As discussed immediately above, Patent Owner's attempts to limit use of an FPGA in Lyke to functions of Lyke's ASICs is unwarranted, as Petitioner does not argue any such limitation in its ground of unpatentability and Lyke does not suggest any such limitation.

Patent Owner also asserts that any FPGA implemented in Lyke necessarily would both 1) operate using the processor clock in Lyke's AIC, and 2) necessarily would run dependent processes as a result of running on the processor clock. PO Resp. 33–36; *see also* PO Sur-Reply 11–12 (citing Ex. 2020 ¶¶ 60, 63, 67 (citing Ex. 1003, Fig. 12)) (arguing that the processor in Lyke supplies the clock to the AIC). We find Patent Owner's argument unavailing for reasons disclosed below regarding Patent Owner's shared clock arguments. *Infra* Sec. II.F.2.

For the foregoing reasons, we find that Petitioner has presented sufficient evidence and argument that limitation [1.5] would have been obvious in view of Lyke.

2. Dehkordi

Petitioner provides evidence and arguments that limitation [1.5] would have been obvious in view of the combination of Lyke with Dehkordi. Pet. 28–29. Petitioner relies on Dehkordi's teaching of a Xilinx 4010 FPGA coupled with Motorola 96002 DSP processor within a MCM. Pet. 28–29; *see* Ex. 1004, 1 (describing implementing the DSP processor and FPGA on a MCM). As can be seen in Figure 1 of Dehkordi, the FPGA is coupled to the DSP processor. Ex. 1004, 1; *see also* reproduction and discussion of Dehkordi's Figure 1 below in this subsection. Patent Owner does not contest that the two are coupled.

As to the gate array being configured to run independent processes in parallel with the processor, Petitioner argues that Dehkordi's FPGA's multi-processing bus arbitration functionality runs "independently and in parallel to whatever computation is happening on the processor (and other portions of the FPGA)." Pet. 28. As Dr. Shanfield explains, bus arbitration essentially is a "traffic cop" function in which data is routed for processing by either the FPGA or DSP. Ex. 1002 ¶ 84. Petitioner argues persuasively that traffic routing is an independent process because it does not depend on other processes run on the DSP processor, i.e., it runs separately from the processor. Dr. Shanfield also relies on Dehkordi's disclosure of the FPGA pre-processing incoming data as teaching independent processing. *Id.* Petitioner argues persuasively that pre-processing of incoming data is an

independent process because it does not depend on other processes run on the DSP processor, i.e., it runs separately from the processor.

Petitioner further submits that additional processes run on the FPGA run independently and in parallel with the processor. Pet. 28–29. These processes include (1) processes controlling an analog-to-digital interface, (2) power management processes, and (3) digital signal processing. *Id.*; *see also* Ex. 1004, 1 (Figure 1, "ADC Interface," "Power-up Circuit," and "DSP Functions" blocks within the Xilinx FPGA). In support of its position, Petitioner relies on Dehkordi's teaching that the FPGA processes are from "RAM-based configuration where the configuration may come from outside the module," which indicates, according to Petitioner, that these processes are separate from those running on the DSP processor. *Id.* at 29 (*quoting* Ex. 1004, 3). Petitioner relies on Dr. Shanfield's testimony to support its arguments. *Id.* (*citing* Ex. 1002 ¶¶ 83–84).

Patent Owner does not address whether the combination of Lyke with Dehkordi teaches/suggests the limitation at issue under the correct construction of "independent processes"—a construction of which Patent Owner was on notice and had a full and fair opportunity to respond. The construction adopted herein was suggested in the Decision on Institution[23]

[23] In the Decision on Institution, in our claim construction analysis we stated that "it appears that Petitioner understands the term 'independent process' indicates a process that runs separately from those that run on the processor." Inst. Dec. 13. We also stated that "[b]y way of example, the *Microsoft Computing Dicitonary*[11] defines 'process' as '[a] program or part of a program; a coherent sequence of steps undertaken by a program." *Id.* 14 (citing Ex. 3002).

and proposed by Petitioner and/or Dr. Shanfield as discussed above, *supra* Sec. II.B.2.

Instead, Patent Owner responds that the processes that run on Dehkordi's FPGA are *dependent* on the DSP processor, because "the DSP processor and the FPGA are connected through a control bus [] used to control the FPGA, with both components relying on the same clock, and which a POSA would understand were synchronized." PO Resp. 38 (emphasis omitted) (footnote omitted). Patent Owner contends that "[a]s such, the FPGA is *dependent* on the processor, and [the FPGA and DSP processor] are necessarily in a dependent configuration. Consequently, no independent processes are possible during normal operation and neither the Petitioner nor [Dehkordi] suggest otherwise." *Id.* at 38–39.

In the Institution Decision, we placed the parties on notice that there was insufficient evidence showing that when an FPGA and processor share a control bus and/or clock signal, the FPGA necessarily depends on the processor, and we encouraged the parties to develop the record as to this issue. *See* Inst. Dec. 37 n. 24. We stated:

> Patent Owner has not shown that sharing a common clock and/or common control bus means the FPGA necessarily acts as a slave to the processor. To the extent Patent Owner's argument is that independent processes cannot run on (1) a common clock, or (2) a common bus, Patent Owner has not developed the record to show why this must necessarily be true. The parties may wish to develop the record as to whether independent processes may share common clocks and/or may share common buses.

Id. (emphasis omitted).

However, Patent Owner does not provide evidence in response to our notice or in support of its assertion that sharing a common control bus and/or

clock necessarily means processes are not independent. PO Resp. 38–39. Although Petitioner bears the ultimate burden of persuasion, here Petitioner has met that burden. As we discussed above in our claim construction, a "process" is "a program or part of a program; a coherent sequence of steps undertaken by a program," and "independent processes" are "processes that run separately from one another." *Supra* Sec. II.B.1–2. In other words, "independent processes" are programs/parts of programs or coherent sequences of steps undertaken by a program that run separately from one another. Here, Petitioner has provided sufficient evidence of FPGAs and CPUs running as independent execution units, and of FPGAs partitioning tasks to run separately, as discussed above. In addition, as discussed above, Dehkordi describes processes/functions performed by the FPGA that are separate from those run on the processer, including pre-processing and bus arbitration.

