

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.

Case IPR2023-00195
U.S. Patent No. 6,938,177

**DECLARATION OF JOHN PECK IN SUPPORT OF
PATENT OWNER'S RESPONSE TO PETITION FOR *INTER*
PARTES REVIEW OF U.S. PATENT NO. 6,938,177**

TABLE OF CONTENTS

I.	INTRODUCTION	4
II.	QUALIFICATIONS	7
III.	BASIC LEGAL PRINCIPLES	10
	Person of Ordinary Skill in the Art.....	10
	Patent Validity	11
	Non-Obviousness.....	12
IV.	TECHNOLOGY BACKGROUND.....	14
V.	OVERVIEW OF CERTAIN PRIOR ART	18
	<i>Dehkordi 1</i> (EX. 1004).....	18
	<i>Dehkordi 2</i> (“D2”) (EX. 2019).....	20
	<i>Steele</i> (EX. 2009).....	22
VI.	Claim Construction.....	24
VII.	Patentability Analysis	26
	Ground 1 - Claims 1 and 4 Are Patentable Over the Combination of <i>Lyke</i> , <i>Dehkordi</i> , <i>Frantz</i> , and <i>Faura</i>	26

Claim 1.....	26
Lyke (EX. 1003).....	27
D1 (EX. 1004)	31
D2 (EX. 2019) in view of Kostarnov (EX. 1036)	33
D1 Theoretical Testing	36
JTAG Testing.....	38
TAP Testing.....	40
<i>Halverson (Ex.1025), Marchal (EX.1035), Lyke and D1 All Do Not Show Independent Operation Which is Not Common Knowledge</i>	<i>43</i>
Kostarnov (EX. 1036) Is Not Used as the Basis of Any Ground.....	44
Ground 4—Based on Lyke, Dehkordi, Franz, Faura, and Steele.....	45
Grounds 2, 3, and 5.....	48
VI. CONCLUSION.....	49

I, **John Peck**, do hereby declare as follows:

I. INTRODUCTION

1. I have been retained by Sentient Sensors LLC (the “Patent Owner” or “Sentient”), through their attorneys Ellenoff Grossman & Schole, as an expert witness on behalf of Patent Owner for this case involving the petition for *Inter Partes* Review (“IPR”) of U.S. Patent No. 8,938,177 (the “’177 patent,” EX. 1001) filed by the Petitioner Xilinx et al. (the “Petitioner” or “Xilinx”).

2. Among other things, I have been asked to render an opinion as to whether claims 1-5, 10-15, and 17-20 of the ’177 patent are patentable over the prior art references listed in the Petition. Additionally, I have been asked to provide brief relevant background information relative to certain technical issues as set forth herein.

3. I am being compensated as an independent consultant at a rate of \$365 per hour for my work on this matter. I have received no additional compensation for my work on this matter, and my compensation depends in no way on my opinions expressed in this declaration, any testimony that I may give, or the outcome of this matter. In addition, I will be reimbursed for reasonable expenses incurred in connection with my work on this matter.

4. In preparing my Declaration, I have relied upon my knowledge, skill, experience, training, and education in my field of expertise.

5. I have also reviewed and considered the '177 patent, its prosecution history, and prior art references below:

- U.S. Patent No. 6,938,177 (“’177 patent”) (EX1001)
- U.S. Patent No. 6,148,399 to James C. Lyke (“Lyke”) (EX1003)
- P. Dehkordi, T. Powell and D. Bouldin, *Development of a DSP/MCM subsystem assessing low-volume, low-cost MCM prototyping for universities*, Proceedings 1996 IEEE Multi-Chip Module Conference (Cat. No. 96CH35893), 1996, pp. 89-94 (“Dehkordi”) (EX1004)
- J. Faura, C. Horton, P. van Duong, J. Madrenas, M. A. Aguirre and J. M. Inserer, *A novel mixed signal programmable device with on-chip microprocessor*, Proceedings of CICC 97 - Custom Integrated Circuits Conference, 1997, pp. 103-106 (“Faura”) (EX1006)
- U.S. Patent No. 5,663,734 to Norman F. Krasner (“Krasner”) (EX1007)
- U.S. Patent No. 4,279,020 to Ronald W. Christian et al (“Christian”) (EX1008)
- U.S. Patent No. 5,675,824 to Randy C. Steele (“Steele”) (EX1009)
- File Wrapper of the ’177 Patent (EX1010)
- Claim Construction Order in *Sentient Sensors v. Cypress Corp.*, 19-1868 (MN), (Jan. 28, 2021, D. Delaware) (D.I. 89) (EX1012)
- Data Sheet for Motorola 96002 Digital Signal Processor (EX1013)
- Data Sheet for Xilinx XC4000E and XC4000X Series (including Xilinx XC4010) (EX1014)
- U.S. Patent No. 5,574,930 to Richard P. Halverson, Jr. (“Halverson”) (EX1025)
- Marchal, Pierre, *Field-programmable gate arrays*, Communications of the ACM, Vol. 42, No. 4 (1999), p. 57-59 (“Marchal”) (EX1035)
- I. Kostarnov, S. Morley, J. Osmany, C. Solomon, *A reconfigurable approach to low cost media processing*. In: Luk, W., Cheung, P.Y.K., Glesner, M. (eds) *Field-Programmable Logic and Applications*. FPL 1997. Lecture Notes in Computer Science, vol 1304. Springer, Berlin, Heidelberg (“Kostarnov”) (EX1036).
- U.S. Patent No 5,892,962 to Jocelyn Cloutier (“Cloutier”) (EX1040)

- U.S. Patent Publication No. to 2002/0129191 to Behram DaCosta (“DaCosta”) (EX1041)
- Enhancement of MCM Testability Using Embedded Reconfigurable FPGA J. York, T. Powell P. Dehkordi, and D. Bouldin, Proceedings 1997 IEEE Multi-Chip Module Conference (Cat. No. 0-7803-4275), 1997, pp. 165-173 (“Dehkordi 2”) (EX. 2019)
- Ch. 11 in [A New Kind of Science](#). Champaign, IL: Wolfram Media, pp. 637- 714, 2002. (EX 2021)
- Sentient’s District Court Identification of claim constructions (EX. 2022)
- Xilinx’s District Court Identification of claim constructions (EX 2023)
- Definition of computation (EX, 2024)

<https://mathworld.wolfram.com/Computation.html>

6. This Declaration sets forth the opinions I have formed in this matter and the bases for those opinions through my independent evaluation and analysis. My opinions are based on the information available to me as of the date that I am submitting this Declaration. If additional information becomes available to me either by production by the parties or third parties, or otherwise, I may, if permitted to do so, offer additional opinions. I may also, if requested and permitted to do so, provide further opinions to rebut any testimony, reports or opinions offered by Xilinx’s witnesses (expert or otherwise). I may present demonstrative or illustrative exhibits at trial to explain my opinions to the Board.

7. It is my opinion that claims 1-5, 10-15 and 17-20 of the ’177 patent are patentable and not obvious over the prior art relied on in the Petition alone or in combination, and/or in view of the knowledge of an ordinary artisan.

II. QUALIFICATIONS

7. I expect to testify concerning my qualifications, background, and experience relevant to the issues in this investigation. Attached as **Appendix A** is a copy of my current curriculum vitae, which describes my academic and professional background and qualifications. I briefly address below some of those background and qualifications.

8. I am a principal engineer and co-founder of Bay2Sierra Silicon Services Inc., which provides engineering consulting services to high-tech companies. I provide services including software development, hardware system architecture and design, and hardware design and implementation. I have designed various microprocessors, including network processors, network switch chipsets, processor chipsets, systems on chip (SoC), custom application specific integrated circuits (ASICs), wafer scale ASICs, and field programmable gate arrays (FPGAs). My clients have included large computer and technology companies as well as smaller companies and start-ups. During the period January 2017 to June 2017, I worked as a consultant to Xilinx Inc, that included work on identifying low-power optimizations for FPGA devices using a hand placement methodology applied to cells from standard libraries.

9. I earned a Bachelor of Science Degree (1992) in Computer Engineering from Carnegie Mellon University. I also earned a Master of Science Degree (1995) in Computer Science from the University of California Los Angeles.

I received the David Tuma undergraduate Laboratory award in 1992 for my design and implementation of an asynchronous transfer mode (ATM) switch prototype.

10. I have been a professional software developer and hardware design engineer for over 25 years. I have designed and developed a variety of computer hardware and software products. Such hardware products include network processors, network switches, compression and decompression hardware, smart power meters, memory interfaces, programmable logic devices, several systems on chip (SoC), FPGA-based systems for product and prototyping, error detection and correction for magnetic memories, hardware emulators, wafer scale machine learning systems, and FPGA designs and various other component level designs. The software products include real-time control systems, design automation flows, cycle accurate simulators, OpenSSL cryptography engines, and video games.

11. I have released some of my work to the public. For example, I published my OpenSSL cryptography engine as an open source project on GitHub.¹

12. My clients include Advanced Micro Devices, Redback Networks, Xilinx, NetApp, Spansion, Silver Spring Networks, MIPS Technologies, Silicon Image, MoSys, Inston, Spin Memory, Tabula, Cerebras Systems, AXBIO, and Activision.

13. My work with FPGA-based systems includes a NVRAM card used by

¹ <https://github.com/AtmelCSO/cryptoauth-openssl-engine>.

NetApp Inc. in their data storage products, a prototype memory controller for spin torque transfer magnetic memory devices at Spin Memory, Inc., and two prototype interfaces for a custom ASIC used for processing biological fluid samples at AXBIO. I have also worked on the design of FPGA devices at Tabula, Inc. and at Xilinx Inc.

14. I have worked on two generations of SoCs for SilverSpring Networks, Inc. that are used in smart electric meters.

15. I have extensive experience with the JTAG protocol, in my work as a design verification engineer on three generations of the wafer-scale ASIC that was inducted into the computer history museum in 2022 as the world's largest ASIC (2.6 trillion transistors). This wafer-scale ASIC is available in the CS-2 system from Cerebras Systems, Inc.

16. My work at Activision included development of video games for the Super Nintendo platform including the published title, "Pitfall the Mayan Adventure."

17. I am named inventor of ten patents granted by the United States Patent and Trademark Office. The patented inventions concern, for example, memory controller performance improvements, virtual to physical address translation, and serial peripheral interfaces.

18. I am a member of several professional societies and organizations. For example, I am a senior member of the Institute of Electrical and Electronics

Engineers (IEEE), a member of the IEEE Consultants' Network of Silicon Valley (IEEE-CNSV), and a senior member of the Association for Computing Machinery (ACM).

19. I have consulted on matters involving various intellectual property disputes, including instances of alleged misappropriation and infringement. My work in this capacity has included, among other things, reviewing and analyzing software source code and hardware source code, reviewing and analyzing patents, reverse engineering hardware and software, and writing expert reports.

20. By virtue of the above experience, I have gained a detailed understanding of the technology that is at issue in this case. In addition, my experience with automotive infotainment systems and technologies is directly relevant to the subject matter of the '177 patent.

III. BASIC LEGAL PRINCIPLES

21. I have been provided certain legal principles and standards by counsel, which are set forth below, relating to the issues in this report. In conducting my analysis set forth in this report, I have been guided by these principles.

A. Person of Ordinary Skill in the Art

22. I understand that "a person of ordinary skill in the art" ("POSA") is a hypothetical person who is presumed to be aware of pertinent art and knowledge in the art, thinks along the lines of conventional wisdom in the art, and is a person of

ordinary creativity, among other things. I also understand that such a person has the normal skills and knowledge of a person in a given technical field at the time of the invention. I further understand that certain factors may be considered in determining the level of ordinary skill in the art, including: (1) the education level of the inventor; (2) the types of problems encountered in the art; (3) the prior art solutions to those problems; (4) the rapidity with which innovations are made; (5) the sophistication of the technology; and (6) the education level of active workers in the field.

I have examined the issue of the relevant field of art for the '177 patent and the level of skill in the art at the time of the invention consistent with the applicable legal standards. In my opinion, an ordinary artisan would have been a person having, as of the filing of the '177 of December 19, 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 vehicle sensor systems or related technologies. Additional education may substitute for lesser work experience and vice-versa.

I consider myself to be a person of at least ordinary skill in the art with respect to the subject matter of the '177 patent.

B. Patent Validity

23. I understand that each claim of an issued patent is entitled to a statutory presumption of validity. I also understand that the party challenging a patent's

validity in the IPR must prove unpatentability by preponderance of evidence to overcome the statutory presumption of validity. I understand that preponderance is based on the more convincing evidence and its probable truth or accuracy, and not on the amount of evidence.

C. Non-Obviousness

24. I understand that a patent claim is invalid if the subject matter of the claim as a whole would have been obvious to a person of ordinary skill in the art of the claimed subject matter as of the time of the invention at issue. I understand that the following factors must be evaluated to determine whether the claimed subject matter is obvious:

- (1) the scope and content of the prior art;
- (2) the difference or differences, if any, between each claim of the patent and the prior art;
- (3) the level of ordinary skill in the art at the time of invention; and
- (4) objective indicia of non-obviousness.

25. I understand that obviousness may be shown by considering more than one prior art reference. The reasons for combining references stem (explicitly or implicitly) from, for example: (a) the prior art references themselves; (b) the prior art as a whole; (c) the knowledge and creativity of those of ordinary skill in the art; (d) the nature of the problem to be solved; (e) the demands in the design community and/or the market place; (f) the simple and predictable substitution of one known

element for another in accordance with their known functions; (g) the application of a known technique or method; (h) the obviousness of trying the combination; and/or (i) the general needs and problems in the field.

26. I understand that the scope and content of the prior art should include references reasonably related to the claimed invention of that patent. A reference is reasonably related if it is in the same field as the claimed invention or is from another field that a person of ordinary skill in the art would look to so as to solve the problem. Prior art is not limited to patents and published materials, but also includes the general knowledge that would have been available to one of ordinary skill in the field of the invention.

27. I understand that so-called objective indicia of non-obviousness, also known as “secondary considerations,” are to be considered, if present, when assessing obviousness. These include commercial success, long-felt but unresolved need, failure of others to solve the problem that the inventor solved, unexpected results, copying of the invention by others, and industry recognition or expressions of disbelief by experts in the field of the claimed invention. I also understand that a nexus, i.e., a tie, must exist between objective indicia of non-obviousness and the novel aspects of the claimed subject matter. I understand that a nexus can be found to exist between the objective indicia and the claimed subject matter if (1) the objective indicia is tied to a specific product that is coextensive with the claimed subject matter; or (2) the objective indicia is a direct result of the unique

characteristics of the claimed subject matter.

28. I understand that the use of hindsight must be avoided when considering whether the alleged invention of the patent claims would have been obvious to the person of ordinary skill in the art in light of the prior art. I understand that in determining obviousness one should not lose sight of the fact that the person of ordinary skill is also a person of ordinary creativity, not an automaton or savant. When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill may have a reason to pursue the known options within his or her technical grasp. I understand that, if a technique has been used to improve one device, and a person of ordinary skill may recognize that it would improve similar devices in the same way, using the technique is obvious unless its application is beyond his or her skill.

IV. TECHNOLOGY BACKGROUND

29. The '177 patent relates to circuit designs that involve programmable logic and processors that may operate independently and in parallel that provide robust computing capabilities that are useful for test and measurement and instrument control among other things.

30. I have been asked to briefly summarize the background of the prior art from the standpoint of an ordinary artisan.

31. A microprocessor is a computer processor in which the data processing

logic and control is included on a single integrated circuit (IC), or a small number of ICs. The microprocessor typically contains the arithmetic, logic, and control circuitry required to perform the functions of a computer's central processing unit (CPU). The IC is capable of interpreting and executing program instructions and performing arithmetic operations. Microprocessors are typically a multipurpose, clock-driven, register-based, digitally-integrated circuit that accepts binary data as input, processes it according to instructions stored in its memory, and provides results (also in binary form) as output. Microprocessors typically contain both combinational logic and sequential digital logic, and operate on numbers and symbols represented in the binary number system.