Patent Owner's argument in its Response concerning sharing common clocks and bus control lines appears to be one of claim construction, namely that when one unit is subservient to the other, their processes by definition are not "independent" despite their processes being sequences of steps that run separately. PO Resp. 38–39. However, Patent Owner appears to abandon this argument in the Sur-Reply, conceding that its claim construction of "independent processes" is agnostic to whether the FPGA and processor share the same clock and that the question of independence does not turn on whether the clock is shared. PO Sur-Reply 12.

Tellingly, Patent Owner does not address whether Dehkordi's FPGA runs separate programs/coherent sequences of steps. Rather, Patent Owner asserts that Dehkordi's FPGA is "subservient" to Dehkordi's DSP processor

and is "directed" by it, without addressing whether the FPGA can or does run any separate programs/coherent sequences of steps:

> [Dehkordi's] technical description [] explains how the MCM subsystem is designed to exploit the multiprocessing capability *of the DSP processor*, which indicates that the FPGA operates in *a subservient support role* under the control of the DSP processor to maximize the *processor's* multiprocessing functionality by, *e.g.*, preprocessing incoming data *as directed by the DSP processor*.

PO Resp. 39 (citing Ex. 1004, 89[24], Fig. 1 and its associated description in the "Design Flow" section; Ex. 2020 ¶ 71). In other words, Patent Owner's argument about subservience and control is not commensurate with the language or scope of what is claimed.

In addition to failing to address whether Dehkordi's FPGA runs separate programs/coherent sequences of steps, Patent Owner's assertion regarding subservience and control is conclusory. The cited portion of Dehkordi does not mention the FPGA in a subservient role or direction by the processor. On the contrary, the cited portion refers to generic functions required to implement the entire subsystem, and does not mention performing functions at the direction of the processor, stating the "FPGA is primarily a hardware pre-processor of incoming data but also provides other generic logic functions required for implementation of the entire subsystem." Ex. 1004, 89.

Even if we were to adopt Patent Owner's argument in the Response that sharing a processor clock precludes independent processes (which we

[24] Patent Owner refers to page "1" of Dehkordi, wherein "1" appears to be pagination provided by Petitioner. We instead cite to the original pagination of the Dehkordi publication.

do not), this would not change the outcome of our Decision. Petitioner provides evidence and argument that Dehkordi's FPGA has an internal system clock, and therefore runs processes independent of the processor even under Patent Owner's unproven premise that a common clock determines whether a process is dependent or not. Ex. 1050 ¶ 122 (in response to Patent Owner's statement that Dehkordi'a FPGA is subservient to the processor, testifying that a POSITA would have known Dehkordi's FPGA has independent internal clocking, as evinced by the data sheet for Dehkordi's FPGA); Pet. Reply 17.

Specifically, the Xilinx 4010—i.e., FPGA used in Dehkordi—includes an internal system clock that "allows processes to run on their own using its internal clock system." Pet. Reply 17 (citing Ex. 2019, 172 (disclosing the FPGA has "an internal system clock"); Ex. 1014, 6-46 (Xilinx 4000 series FPGA datasheet describing multiple modes of operation and internal clocking); Ex. 1050 ¶¶ 120–124 (citing Ex. 1014)); Ex. 1004, 89 (disclosing Xilinx 4010 FPGA).

Petitioner further provides evidence and argument that Dehkordi's FPGA has independent control. Pet. Reply 18–20. Dehkordi discloses that its FPGA processes have separate code that "could come from outside the module," and therefore is not provided by the processor which is inside the module. *Id.* at 18 (citing Ex. 1004, 3, which states "[t]he FPGA has a RAM-based configuration where the configuration may come from outside the module"; citing Ex. 1050 ¶¶ 125–136).

Accordingly, even if we were to accept Patent Owner's argument that a common clock precludes independent processes—and we do not for reasons discussed above—Petitioner has shown sufficiently that it was well

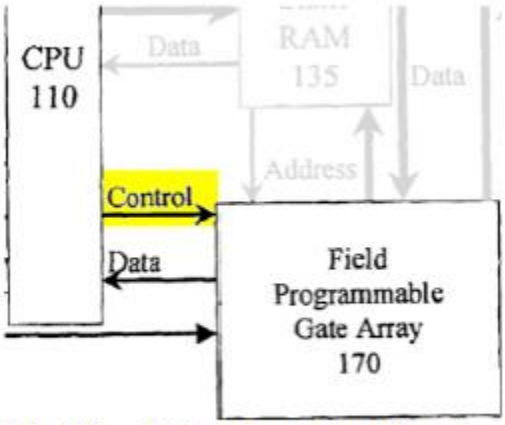
known for Dehkordi's FPGA to use its own internal clock rather than a clock in common with the DSP processor.

As to the sharing of control lines, Patent Owner does not explain how this is pertinent to whether Dehkordi's FPGA can run independent processes. PO Resp. 38–39. In pertinent part, Patent Owner does not provide evidence and argument that shows that the sharing of control lines precludes Dehkordi's FPGA from running separate programs/coherent sequences of steps. *Id.* In other words, Patent Owner does not explain how control lines being separate versus shared relates to "independent processes" under the correct interpretation—indeed, Patent Owner does not even explain how its own proposed construction of this term relates to control lines. Rather, Patent Owner merely asserts that the FPGA in Dehkordi is shown in Figure 1 as being connected through a control bus with both components relying on the same clock, then concludes the FPGA therefore necessarily is dependent on the DSP processor and necessarily these components are in a dependent configuration. *Id.* Patent Owner then concludes that "[c]onsequently, no independent processes are possible during normal operation." *Id.* at 39 (citing Ex. 2020 ¶ 68). The portion of Mr. Peck's declaration cited by Patent Owner does not support Patent Owner's conclusion. Ex. 2020 ¶ 68. It is merely a conclusory statement that Lyke, the intrinsic record, and "supporting references relied on in Petition" neither teach nor suggest limitation [1.5] regardless of whose construction of "independent processes" applies. *Id.*

For the foregoing reasons, Patent Owner's argument regarding control lines does not undermine Petitioner's showing.

In addition to the above reasons, which are sufficient to support our findings and conclusions regarding obviousness, Petitioner provides evidence and argument that despite Patent Owner's argument regarding control lines, Dehkordi teaches the FPGA running separate code not provided by the processor, and therefore, teaches running processes separately from the processor. Pet. Reply 18 (citing Ex. 1050 ¶¶ 125–136); Ex. 1004, 3.

In response to Patent Owner's statement that Figure 1 of Dehkordi's depiction of a control line shared between the FPGA and DSP processor necessarily means the FPGA cannot run independent processes, Petitioner points out that even the '177 patent shows a control line from the CPU to the FPGA, which indicates control signals are sent from the CPU to the FPGA. Pet. Reply 18–19; Ex. 1001, Fig. 1. The following figure is illustrative.



EX1001, Fig. 1 (extract, emphasis added)

Pet. Reply 19 (reproducing a portion of Figure 1 of the '177 patent (Ex. 1001, Fig. 1) with highlighting added). The Figure reproduced above is an excerpt of Figure 1 of the '177 patent showing the control line between CPU 110 and FPGA 170 highlighted in yellow. In view of Figure 1 of the '177 patent, Petitioner argues that depiction of a control line shared between

a CPU and FPGA is not the litmus test for independence. Pet. Reply 19 (citing Ex. 1001, 3:34–37; Ex, 1065, 15:24–16:10; Ex. 1050 ¶¶ 133–134). Dr. Shanfield points out that, because the '177 patent shows a common control line yet discloses the processor and FPGA running independent processes, the inclusion of a common control line in a figure "doesn't restrict the 'independent processes' running on the FPGA in any way.'" Ex. 1050 ¶ 131.

Petitioner notes that the control line arrow in Figure 1 of the '177 patent is unidirectional versus bi-directional as shown, e.g., in Figure 1 of Dehkordi (*see* Ex. 1004, Fig. 1). Pet. Reply 19. However, we find such distinction to be inapposite. In the '177 patent, because it is the CPU sending control signals to the FPGA, it appears that this would indicate, if anything, the CPU controls the FPGA. A bi-directional arrow (i.e., showing the FPGA can send control signals to the CPU) would not negate this.

Moreover, despite the depiction of control lines in various figures, Petitioner argues that the related control signals involving Dehkordi's FPGA "are not about *process* control." Pet. Reply 19 (citing Ex. 1050 ¶ 132 (citing Ex. 1014)). As Dr. Shanfield testifies, "[l]ow level control lines, like the control signals disclosed on the Xilinx 4000 series FPGA present in Dehkordi are not about process control." Ex. 1050 ¶ 132. Patent Owner does not respond to this argument.

For the foregoing additional reasons, we find Patent Owner's arguments regarding control lines do not undermine Petitioner's showing regarding Dehkordi combined with Lyke.

Patent Owner improperly introduces a new argument in the Sur-Reply. Specifically, Patent Owner changes tack regarding its arguments and

construction of the term "independent processes," abandoning its arguments about common clocks and shared bus control lines, and instead argues that claim limitation [1.5] requires that "independent processes" occur at a "system level," rather than "the sub-application level." PO Sur-Reply 10–13. Patent Owner implicitly acknowledges this argument is one of claim construction, but does not provide any analysis, evidence, or argument showing why this is the proper construction of limitation [1.5]. *Id.* at 10 (asserting that "the claim term involving 'independent processes' must be interpreted within the claimed 'instrument controller' as [a] whole, which requires viewing the 'independent processes' on a system level instead of the sub-application level.") (footnote omitted). Furthermore, Patent Owner does not clarify the boundaries of its proposal that a "system level" is the appropriate measure of whether processes are independent, much less how the claim language and '177 specification—which does not mention "system level" versus "sub-application" level processes—supports its new claim construction.

In addition, Patent Owner does not show good cause to justify introducing this new claim interpretation in the Sur-Reply, for which Petitioner did not have the opportunity to respond as of right. Patent Owner attempts to treat this new claim construction as the same as its original construction. Patent Owner argues that "without computational assistance" means at a system level rather than sub-application level. PO Sur-Reply 10–11. However, we disagree that "system level" independence means the same thing as "without computational assistance" from one another. Also, we find the phrases "system level" versus "sub-application level" vague and

unhelpful in clearly defining the contours of what it means for processes to be considered independent.

Patent Owner provides some purported guidance concerning what is meant by "system level," arguing essentially that if a computational output of circuit A is input to and used by circuit B to perform a process, the two processes are dependent. PO Sur-Reply 13. We do not discern how the phrase "system level" is helpful in describing this scenario or in resolving claim construction. However, focusing on Patent Owner's argument regarding inputs and outputs of circuits, rather than on the phrases "system level" versus "sub-application level," Patent Owner has not shown that the correct construction of "independent processes" are processes in which the computational output of circuit A cannot be input and used by circuit B to perform a process. Patent Owner has not shown that such construction is supported by the intrinsic evidence of record, and we do not alter our claim construction of "independent processes" in view of this new argument.

Even if we were to adopt Patent Owner's implicit construction, Patent Owner's argument that the combined prior art would have involved circuit B receiving computational input from circuit A rests on a combination that Petitioner does not rely on, namely a combination involving Dehkordi 2. PO Sur-Reply 13. For this additional reason, Patent Owner's arguments do not undermine Petitioner's showing.