32. The integration of a whole CPU onto a single or a few integrated circuits using Very-Large-Scale Integration (VLSI) greatly reduced the cost of processing power. Integrated circuit processors are produced in large numbers by highly automated metal-oxide-semiconductor (MOS) fabrication processes, resulting in a relatively low unit price. Single-chip processors increase reliability because there are fewer electrical connections that could fail. As microprocessor designs improve, the cost of manufacturing a chip (with smaller components built on a semiconductor chip the same size. The first commercially available microprocessor was the Intel 4004, designed by Federico Faggin and introduced in 1971.

33. A field programmable gate array (FPGA) is a digital circuit that

contains logic elements and routing resources. Both are controlled by configuration memory programmed by the user. Logic elements range in complexity from simple combinatorial logic functions up to complete embedded processors. Logic elements and routing are arrayed in a grid of tiles. The structure of an FPGA is homogenous. Each tile contains one of a small variety of VLSI circuits dedicated to logic or routing.

34. Computer systems normally contain data bus-based architectures in which several different devices compete for control of the bus. Such devices include signal processors, computational circuitry such as FPGAs, memories and input/output controllers and the like. An arbitration scheme is incorporated into the design of the system to determine which of these devices can access and control the bus at a given time to prevent conflicts. Such schemes are adopted based on the desired operational goals. For example, some schemes allow high priority devices to have control at the expense of other devices. Other schemes have a more egalitarian approach for bus allocation to devices more equally or on an “as needed” basis. EX. 2018.

35. The most basic arbitration scheme is the so-called FIFO (First-In-First-Out). In this scheme requests to access a memory port interface through a bus are treated strictly in the order in which they arrive. This scheme may not take into account the relative importance of each of the access requests. Thus, any device urgently requiring access to the memory may have to wait while other lower-

priority devices complete their operations.

36. Another commonly used method in processor-based with systems with FPGAs for bus arbitration involves the use of a central arbiter to communicate with all the devices desiring to use the bus. The central arbiter determines which device has the highest bus use priority taking into account how important the device is and how long the device has been requesting the use of the data bus. It then sends out a signal permitting the device to use the bus. This processing system typically fails if the central arbiter ceases to function.

37. Another commonly used arbitration scheme is processor systems that include FPGAs, is “daisy-chaining.” In this method a signal indicating that the bus is available for use is transmitted from device to device in priority order. Any device requiring access “captures” the signal and as a result may control the bus. Low priority devices may have to wait an undesirable length of time until they are granted access to the bus.

38. A two-level internal bus arbitration for the system IO bus, teaches a fourth prior art system in which a fixed priority scheme is used to determine the arbitration. All requests received while a bus is mastered by another device are latched by common logic and will be processed in priority order when the next bus becomes free.

39. A rotating priority scheme offers bus access to each device in strict rotation. This reduces bus access requests being delayed for too long, but it fails to

take into account the need for some devices to access the bus prior to others.

40. Importantly, none of these arbitration schemes, which are used in processor systems that may include FPGAs, require the independent or parallel operation of any other device, whether they have logic configured from an external non-volatile memory or not.

V. OVERVIEW OF CERTAIN PRIOR ART

Dehkordi 1 (EX. 1004)

41. Dehkordi 1 (“D1”) describes certain ongoing, in-process development efforts to produce an MCM-based DSP subsystem that includes a Motorola 96002 DSP processor and Xilinx 4010 series FPGA. At the time of its publication, the DSP system was still in fabrication, with its operation and configuration still theoretical and therefore not enabled. EX. 1004 at 3-4 (“testing *and* rework issues will *continue* to be challenging” and “looking forward to *confirming functionality*”). Only a basic block diagram is provided without any operational details and the aspirational MCM subsystem cannot be produced from this document without additional information. As shown in the annotated figure from D1 below, the DSP processor and FPGA are connected through numerous buses, the most notable of which is the *control bus* (highlighted in yellow). A POSA would understand that the designers of D1 intended the control bus to allow the processor to control the FPGA with operational instructions, including providing a clock signal. Thus, for normal operation, the designers selected the FPGA to be *dependent* on and under

The diagram illustrates the MCM-D Substrate architecture. At the top, the MCM-D Substrate provides signals to Port B of the DSP: Quiet Power, IRQs (Mode), Clk, Res, and OnCE Port. Port B is also connected to the DSP's internal logic. The DSP has Port A at the bottom, which connects to a Data Bus, Address Bus, and Control Bus. The Data Bus connects to SRAM and FLASH memory blocks. The Address Bus connects to the same memory blocks and to the Xilinx block. The Control Bus connects to the Xilinx block. The Xilinx block contains a Power-up Circuit, Memory CS, Rd, Wr, BIST, ADC Interface, DSP Functions, Address Buffer, Bus Arbit., and Data Buffer. The ADC Interface is connected to an external ADC block. The Xilinx block is connected to Program & JTAG and General Purpose I/O blocks.

42. This is consistent with D1’s technical description that explains how the MCM subsystem design is intended to specifically exploit the multiprocessing capability of the DSP processor. EX. 1004 at 1, “Design Flow” section. This

technical description means that the FPGA is intended to operate in a support role under the control of the DSP processor to maximize the DSP processor's multiprocessing functionality by, *e.g.*, preprocessing incoming data as directed by the DSP processor to relieve that burden from the DSP processor. No other specialized connection or functionality is shown.

43. D1 further theorizes a special testing mode of operation might employ the well-known industry standard JTAG and TAP testing methodologies used for testing the operation of the DSP subsystem. *See* EX. 1003 at 3, “*Cost and Testing Issues*” section. Notably, however, JTAG and TAP testing operations typically require an external JTAG computer or other controller that controls both the DSP processor and the FPGA during testing. Accordingly, because both the FPGA and DSP processor are under common control during testing, they do not operate any independent processes (i.e., they are both under the control of another common device). Xilinx's own JTAG/TAP documentation used for the 4010 series FPGA explains and confirms this. EXs. 2004-2005.

Dehkordi 2 (“D2”) (EX. 2019)

44. FIG. 1 of D2 (below) is simplified functional representation of FIG. 4, which is a more complete version of the MCM subsystem in FIG.1 of D1 (EX. 1004).

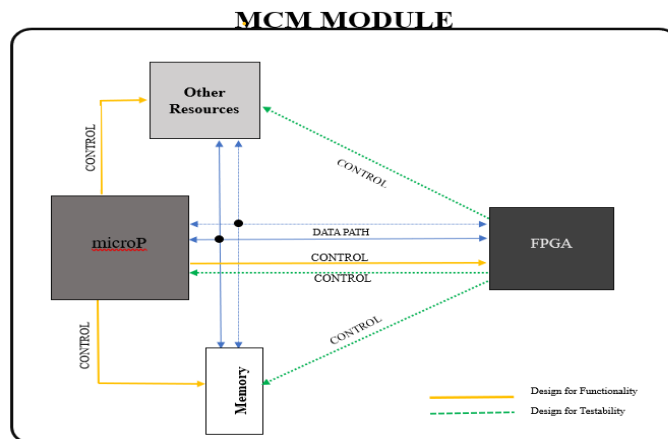


Figure 1.: Designing for Testability for a Typical MCM.