In particular, Patent Owner introduces a new reference that is not relied on in the Petition, which Patent Owner refers to as "Dehkordi 2" or "D2." PO Resp. 17–18, 39–42. Patent Owner asserts that Dehkordi 2 is "a highly relevant reference" that "describes a more complete version of the MCM subsystem by the same authors [of Dehkordi]." *Id.* Patent Owner

selects description of specific features disclosed in Dehkordi 2 and asserts the system disclosed in Dehkordi necessarily operates in the same way and has the same features. *Id.* at 17–18. We disagree that Dehkordi 2 is a more complete version of the system taught in Dehkordi.

Dehkordi's abstract states "[t]his paper discusses the design and development of a general-purpose programmable DSP subsystem packaged in a multi-chip module." Ex. 1004, 89. The purpose of the design is to achieve low-volume MCM prototyping that is "achievable and somewhat affordable for universities." *Id.* (emphasis omitted). Dehkordi explains, therefore, that it discusses "design flow, electrical and thermal analyses, CAD tools, costs and lessons learned." *Id.* Dehkordi concludes that

> Low-cost, low volume prototyping capabilities are a "must" for university related programs as well as small/medium size companies who wish to utilize [MCM] technology. We have found out that accessing this technology is becoming a reality for university programs at *almost* low-cost. The MMS design kit was *very* useful not only with the physical layout but also since it provided a variety of check points and design tips. Availability of the bare dies (and their I/O buffer modules), testing and rework issues will continue to be challenging issues to be considered.

Ex. 1004, 91.

Dehkordi 2 is a paper directed to enhancing testability of an MCM that has an embedded FPGA. Ex. 2019, 165. Dehkordi 2 states that "MCM testing involves the verification of substrate interconnects, logical connections and IC interaction functionality," and that testing is difficult because MCM's include a heterogenous mix of components. *Id.* Dehkordi 2 discloses that "MCMs with an embedded reconfigurable FPGA in the design can enhance its testability," stating that "[t]he FPGA can be

configured as a pseudo built-in logic analyzer that can monitor core dies within the MCM in real time," which "allows testing of not only the hardware aspect, but also allows monitoring of software execution in a limited manner." *Id.* at 166–167.

However, Dehkordi 2 states that configuring an FPGA for testing requires "proper connectivity and configuration," and furthermore that it is not always feasible, stating "design constraints may not allow an embedded FPGA to be used for testing. For instance, to perform testing of a MCM may require the FPGA to be larger in gate size, have more I/O or require additional routing within the substrate." *Id.* at 169. In other words, Dehkordi 2's MCM designs, which provide connections and configurations that allow FPGAs to be used for MCM testing, are not always desirable or even possible. As such, we disagree with any suggestion that a skilled artisan reading Dehkordi 2 would have understood that its MCM designs necessarily must be applied to Dehkordi. On the contrary, it appears the designs in Dehkordi 2 had not been developed by the paper's authors at the time they authored Dehkordi. Moreover, we find that the designs disclosed in Dehkordi 2 were not intended to limit the designs that would have occurred to the ordinarily skilled artisan when reading Dehkordi, but rather would have been understood as possible designs only in situations where design constraints did not preclude using the embedded FPGA for testing.

Indeed, Dehkordi 2 makes reference to Dehkordi, but requires several modifications in order to adapt the MCM design to allow for testing. Ex. 2019 n. 11; *id.* at 169–170 (reproducing Figure 1 of Dehkordi). Dehkordi 2 emphasizes that, with regard to the design in Dehkordi, "[i]t is important to note that the FPGA in the MCM was not originally designed

into the system for testability. The FPGA was deemed as necessary glue logic to provide operational resources for the system. For example, the FPGA provides logic for an address decoder, an interactive user I/O interface and the means for reconfigurable computing with the DSP processor." *Id.* at 170. Among the several alterations and increased complexity to the design taught in Dehkordi in order to allow for testability, Dehkordi 2 discloses allowing the FPGA to be the main interface between the MCM and its environment and "wrapped around" connections on the printed circuit board to allow connections between the FPGA and the main address bus and control line of other ICs including the main processor unit. *Id.* at 170–171. Dehkordi 2 explains that this increased complexity is needed to provide a testing environment. *Id.*

In sum, the evidence does not support Patent Owner's argument that we must view Dehkordi through the lens of Dehkordi 2, much less that the only implementation of the circuit disclosed in Dehkordi is the "more complete version of the MCM subsystem" disclosed in Dehkordi 2. PO Resp. 17–18. Rather, Dehkordi 2 clearly teaches that its modified MCM subsystem is an adaptation that may be desirable in some circumstances—e.g., by configuring and providing connectivity that allows for using the FPGA for testing the MCM—modifications which Dehkordi 2 states are not always feasible. Ex. 2019 at 169–170.

For the foregoing reasons, we find that Dehkordi 2 (which is not even a reference relied on by Petitioner) does not detract from the teachings in Dehkordi or impose its modified design(s) as a necessary version of the design(s) disclosed in Dehkordi.

Because Patent Owner's argument that the combined prior art would have involved circuit B receiving computational input from circuit A relies on Dehkordi 2, we find it unavailing. PO Sur-Reply 13.

Petitioner also relies on Dehkordi 2 for another purpose, namely in providing its only substantive argument regarding Kostarnov. PO Resp. 40–42. Kostarnov discloses three modes of operation involving the FPGA and central processing unit ("CPU"). Ex. 1036, 3–4. The modes include functional unit mode, lockstep mode, and data path mode. *Id.* As discussed above, Petitioner relies on Kostarnov's ***datapath mode***, in which the CPU and FPGA operate as "independent execution units." Pet. 28 (citing Ex. 1036, 4); *see also* Ex. 1002 ¶ 83 (Dr. Shanfield relying on Kostarnov's datapath mode). Patent Owner does not address ***datapath mode***, much less dispute that it teaches or suggests the FPGA running independent processes in parallel.