45. D2, a highly relevant reference is a follow up to D1 published a year later. D2 describes a closer-to-complete version of the MCM subsystem by the same authors regarding the same MCM subsystem and the same testing features described in D1, and that clarifies the operation of the MCM subsystem and its testing. D2, entitled “*Enhancement of MCM Testability Using Embedded Reconfigurable FPGA*,” confirms that D1 does not show independent processes. As annotated FIG. 1 of D2 above shows, in normal operation (the “design for functionality” configuration - highlighted in orange), the processor controls the entire subsystem including the FPGA, the memory, and the other resources necessary to operate the system. Alternatively, in the special testing mode (the “design for testability” configuration - highlighted in green) it is the FPGA that controls the entire subsystem including the processor, the memory and the other resources. There is no discussion of bus arbitration in D2, which appears to have been abandoned as the design progressed. However, D2 does employ JTAG

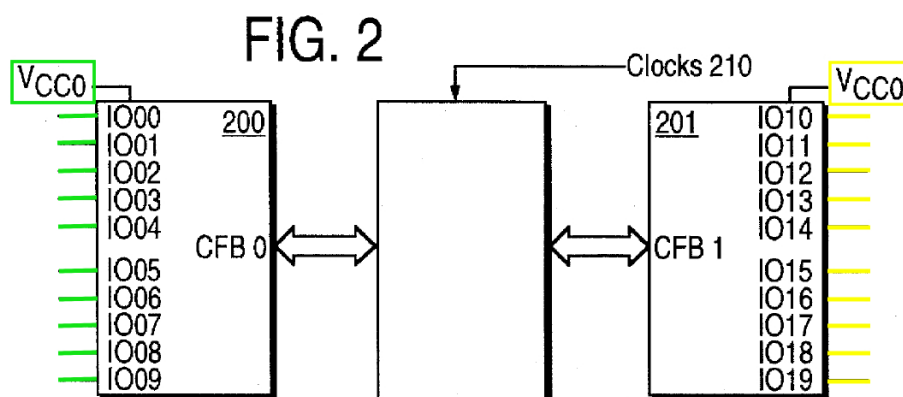
boundary scans (IEEE 1149.1). D2 at 167. There are no other modes of operation.

Thus, at no point does the D1-D2 MCM subsystem FPGA operate either parallel or independent processes. Either the processor or FPGA are in control.

46. Importantly, D1 is a system that uses an external RAM-based configuration where the FPGA does not run independently or in parallel with the processor.

Steele (EX. 2009)

47. *Steele* describes a programmable logic device configurable to operate as a voltage replicator that merely distributes an applied input voltage for output at other pins. Simply stated, it operates by accepting a fixed input voltage from an external source at the VCCO pin of a Configurable Logic Block (CFB) and replicates that fixed voltage at the input/output (I/O) pins for use in external devices. EX. 1009, 4:28-58. This is explained in detail at 4:28-43 of *Steele* and shown in *Steele*'s FIG. 2 below (annotated to show the 5V level (CFB 0) (green) and the 3V level (CFB 1) (yellow)).



48. Notably, *Steele* does not generate the applied VCCO voltage levels itself, but rather merely uses them. The voltage generation is done elsewhere on another external device and those voltages are applied to the VCCO input from that external source for reproduction by *Steele*'s CFB. *Steele* does not convert power because it does not change one voltage level to another as the claims require under Patent Owner's claim construction and as described in the '177 specification. Instead, *Steele* merely distributes the same voltage level applied VCCO at the output I/O pins. For example, IO00-IO09 have 5V outputs whereas IO10-IO19 have 3V outputs. See EX. 1009, 2:22-26; 4:28-43. There is no change or conversion (*i.e.*, according to the plain meaning of changing one thing to another). Merely distributing the same voltage multiple times is not converting power under any reasonable definition.

VI. Claim Construction

49. In order to properly evaluate these claims, I understand that the terms of the claims must first be construed. In my opinion, only three terms of the '177 Patent require specific construction. I provide the definitions I have used for these terms in connection with my analysis below. I understand that the remaining claim terms all have a plain and ordinary meaning that an ordinary artisan would understand in light of the '177 specification.

50. In my opinion, the proper construction of “process” is “*a computational process programmed into the FPGA gates.*” This definition is clear from the '177 specification and one of ordinary skill in the art would understand that the claimed process must be a computational process to achieve the goals of the invention. 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. 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. Routing operations are computations. The '177 FIG. 2 and its associated description explains that the initial conditions are the input values present on each of X1, X2, X3. The definite set of rules are the AND gates and control values presented on

the inputs to the AND gates. The outputs Y1, Y2, Y3 are output values from the cascaded arrangement of gates with corresponding control values that comprises the definite set of rules applied to the input values X1, X2, X3.

51. In my opinion this definition is consistent with what “computation” means in the field of computing systems, which is a “computation is an operation that begins with some initial conditions and gives an output which follows from a definite set of rules.” The most common example are computations performed by computers, in which the fixed set of rules may be the functions provided by a particular programming language. The field of computer science studies the nature of computation and its uses, among other things. A set of rules used to carry out a computation is known as an algorithm. EXs. 2021; 2024.

52. In my opinion, the proper construction of “independent processes” is “*without computational assistance from another*” which is consistent with the specification and Patent’s Owner’s definition in a prior litigation involving the ’177 patent. See, for example, EX. 1001 2:9-18; 2:48-55; EX. 1012; In my opinion, 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 correctly opines - the specification supports both. Institution Decision at 16.

53. In my opinion, the proper construction of “power converter” is “a *converter which changes a voltage level to another voltage level*” which is

consistent with Patent Owner's litigation definition (EX. 1012, 2023) and the specification. See, for example, EX. 1001 8:64-67.

VII. Patentability Analysis

Ground 1 - Claims 1 and 4 Are Patentable Over the Combination of Lyke, Dehkordi, Frantz, and Faura

Claim 1

54. The combination in Ground 1 of *Lyke, Dehkordi, Franz, and Faura*² fails to teach or suggest, among other deficiencies, a “*gate array coupled to the processor and configured to run independent processes in parallel with the processor*” as specified in claim 1 (element 1.5) which was a point of novelty described in the '177 Patent. Claims 2 -5 and 10-12 are patentable for at least the same reasons as claim 1.

55. Xilinx offers only two references for the proposition that element 1.5 would have been obvious to a POSA at the time of the inventions described and claimed in the '177 Patent: *Lyke* EX. 1003 (see Petition at 27-28,) and *Dehkordi* EX. 1004 (see Petition at 28-29). Institution Decision a 26-27. But neither *Lyke* nor *Dehkordi* teaches or suggests this novel feature.

² I see no other combinations enumerated in Ground 1. Nor do I see a combination that specifically relies on the general knowledge of an ordinary artisan.

Lyke (EX. 1003)

56. As noted by Petitioners, Patent Owner was indeed very familiar with technology discussed in the '399 Patent. *See* Petition at 4-5. Patent Owner made no attempt to hide or conceal that fact. The '177 Patent explains the function of the '399 Patent's technology in the section titled DESCRIPTION OF THE PRIOR ART. *See* '177 Patent at 1:5-2:8.

57. This attribution to the '399 Patent was based on the U.S. Air Force's desire to advance the '399 Patent technology and specifically commissioned Patent Owner through the Space Directorate CRADA (EX. 2001) to improve on the systems disclosed in the '399 Patent.

58. During the development period that began in 1998, numerous technical limitations of the systems of the '399 Patent were noted. For example, as specifically explained by the '177 Patent, the '399 Patent—and the AIC it describes—suffers from several deficiencies, such as its *inability to run parallel and independent processes*, and there was a need for a circuit design capable of executing multiple parallel processes in an independent fashion. *See* '177 Patent at 2:9-18:

However, there are several limitations to the design in
10 U.S. Pat. No. 6,148,399. 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
15 processes are not possible. Accordingly, what is needed is an
MCM design capable of executing multiple parallel pro-
cesses in an independent fashion.

59. This theme is repeated throughout the '177 Patent's specification, such as at, *e.g.*, Ex. 1001 2:48-55 and 8:18-22. It is also a specific requirement in the claims, such as, for example, independent claims 1 (element 1.5), 6 (not challenged), 10 (element 10.5, and 15. See '177 Patent, *e.g.*, 10:61-64, 11:42-44, 12:44-46.

60. Because a memory with program instructions might be used to program logic in the '399 Patent (*i.e.*, EX. 1003, 8:55-59), it necessarily means the FPGA runs independently in parallel. Moreover, in the AIC described in Lyke the CPU provides the ASIC clock, so as correctly pointed out in the "177 Patent, all of those processes are under control of the CPU and thus dependent on the CPU. EX. 1003 FIG. 12 ALOGCLK.

61. A POSA would understand it is axiomatic that every functional FPGA must be programmed by a memory with program instructions and that such memory was likely external. Otherwise, the programmable logic gates are non-functional (a "blank device"). A set of unspecified "program instructions" does not confer on any given FPGA an unstated ability to run "independent parallel

operations.” In fact, as common sense would dictate, there is no mandatory connection between them and, notably, the Petition does not include any evidence other than conclusory expert testimony to substantiate its claim. For example, Petitioner’s reference, *Marchal* (EX 1035) explains that it “operates *whatever program is contained* in that memory,” with no other implications.

62. I agree there is no discussion *at all* of how the FPGA is used in the ‘399 Patent. Therefore, it teaches little if anything other than its mere potential presence (no functions). In fact, there is no discernable pragmatic use for the FPGA in the system of the ‘399 Patent at all, as explained below.

63. First, to the extent that there is any question whether the ‘399 Patent shows any independent parallel processes, the clear answer is no. As the ‘177 specification correctly notes, the ‘399 Patent runs from a clock controlled by the processor, making computing circuitry dependent on that processor-controlled clock. EX 1001 2:48-50; EX. 1003 FIG 12 and ALOGCLK. Furthermore, neither the resistive or analog ASIC are capable of any “autonomous,” “independent” or “parallel” operation³. For example, the ‘399 Patent explains that the “resistive” ASIC is merely a set of “dumb resistors” such as a custom resistor IC that are passive circuit elements and whose only purpose is to conserve space. EX.1003; 8:49-53.

³ Terms also all specifically searched for by the PTO. EXs. 2007-2014.

A resistor ASIC is used in the AIC design to eliminate over 50 individual resistors. Its primary function is to provide support for the analog ASIC. Embedded passive elements within the interconnection system can also be used to further reduce floor plan congestion within the MCM.

64. FPGA devices of the '399 Patent's era do not supply features that relate to the implementation of resistors and are therefore unsuitable for the application requirements described in *Lyke*. Even if they did, the great cost, size, and power consumption disparity would strongly discourage such use (the FPGA being much larger, expensive and power intensive).

65. The analog ASIC in the '399 patent (EX. 1003 4:59-5:16) is described as implementing analog-to-digital converters (ADCs), digital-to-analog converters (DACs), a band-gap reference, and a proportional to absolute temperature sensor. The DACs and ADCs require processor control. The reference and sensor are simple circuit elements, that are not capable of independent parallel processes. Moreover, FPGAs devices that existed at the time the '399 Patent was filed do not supply any of these sensors or measurement devices as described in the '399 Patent.

66. Additionally, the '399 patent explains that "the use of a separate integrated circuit for analog functions *is important* as it permits better electrical isolation from ICs within the IC containing digital switching circuitry (*e.g.* the processor and memory components). Such an approach permits better control of the analog system electrical environment." EX. 1003, 4:67-5:59 (emphasis

added). A POSA would understand that analog ASIC components are susceptible to electrical noise created by digital switching circuitry, including voltage spikes on power rails caused by inductive effects. Therefore, a POSA would not implement such analog functions on an FPGA of that era even if it were possible. Indeed, the '399 patent's specification teaches away from such an implementation.

67. Accordingly, in my opinion, the '399 Patent's silence on the FPGA coupled with its unsuitability for use as an ASIC confirm the need for the '177 technology. However, one technical constraint is clear, even if an FPGA was added to the '399 system in an unspecified capacity, it would necessarily be governed by the same processor-controlled clock as shown in FIG. 12 of the '399 Patent, (ALOGCLK) and suffer from the deficiencies that inspired the '177 Patent.

68. Therefore, in my opinion, *Lyke*, the intrinsic record, and all the supporting references relied on in Petition demonstrates that the '399 Patent does not teach or suggest a “*gate array coupled to the processor and configured to run independent processes in parallel with the processor*” whether using Patent Owner's definition or any other reasonably appropriate definition. Dr. Shanfield substantiate the assertion that *Lyke* would operate in the way they suggest or how to overcome the technical incompatibilities that would result.

D1 (EX. 1004)

69. D1 is the only other reference offered by in the Petition for allegedly teaching parallel independent operation. D1 describes an MCM-based DSP

subsystem that includes a Motorola 96002 DSP processor and Xilinx 4010 series FPGA shown below:

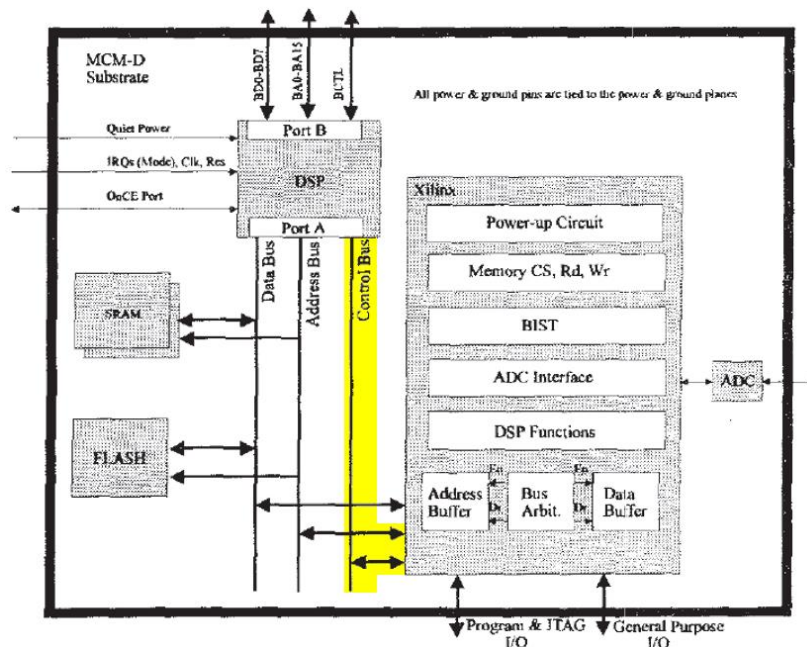


Figure 1. The Block Diagram of the DSP Sub-system.

70. As shown above, the DSP processor and the FPGA are connected through a control bus (highlighted in yellow) used to control the FPGA, with both components relying on the same clock, and which a POSA would understand were synchronized. As such, the FPGA is dependent on the processor, and they are necessarily in a dependent configuration. Consequently, no independent processes are possible during normal operation and neither the Petition nor D1 suggest otherwise.

71. This is consistent with D1's technical description that 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. *Dehkordi* show any specialized connection or functionality that would suggest a different mode of operation. See EX.1004 at 1, FIG. 1 and its associated description in the “*Design Flow*” section.

D2 (EX. 2019) in view of Kostarnov (EX. 1036)

72. D2, a highly relevant reference is a follow up to D1 published a year later, that describes a closer to complete version of the MCM subsystem by the *same authors* regarding the *same MCM subsystem* and the *same testing features* that clarifies certain operations of the MCM subsystem and its testing. FIG. 1 of D2 (EX. 2019) is simplified functional representation of FIG. 4, which is a more complete version of the MCM subsystem in FIG.1 of D1(EX. 1004. As annotated FIG. 1 of D2 shows, in normal operation (the “design for functionality” configuration - highlighted in orange) the processor controls the entire subsystem including the FPGA, the memory and the other resources necessary to operate the system. Alternatively, in the special testing mode (the “design for testability” configuration - highlighted in green) it is the FPGA that controls the entire subsystem including the processor, the memory and the other resources. There is *no discussion* of bus arbitration in D2, which appears to have been abandoned as the MCM design changed as compared to D1. However, D2 employs JTAG

boundary scans (IEEE 1149.1). D2 at 167. There are no other modes of operation.

73. Thus, at no point does the D1-D2 MCM subsystem FPGA operate either parallel *or* independent processes. Either the processor or FPGA are in control.

74. Importantly, D1, is a system that uses an external RAM-based configuration where the FPGA does *not* run independently or in parallel with the processor.