Instead, Patent Owner argues that Kostanov's ***functional unit mode*** is the "relevant" mode of operation—even though this is not the mode identified by Dr. Shanfield to corroborate his testimony—and that such mode does not teach the claimed feature at issue. PO Resp. 40–42. Patent Owner's rationale behind arguing an operational mode not identified by Dr. Shanfield is that a skilled artisan would have understood that the FPGA in the test design of Dehkordi 2 would have used the functional unit mode. *Id.* However, as we discussed above, we disagree that Dehkordi is limited to the test designs described in Dehkordi 2, and therefore, we disagree that Kostarnov's functional unit mode is the relevant mode of operation as argued by Patent Owner. Patent Owner's arguments regarding Kostarnov, therefore, do not undermine Petitioner's showing.

Patent Owner also provides arguments regarding JTAG and TAP testing. PO Resp. 42–48. Patent Owner states that Dehkordi theorizes about "*a special testing mode of operation* that uses the well-known industry standard JTAG and TAP testing methodologies for testing the operation of the DSP system." *Id.* at 42. Patent Owner argues, in essence, that these testing modes of operation do not involve independent processes because in these modes an external controller controls both the DSP processor and FPGA during testing. *Id.* at 42–43. These arguments are inapposite because Petitioner does not rely solely on Dehkordi's operation in these testing modes. As we discussed above, Petitioner relies on Dehkordi as a whole, including teachings regarding pre-processing and bus arbitration. Pet. 28–29.

For the foregoing reasons, Petitioner has made a sufficient showing regarding limitation [1.5] in view of the combination of Lyke with Dehkordi.

3. Remaining Limitations of Claim 1

As to the remaining limitations of claim 1, as well as the rationale supporting obviousness, we have reviewed the arguments and evidence in the current record, and determine that Petitioner has made a sufficient showing. Pet. 12–32. Patent Owner does not dispute or argue the remaining limitations of claim 1.

As we discussed above, the '177 patent acknowledges that Lyke discloses several elements recited in claim 1. *Supra* Sec. II.E. Indeed, the '177 patent purports to improve upon Lyke's instrument controller and there is no dispute amongst the parties that Lyke discloses several features recited in claim 1. *See, e.g.*, Ex. 1001, 1:21–2:31; *see also* PO Resp. 29–59 (arguing only limitation [1.5]).

Petitioner shows sufficiently that Lyke teaches the preamble of claim 1, which recites "[a]n instrument controller," because Lyke discloses "a tightly coupled multi-chip Module (MCM)," which uses an "Advanced Instrument Controller (AIC)." Pet. 19 (citing Ex. 1003, Abstract, 1:58–62; Ex. 1002 ¶¶ 67–68).

Petitioner shows sufficiently that Lyke teaches limitation [1.1], which recites "a non-volatile memory storage component for program and data storage," because Lyke discloses "a non-volatile memory . . . used for both program storage and data storage." Pet. 19 (citing Ex. 1003, 4:14–15; Ex. 1002 ¶ 70).

Petitioner shows sufficiently that Lyke teaches limitation [1.2], which recites "a large volatile memory storage component for additional program and data storage," because Lyke discloses an "instrument controller comprising . . . a large volatile memory storage system with storage components comparable to or larger than the address space that the CPU operates upon," thereby providing memory for additional program and data storage. Pet. 21 (citing Ex. 1003, 9:15, 9:19–22, 4:35–41; Ex. 1002 ¶ 73).

Petitioner shows sufficiently that the combination of Lyke and Frantz teaches limitation [1.3]. Petitioner shows that Lyke teaches "a processor coupled to both the non-volatile memory storage component and the large volatile memory storage components," as recited in limitation [1.3], based on Lyke's disclosure of a "tightly coupled MCM design" with a processor coupled to the "non-volatile memory" and the "volatile static random access memory." Pet. 22 (citing Ex. 1003, 3:66–4:41, Fig. 1; Ex. 1002 ¶ 75). Petitioner shows that Lyke teaches "the processor capable of high-frequency and low-frequency operations," as recited in limitation [1.3], based on

Lyke's disclosure of an "AIC contain[ing] at least two [internal oscillators], one for high-frequency operation and one for low-frequency operation." *Id.* at 23 (citing Ex. 1003, 4:42–44). As argued by Petitioner, Lyke teaches that the "high and low oscillators are depicted as connected to the processor, and described as providing the 'high and low frequency internal reference' for Lyke's processor's high-frequency and low-frequency operations." *Id.* (citing Ex. 1002 ¶ 77 (citing Ex. 1003; Ex. 1026)).

Petitioner shows that Frantz combined with Lyke teaches "and having an embedded memory for storing an initialization program that enables the processor to start up processing without first retrieving a program from the non-volatile memory," as recited in limitation [1.3], based on Frantz's disclosure of a "TMS320C25 digital signal processor" which has "[f]our-k words of on-chip program ROM" for storing programs run by the processor. Pet. 24 (citing Ex. 1005, 2); *see also id.* at 25–26 (arguing Franz's teachings). As argued by Petitioner, Frantz taches that the read-only memory (ROM) is an embedded memory within the TMS320C25 processor. *Id.* at 24 (citing Ex. 1005, 2; Ex. 1002 ¶ 78). Petitioner further shows that an ordinarily skilled artisan would have recognized that Frantz's teaching of up processing with the ROM-stored initialization program without first retrieving a program from the non-volatile memory, especially in view of Frantz contrasting that use with the alternate scenario of needing to first download a program from a "slow external memory," e.g., an off-chip nonvolatile RAM. *Id.* at 25–26 (citing Ex. 1005, 2–4, 13, 17, Fig. 6; Ex. 1002 ¶ 78). Petitioner further shows that using an embedded memory for storing an initialization program as recited in limitation [1.3] was a well-

understood microprocessor design approach. *Id.* at 26 (citing Ex. 1027, 2, 4; Ex. 1037; Ex. 1039, 2).