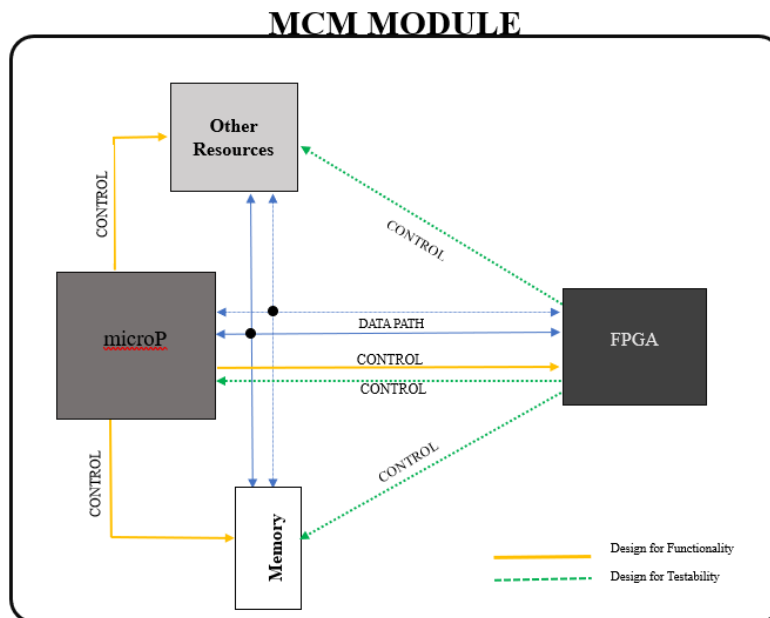


Figure 1 : Designing for Testability for a Typical MCM.

75. The technical documentation relied by Dr. Shanfield illustrates this. When D2 is considered in view of *Kostarnov* (EX. 1036) - a publication referenced by Dr. Shanfield (but not used as a basis for any Ground) - illustrates the relevant mode of operation in D2 is the "Functional Unit Mode" of the Xilinx 4013 FPGA from *Kostarnov*. As shown, the FPGA receives instructions from the

CPU confirming that it is *not* operating independently in parallel. There, he makes no assertions that the FIG. 2 Functional Unit Mode configuration is operating independently in parallel with the processor. Nor can he; the references themselves clearly explain they are not. For example, in *Kostarnov* at 4, FIG. 2 the FPGA is controlled by the CPU and all data bus operations are controlled by the CPU. The FIG. 2 architecture receives instructions (in orange) only from the CPU indicating that it is a unidirectional (one-way) path. See EX. 1036 at 3-4. Thus the “Datapath Mode” mentioned in *Kostarnov* is not relevant to the operation of the D1-D2 MCM subsystem and POSA would clearly understand it as such and that no such operation is shown or suggested.

76. This is illustrated below by comparing the first operating mode of D1-D2 the “design for functionality” mode with Functional Unit Mode of *Kostarnov* which is the most relevant.

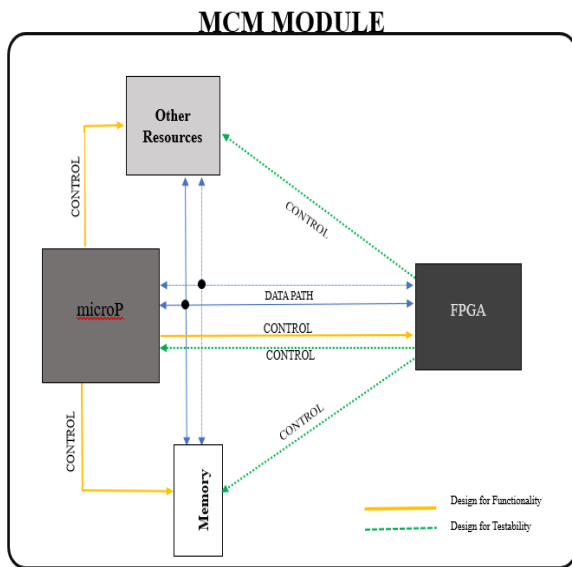


Figure 1: Designing for Testability for a Typical MCM.

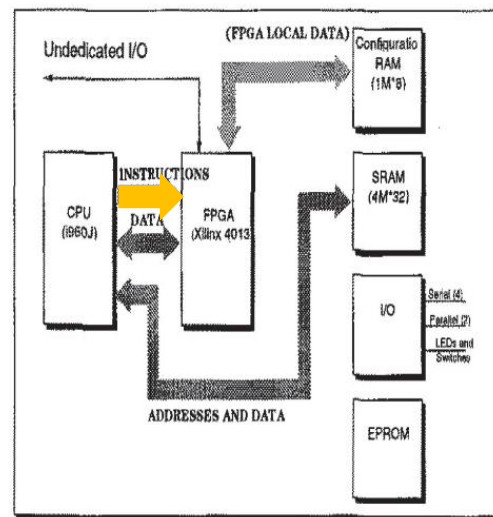


Fig. 2: Functional unit mode

77. As is clearly apparent from the above, D1-D2 operates under the control of the processor, meaning that its FPGA neither operates independently or in parallel during normal operation. Thus, there is agreement between D1-D2 and *Korstanov*, that in light of D2, D1 is not operating independently or in parallel. In the other mode (design for testability), the FPGA is in control and the opposite is true – that is the CPU is under FPGA control.

78. Accordingly, for the reasons above, D1 does not show or suggest “gate array coupled to the processor and configured to run independent processes in parallel with the processor” whether using my definition or any other reasonably appropriate definition.

D1 Theoretical Testing

79. I note that D1 at 3, does theorize 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 subsystem. During JTAG and TAP testing operations, however, an external JTAG computer or other similar common controller is required to control both the DSP processor and the FPGA during testing. Independent parallel processing (such as that alleged is Datapath Mode on *Kostarnov*) is not shown or suggested. See *Dehkordi* at 4:

Cost & Testing Issues

* * * * *

Due to their small feature sizes, MCMs cannot easily be probed or reworked [8]. Therefore, testing and fault isolation becomes a challenging area for MCM designs. Complete boundary scan testing was not feasible for this MCM since only the FPGA supported JTAG. The DSP supports a serial emulation port which was connected to the module pins along with the FPGA TAP port for testing purposes. 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.

80. Because JTAG is control of FPGA which controls DSP they are both under common control of another device while under test, they do not operate any independent processes. This is confirmed by Xilinx's own JTAG/TAP documentation used for the 4010 series FPGA. See EXs. 2002-2005.

81. The testing procedures described in D1 and relied on by Dr. Shanfield are only speculative, as D1 explicitly explains that "[t]esting procedures [were] being constructed at the time of [*Dehkordi's*] publication." See D1 at 4.

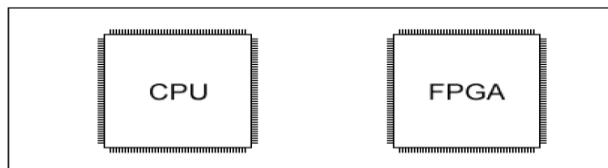
Nor does D1 establish that such procedures were ever completed or, if they were, that they operated as hypothesized. In fact, the reference itself confirms as of the publication date of D1, the MCM system itself was still in fabrication and its overall operation and including yet to be formulated testing procedures, theoretical and therefore not enabled.

JTAG Testing

82. JTAG testing is very well known in the art and was developed in the early 1980s to test continuity and basic functionality of circuits as designs became increasingly more complex and difficult to probe and troubleshoot manually. JTAG testing relies on an external computer to make connections between the components under test by applying various test signals as specified by the JTAG program to confirm continuity and functionality. This is shown below with excerpts from EX. 2002 illustrating how a system, such as an MCM, with both an FPGA and processor are tested via JTAG.

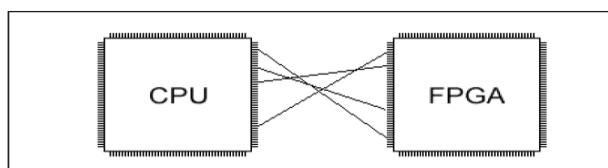
Boundary testing

Here's a simple electronic board (also called "PCB" for "printed circuit board") with two ICs ("integrated circuits"), a CPU and an FPGA. A typical board may have many more ICs.