Petitioner shows sufficiently that Lyke teaches limitation [1.4], which recites "at least two internal oscillators coupled to the processor, for providing clock signals for the low-frequency and high-frequency operations," because Lyke discloses an "AIC contain[ing] at least two [internal oscillators], one for high-frequency operation and one for low-frequency operation," and shows them 'as coupled to the processor." Pet. 26 (citing Ex. 1003, 4:42–44; Figs 1, 2; Ex. 1002 ¶ 79 (citing Ex. 1026, 98:59–61, 14:16–17)); *see also id.* (arguing Lyke teaches that oscillators are used for "precise timing" and that "[a] typical AIC has an 11 MHz oscillator and 200 Hz oscillator for the high and low frequency internal reference," i.e., clocking signals) (citing Ex. 1003, 4:42–57; Ex. 1002 ¶ 79).

Petitioner shows sufficiently that the combination of Lyke and Faura teaches limitation [1.6]. Petitioner shows that Lyke teaches "a plurality of analog-to-digital converters for receiving a plurality of analog inputs," as recited in limitation [1.6], because Lyke teaches of a collection of analog function blocks including multiplexed or non-multiplexed analog to digital converters. Pet. 29–30 (citing Ex. 1003, 3:38–33, 7:30–32, Fig 1; Ex. 1002 ¶¶ 85–86). Petitioner's showing is further persuasive because Faura teaches use of analog to digital converters for receiving a plurality of analog inputs. *Id.* at 30–31 (citing Ex. 1006, 1, 2; Ex. 1002 ¶ 87 (citing Ex. 1030, 6:27–29; Ex. 1031, 10:5–12); *id.* at ¶ 88 (citing Ex. 1006, 1, 2; Ex. 1032, 3)).

Petitioner shows that the combination of Lyke with Faura teaches "digitizing the analog inputs at one of at least two possible bit depths," as recited in limitation [1.6], because Lyke teaches digitizing "a large number

of analog inputs at high resolution (equal to or greater than 10 bits)," wherein 10 bits and bit depths greater than 10 bits are "two possible bit depths." Pet. 31 (citing Ex. 1003, 28:33; Ex. 1002 ¶ 89) (emphasis omitted); *see also id.* (citing Lyke's disclosure of 10-bit and 12-bit resolution as another example of "two possible bit depths") (citing Ex. 1003, 4:58–5:5). Although not necessary in view of Lyke's teachings, Petitioner shows further that Faura teaches the language at issue, because Faura describes multiple examples of digitizing the analog inputs at one of at least two possible bit depths, including 9-bit ADCs and 8-bit ADCs at the same time. *Id.* at 32 (citing Ex. 1002 ¶ 90).

Petitioner shows that the combination of Lyke with Faura teaches "thereby generating digital inputs and providing the digital inputs to the processor," as recited in limitation [1.6], because the analog to digital converters in both Lyke and Faura each disclose digital signals output from the analog to digital converter to the processor. Pet. 32 (citing Ex. 1002 ¶¶ 91–92; Ex. 1029; Ex. 1030; Ex. 1003, Fig. 1, 2; Ex. 1006, Fig. 3).

Petitioner shows that the combination of Lyke with Dehkordi teaches "wherein a first portion of the gates in the field programmable gate array is configured to perform signal processing," as recited in limitation [1.7], because Dehkordi teaches a first portion of the gates in the FPGA being configured to perform signal processing, including pre-processing incoming data, as well as DSP functions. Pet. 33 (citing Ex. 1004, 1, Fig. 1; Ex. 1002 ¶ 93).

Petitioner shows that the combination of Lyke with Dehkordi teaches "wherein a second portion of the gates in the field programmable gate array is configured to operate as a signal distribution matrix for rerouting signals

within the instrument controller," as recited in limitation [1.8], because Dehkordi discloses "[a] portion of the FPGA is configured for multi-processing bus arbitration and is connected to the internal data and address bus on the module. The FPGA can easily be reconfigured to route these lines to the module pins for possible probing and debugging during tests." Pet. 33–34 (quoting Ex. 1004, 91) (emphasis omitted). As argued by Petitioner, a portion of the FPGA is for signal processing, as recited in limitation [1.7], and a portion of the FPGA is for routing signals to module pins. *Id.* (citing Ex. 1002 ¶ 94 (citing Ex. 1045; Ex. 1015)); *id.* at 34 (quoting Dehkordi's disclosure of a "second portion of the gates in the field programmable gate array is configured to operate as a signal distribution matrix for rerouting signals within the instrument controller") (quoting Ex. 1004, 1).

Petitioner sufficiently articulates a rationale to combine Lyke, Dehkordi, Frantz, and Faura. Pet. 12–18. As we discussed above, *supra* Sec. II.E, during the prosecution of the '177 patent, the Examiner relied on Lyke as a primary reference in rejecting a claim containing all but the last two limitations of claim 1. Ex. 1010, 56–57. Also discussed above, like the Examiner, Petitioner relies on Lyke for teaching most of the limitations of claim 1. Indeed, the '177 patent acknowledges that it is based on Lyke, but states that it is an improvement over Lyke. Ex. 1001, 1:21–3:24 (referring to Lyke as U.S. Patent No. 6,148,399). However, as argued by Petitioner (and discussed above), to the extent features of claim 1 are not disclosed in Lyke, they would have been obvious.