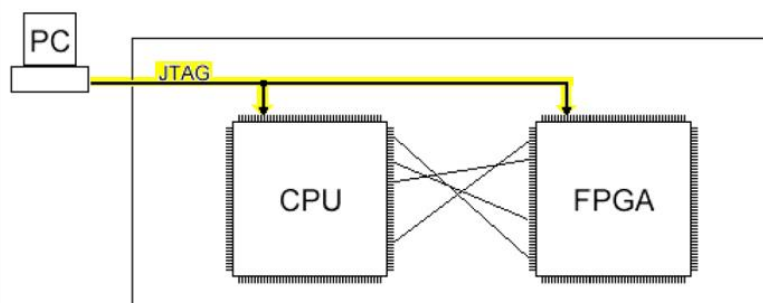


ICs can have lots of pins. So of course, ICs are connected together with lots of connections (PCB traces).

We show just four here. But you can easily have a few thousands on a PCB.



hand. So JTAG was created.



JTAG can take control (or hijack) the pins of all the ICs. On the picture, maybe JTAG is going to make all the CPU pins outputs, and all the FPGA pins inputs. Then by sending some data from the CPU pins, and reading the values from the FPGA pins, JTAG can make sure that the board connections are fine.

83. As can be seen from EX. 2002, the JTAG PC takes control of the pins of all of the ICs (FPGA and processor) through the JTAG connection highlighted in yellow above. This illustrates both the CPU and FPGA are under the common control of another device and therefore not independent. The JTAG PC then conducts testing by controlling the flow of data and commands to and

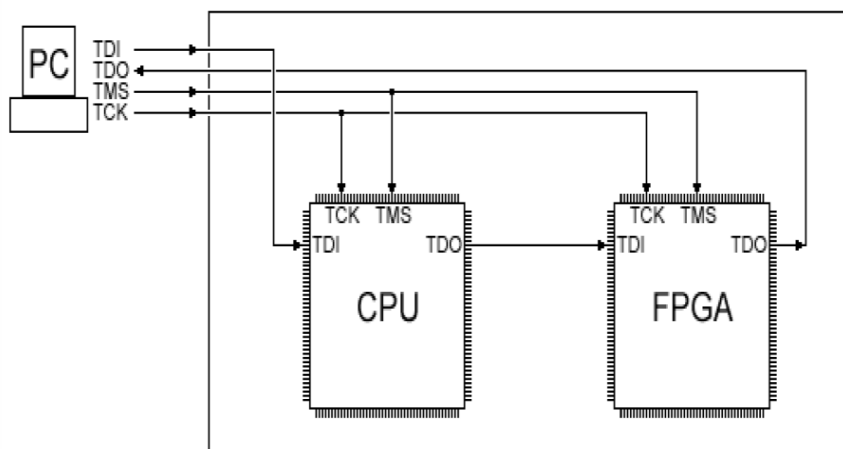
from both the processor and FPGA. During this process, both devices are always under the common control of the JTAG PC and thus never operate independently.

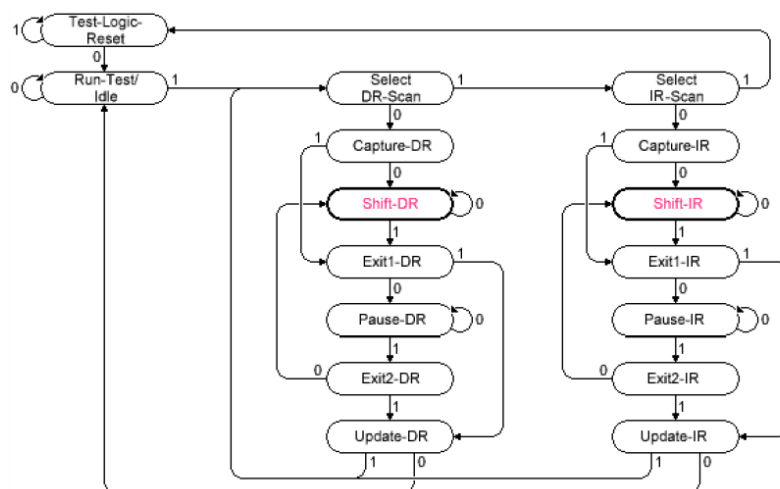
TAP Testing

84. TAP testing is merely a more sophisticated form of JTAG testing that relies on the same architecture and control methodology. As shown below in excerpts from EX. 2003, the same connections from the JTAG PC go to both the FPGA and processor as generally illustrated in EX. 2002 except implements a more detailed testing procedure involving a TAP state machine having 16 states as shown in the flowchart below:

The JTAG TAP controller

We know that a PC is connected to the JTAG bus as illustrated here:





85. The consequence is that the FPGA and processor are again both controlled by the JTAG PC, which merely implements a more thorough TAP methodology. As explained, “all TAP controllers move together (to the same state)” thus demonstrating that both the FPGA and processor are under common control and never operate independently.

86. The content of EXs. 2002 and 2003 are confirmed by Xilinx’s own references which contain the same materials. EX. 2002 is confirmed by the content of EX. 2004, with EX. 2003 confirmed by EX. 2005.

87. Moreover, bus arbitration is not equivalent to independent operation, nor is independent operation required for bus arbitration. The only place that D1 mentions bus arbitration is in the section related to JTAG/TAP testing. D1 at 3. Notably, however, D1 does not mention anything about independent or parallel operation anywhere, including during JTAG/TAP testing. Simply put, D1 does not teach independent operations.

88. Moreover, the Xilinx 4010 datasheet (EX. 1014 at 6) explains that whenever data are written into the FPGA, as is the case with JTAG/TAP testing, it acts in slave and peripheral modes—confirming yet again that the FPGA does not operate independently.

Description

XC4000 Series devices are implemented with a regular, flexible, programmable architecture of Configurable Logic Blocks (CLBs), interconnected by a powerful hierarchy of versatile routing resources, and surrounded by a perimeter of programmable Input/Output Blocks (IOBs). They have generous routing resources to accommodate the most complex interconnect patterns.

The devices are customized by loading configuration data into internal memory cells. The FPGA can either actively read its configuration data from an external serial or byte-parallel PROM (master modes), or the configuration data can be written into the FPGA from an external device (slave and peripheral modes).

Franz (EX. 1005) and Faura (EX. 1006)

89. Neither *Franz* nor *Faura* cures the deficiencies of *Lyke* and D1. Accordingly, the combination of D1, *Lyke*, *Franz*, and *Faura* in Ground 1 fails to teach or suggest a “gate array coupled to the processor and configured to run independent processes in parallel with the processor” as specified in claims 1 (element 1.5) claim 10 (element 10.5) Thus, it is my opinion claims 1-5, 10-12 and are patentable for at least this reason alone.

Halverson (Ex.1025), Marchal (EX.1035), Lyke and D1 All Do Not Show Independent Operation Which is Not Common Knowledge

90. In forming his opinions, Dr. Shanfield relies broadly on his assertion that that FPGA independent operation based on external configuration was common knowledge. However, there is no substantial evidence on the record that supports that assertion. In fact, the opposite is true. As discussed above, D1 in view of D2 does not run independently or in parallel with the processor despite being externally configured. This is also true for *Halverson* and *Marchal*. The same is true for *Lyke* because despite external configuration – it still operates on a processor-controlled clock.

91. *Halverson* does not disclose independent parallel processing but rather discloses just parallel processing. Dr. Shanfield does not assert that *Halverson* shows *independent* parallel processing. This observation is also corroborated by the fact that the USPTO found *Halverson* during examination but decided not to use it as a basis for a rejection. See EX. 2014.

92. Furthermore, *Halverson* itself also explains that its system processor and FPGA devices *do not operate independently*. *Halverson* (EX 1025 at 9:62-10:3) describes the port 1 register that appears in Fig. 14. All bits of the register are set to values of "1" when the machine is reset. Fig. 14 describes the register bit P1.1 as "START mP Run = 0, mP Rset = 1". The minimal processor

(implemented by a Xilinx 3090 FPGA as shown in Fig. 10) is dependent on the system processor to set P1.1 to value 0 in order for the minimal processor run, 10:2-3. In view of this dependency, the other FPGA(s) in *Halverson* (e.g. Fig. 2 FPGA labeled 304) which have their operations directed by the minimal processor, are also *necessarily dependent* on the minimal processor and the system processor by implication.