Petitioner relies additionally on the combination of Dehkordi with Lyke in arguing that the art teaches or suggests the FPGA, as taught in Lyke, being configured to run independent processes in parallel with the processor,

as recited in limitation [1.5], as well as in arguing limitations [1.7] and [1.8] are taught in the prior art. Pet. 27–29, 33–34; Ex. 1010, 58. Petitioner argues that a skilled artisan would have combined the teachings of Lyke with Dehkordi, which we find persuasive because both Lyke and Dehkordi relate to MCM designs, and although Lyke teaches using an FPGA in its MCM design it does not provide the level of detail included in Dehkordi concerning how an FPGA would have been configured for various MCM functions. *Id.* at 13 (citing Ex. 1002 ¶¶ 50–55; Ex. 1003, 8:55–59, Abstract, Fig. 1; Ex. 1004, 1, Figs. 1, 2). Moreover, we agree with Petitioner that a skilled artisan would have had a reasonable expectation of success in making the combination because Dehkordi describes how to realize its teachings in a system similar to Lyke. *Id.* at 13. Petitioner also provides persuasive evidence that an FPGA would have been especially useful in an MCM design, because contemporaneous prior art, e.g., Avery[25], discussed the "intriguing possibility" for space-targeted MCMs, specifically referencing Lyke (in addition to other MCMs), to include "embedded programmable logic arrays (i.e., field programmable gate arrays, and even a programmable internal switch matrix for signal re-routing," and a skilled artisan therefore would have been motivated to seek Dehkordi's teachings for, e.g., details concerning configuring an FPGA within an MCM (such as Lyke's). *Id.* at 13–14 (citing Ex. 1002 ¶ 53 (citing Ex. 1003, 5:6–16, 8:27–40); *id.* at ¶ 54 Ex. 1015, 8; Ex. 1042, 1; Ex. 1004, 3; Ex. 1024, Abstract, 3:7–16).

[25] Avery, Keith et al., "Recent Progress in Space-Based Micro-Controller Design," 16th DASC. AIAA/IEEE Digital Avionics Systems Conference, Reflections to the Future, Proceedings, 1997, pp. 2.1–2.7 (Ex. 1015, "Avery").

Petitioner also argues persuasively that the result would have been the combination of prior art elements according to known techniques to a known system to yield a predictable result, because it would have been using an FPGA for signal processing and other tasks in connection with a processor in a known MCM system to yield the above described benefits. *Id.* at 14–15. Accordingly, we are persuaded that a skilled artisan would have been expected to succeed. *Id.* at 15 (citing Ex. 1002 ¶ 55 (citing Ex. 1025, 2:27–31)).

For the embedded memory recited in limitation [1.3], Petitioner relies on Frantz combined with Lyke (or Lyke-Dehkordi). Pet. 24–26; Ex. 1010, 57–58. Petitioner argues that the skilled artisan would have been motivated in view of Frantz to add an embedded memory for storing an initialization program as recited in limitation [1.3], which we find persuasive because Lyke teaches a DSP and Franz teaches that an on-chip ROM (e.g., an embedded memory) would have been used to store initialization programs for a DSP. Pet. 24–26. Petitioner further bolsters its argument by showing the additional benefits of storing an initialization program in an embedded memory, and showing that its use would have been routine and merely the application of known techniques to a known system to yield the predictable result of allowing a processor to start up without having to first retrieve a program from a separate non-volatile memory, among other benefits. *Id.* 15–17.

For limitation [1.6], Petitioner relies on Lyke, as did the Examiner, but Petitioner provides an alternative argument that Faura in combination with Lyke (or Dehkordi-Lyke-Frantz) additionally teaches this limitation. Pet. 29–31; Ex. 1010, 57. Petitioner argues that a skilled artisan would have

combined Lyke (or Lyke-Dehkordi-Frantz) with Faura to add a plurality of analog-to-digital converters (ADCs) for receiving a plurality of analog inputs as recited in limitation [1.6], which we find persuasive because Lyke already teaches use of a collection of ADCs as recited in the claim, and that Faura provides more specific details of implementing ADCs for receiving a plurality of analog inputs. Pet. 29–31. We are persuaded by Petitioner's arguments because Lyke teaches a system that performs a variety of over 15 general applications, each with varying converter resolution requirements, such that a skilled artisan would have sought out Faura's teachings of an especially flexible ADC block because it would have been capable of providing multiple bit depth converters to suit the variety of applications of Lyke's MCM. *Id.* at 18. Moreover, we are persuaded that a skilled artisan would have had a reasonable expectation of success, because commercially available products at the time already included multiple independent converters and a skilled artisan would have been able to combine the teachings of the art according to known methods to yield the predictable result of flexibility of the ADC. *Id.* at 18.

Patent Owner does not provide arguments undermining Petitioner's showing regarding motivation to combine the references, except as discussed in Sec. II.F.1–2.

For the foregoing reasons, we determine that Petitioner has demonstrated by a preponderance of the evidence that claim 1 is unpatentable under § 103 as obvious over the combination of Lyke, Dehkordi, Frantz, and Faura.

G. Obviousness of Claims 5, 10, 11, 13, 14, 15, 17, 19, 20

Apart from those issues discussed above with respect to independent claim 1, Patent Owner does not separately address Petitioner's evidence or arguments directed to claims 5, 10, 11, 13, 14, 15, 17, 19, 20, except for arguments made with regard to Steele. *See* PO Resp. 52–56.

Claims 5, 10, 11, 13, 14, 15, 17, 19, and 20 recite a power converter. By way of example, claim 5 recites:

> The instrument controller of claim 1, wherein a third portion of the gates in the field programmable gate array is configured to operate as an internal embedded power converter capable of receiving an input voltage level and generating each operating and reference voltage needed within the instrument controller.

Ex. 1001, 10:38–43.

As argued by Petitioner, Lyke discloses an embedded power converter as recited in claim 5, except that it does not disclose explicitly that it is the FPGA that is configured to perform such conversion. Pet. 48 (citing Ex. 1003, 5:54–6:19). Specifically, Lyke discloses "an internal embedded power converter 10 that performs . . . generation of each operating and reference voltage needed within the AIC module." Ex. 1003, 5:54–56. Petitioner argues that configuring an FPGA to perform power conversion in the manner claimed was well known, as evinced by Steele. Pet. 48–49.

Steele discloses an FPGA, comprised of eight configurable function blocks (CFBs), wherein the FPGA is powered by a 5 V supply voltage on the V_{CC} input pin, and can be configured to output a voltage on the CFBs' eight V_{CCO} output pins. Ex. 1009, 4:7–43. Specifically, the FPGA can be configured so that different CFBs output different voltages on their respective V_{CCO} pins. *Id.* at 2:35–47, 2:54–65, 4:7–43. In an exemplary

embodiment, either 3 V or 5 V is output on the various V_{CCO} pins. *Id.* at 2:66–67.

Patent Owner attempts to distinguish the claimed "power converter" from Steele, arguing that power conversion requires changing one voltage level to another, and that the output voltages on Steele's V_{CCO} pins are merely a distribution of the same voltage level across several pins. PO Resp. 53–54. Patent Owner's argument is premised on its proposed construction of "power converter" as "a converter which changes voltage level to another voltage level." *Id.* at 29.

Patent Owner's arguments do not undermine Petitioner's showing. As discussed above with regard to claim construction, *supra* Sec. II.B.3, we reject Patent Owner's construction of "power converter." As we discussed, the claims provide sufficient description of "power converter," such that further construction is not required. *Id.* The claims do not recite any requirement of changing one voltage level to another. Rather, the claims require only that the power converter be capable of receiving an input voltage level and generating each operating and reference voltage needed within the instrument controller. *See, e.g.*, Ex. 1001, 10:40–43 (claim 5). Steele satisfies this claimed requirement. As discussed above, Steele discloses a 5 V supply voltage on the V_{CC} input pin (i.e., receives an input voltage level), and can be configured to output a voltage on the CFBs' eight V_{CCO} output pins (i.e., generates each operating voltage and reference voltage needed within the instrument controller). Ex. 1009, 4:7–43. Specifically, the FPGA can be configured so that different CFBs output different voltages on their respective V_{CCO} pins. *Id.* at 2:35–47, 2:54–65, 4:7–43.

We note Steele's teachings are consistent with the power converter disclosed in the '177 patent, which describes employing "subsets of gates in the FPGA" to "provide power conversion," and provides a functional block diagram illustrating the power conversion concept. Ex. 1001, 8:45–9:17, Fig. 3. The power converter in a first stage converts an unregulated supply voltage (from a source external to the FPGA) to a fixed, regulated supply voltage VREG. *Id.* at 8:64–66, Fig. 3. The power converter in the second stage converts VREG to one of the following: internal digital power, internal analog power, or one of various reference voltages. *Id.* at 9:1–4, Fig. 3. The blocks illustrated in Figure 3 do not show any internal details such as circuitry or logic, and the '177 patent does not provide details as to how either power conversion or "generating" voltages is accomplished, other than to state, "[m]any topologies known in the art of power convertor design may be applied, whether switching based, linear based, or a combination of the two." *Id.* at 9:4–6. Based on the '177 disclosure, we discern no reason why power conversion (including "generating" voltages) should preclude Steele's teachings, in which the FPGA is configured to, e.g., receive a 5 V input at V_{CCO} and output either, e.g., 3V or 5V at various V_{CCO} pins.

Having reviewed and considered the evidence and arguments presented by the parties, based on the present record we determine Petitioner has demonstrated, by a preponderance of the evidence, that claims 5, 10, 11, 13, 14, 15, 17, 19, and 20 are unpatentable under § 103.

H. Obviousness of Claims 2, 3, 4, 12, and 18

As to claims 2, 3, 4, 12, and 18, as well as the rationale supporting obviousness, we have reviewed the arguments and evidence in the current record, and determine that Petitioner has made a sufficient showing.

Pet. 34–43, 62–63. Patent Owner does not separately address Petitioner's evidence or arguments directed to claims 2, 3, 4, 12, and 18, apart from those issues discussed above with respect to independent claim 1 and independent claims 10 and 15. *See* PO Resp. 29 (arguing claims 2–5 and 10–12 are patentable for at least the same reasons as claim 1); *see also id.* at 56–57 (arguing claims 2 and 3 are patentable for at least the same reasons as claim 1, and claims 12 and 18 are patentable for at least the same reasons as claims 10 and 15).

Having reviewed and considered the evidence and arguments presented by the parties, we determine that Petitioner has demonstrated by a preponderance of the evidence that claims 2, 3, 12, and 18 are unpatentable under § 103.

III. CONCLUSION[26]

Petitioner has shown that by a preponderance of the evidence that the challenged claims are unpatentable as summarized below:

Claim(s)	35 U.S.C. §	Reference(s)/Basis	Claim(s) Shown Unpatentable	Claim(s) Not Shown Unpatentable
1, 4	103(a)	Lyke, Dehkordi, Frantz, Faura	1, 4	

[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), (b)(2).

Claim(s)	35 U.S.C. §	Reference(s)/Basis	Claim(s) Shown Unpatentable	Claim(s) Not Shown Unpatentable
2	103(a)	Lyke, Dehkordi, Frantz, Faura, Krasner	2	
3	103(a)	Lyke, Dehkordi, Frantz, Faura, Christian	3	
5, 10, 11, 13, 14, 15, 17, 19, 20	103(a)	Lyke, Dehkordi, Frantz, Faura, Steele	5, 10, 11, 13, 14, 15, 17, 19, 20	
12, 18	103(a)	Lyke, Dehkordi, Frantz, Faura, Steele, Christian	12, 18	
Overall Outcome			**1–5, 10–15, 17–20**	

IV. ORDER

In consideration of the foregoing, it is

ORDERED that Petitioner has shown that challenged claims 1–5, 10–15, and 17–20 of the '177 patent are unpatentable; and

FURTHER ORDERED that, because this is a Final Written 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:

David Hoffman
Kenneth Darby
Craig Deutsch
FISH & RICHARDSON P.C.
hoffman@fr.com
kdarby@fr.com
deutsch@fr.com

For PATENT OWNER:

Gerald J. Flattmann, Jr.
Andrew J. Cochran
John C. Stellabotte
CAHILL GORDON & REINDEL LLP
gflattmann@cahill.com
acochran@cahill.com
jstellabotte@proskauer.com