93. Importantly, *Halverson*, like D1, is another system that uses an external-based configuration where the FPGA configuration does not run independently or in parallel with the processor – again disproving Petitioner’s theory at externally configured FPGAs must operate independently.

94. *Marchal* (EX. 1035) – also does not relate to independent processes. Indeed, the term independent does not appear in *Marchal*, nor does process. Petitioner and its expert acknowledge the distinction between these two devices in EX. 1035, 1; see also EX. 1035, 2-3 (describing loading the FPGA configuration string from “memory”). The FPGA merely loads this configuration (the program) and then operates *whatever program is contained* in that memory - which has nothing to do with independent or parallel operation.

Kostarnov (EX. 1036) Is Not Used as the Basis of Any Ground

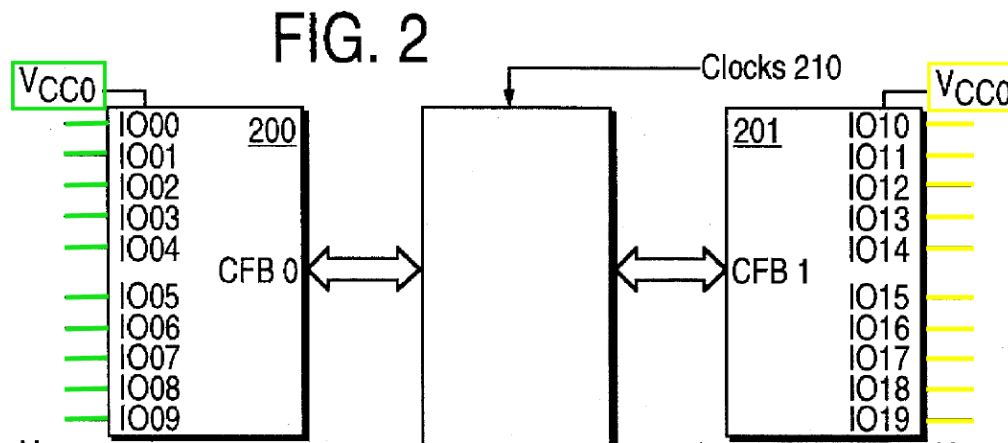
95. The Petition references provided to me do not that FPGAs operated independently in parallel let alone this was “common knowledge” or “a standard

way that FPGAs were deployed with processors” as Petitioner contends. In fact, as established above, all Ground 1 references teach otherwise, with the alleged exception of background reference *Korstanov*. Thus, Petitioners have failed to establish this feature was common knowledge of one of ordinary skill rather than isolated specialized reference.

Ground 4—Based on Lyke, Dehkordi, Franz, Faura, and Steele

96. Ground 4 is simply the references of Ground 1 further combined with *Steele*. Accordingly, Ground 4 suffers from all the same deficiencies as noted above with respect to Ground 1. Claims 5, 10-11, 13-14, 15, 17, 19-20 are patentable additionally because *Steele* fails to show or suggest a “power converter” as specified in these claims.

97. *Steele* describes a programmable logic device configured to operate like a voltage replicator that distributes an applied input voltage for output at other pins. Simply stated, *Steele* operates by accepting an input voltage generated at an external source at the VCCO pin of a Configurable Logic Block (CFB) and merely distributes that voltage at the input/output (I/O) pins for to external devices. This is explained at EX. 1009 at 4:28-43, and is shown in the annotated version of FIG. 2 below where green represents a 5V level (CFB 0) and yellow represents a 3V level (CFB 1):



98. Notably, *Steele* does not generate the applied VCCO voltage levels itself, but rather merely uses them. The voltage generation is done elsewhere on another external device and those voltages are applied to the VCCO input from that external source for reproduction by *Steele*'s CFB. *Steele* does not convert power because it does not change one voltage level to another as the claims require under Patent Owner's claim construction. Instead, *Steele* merely distributes the same voltage level applied VCCO at the output I/O pins. For example, IO00-IO09 have 5V outputs whereas IO10-IO19 have 3V outputs. See EX. 1009, 2:22-26; 4:28-43. There is no change or *conversion* (i.e., changing one thing to another – plain meaning). Merely distributing the same voltage eight times is not converting power, but rather is merely distributing the same voltage multiple times.

99. This is different from the challenged claims of the '177 Patent, which require, e.g., a portion of the gates in the FPGA to operate as a power converter capable of receiving an input voltage level and changing it to another

voltage level as the claims require. *See, e.g.,* '177 Patent, claim 5.

100. *Steele* does not convert power but rather merely *distributes* the *same* voltage level VCCO at the output I/O pins. For example, IO00-IO09 have 5V outputs whereas IO10-IO19 have 3V outputs. *See* EX. 1009 at 2:22-26; 4:28-43. Notably the phrase “power converter” does not appear in *Steele* nor does *Steele* describe converting power in any way.

101. Accordingly, *Steele* fails to teach or suggest “[a] ... *embedded power converter* ...” for at least three reasons. The Petition fails to address the power conversion claim term which requires changing one voltage level to another different voltage level. Power conversion is not mentioned in the *Steele* reference. The Petitioner never addresses what power conversion is or how it is accomplished in *Steele*. Further, such treatment and explanation are also missing from the reference itself. The Petition thus offers nothing more than conclusory statements that are unsubstantiated by any analysis or explanation based on *Steele*.

102. *Third*, no such combination would ever be made. The Petition states that “[i]n the combined system with *Steele*, a portion of the gates in the FPGA are utilized to carry out power conversion, for the predictable benefits described above, including obviating the need for separate converter hardware apart from the existent FPGA.” Petition at 48, *emphasis added*. But this statement is demonstrably wrong. As explained above, the VCCO signal that establishes the required voltage level is not generated by any circuitry in *Steele* instead is created

externally. This appears not be in dispute See Petition at 48 (“a plurality of different voltages can be supplied to the V_{cc0} pins of different functional blocks, resulting in a plurality of different voltages.”). Emphasis added.

103. Thus, the Petition states the VCCO voltages must come from somewhere outside of *Steele*. Therefore, *Lyke* alone provides all the necessary reference and operating voltages specified in the claim and *Steele* provides *no* added benefit whatsoever to the proposed combination. Accordingly, *Steele* is merely a superfluous, cumulative reference that belies the motivation for making the combination of Ground 4 which appears to be offered only for the sole (impermissible) reason of alleging unpatentability.

104. The combination of *Lyke*, *Dehkordi*, *Franz*, *Faura*, and *Steele* in Ground 4 fails to teach or suggest the limitations in independent claims 13, and 20 of a “*power converter capable of... generating... reference voltage*” (elements 13.6 and 20.1) or *an internal embedded power converter* (claim 15). Thus claims 5, 14-15, and 20 are patentable for at least this reason alone.

Grounds 2, 3, and 5

105. Because Grounds 2-3 are based on Ground 1, claims 2 and 3 are patentable for at least the same reasons that claim 1 is patentable over Ground 1. Similarly, because Ground 5 is based on Ground 4, claims 12 and 18 are patentable for at least the same reasons that claims 13 and 15 are *patentable* over Ground 4. Claims 2, 3, 12, and 18 are independently patentable.

VI. CONCLUSION

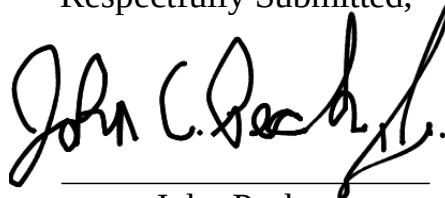
106. In my opinion, all of the challenged claims are patentable over the prior art relied on by Petitioner.

107. I reserve the right to supplement my opinions to address any information obtained, or positions taken, based on any new information that comes to light throughout this proceeding.

I hereby declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct, and that all statements made of my own knowledge are true and that all statements made on information and belief are believed to be true. I understand that willful false statements are punishable by fine or imprisonment or both under Section 1001 of Title 18 of the U.S. Code.

Dated: 12/21/2023

Respectfully Submitted,



John C. Peck

John Peck

John Peck

Bay2Sierra Silicon Services, Inc.

601 Duncan St.

San Francisco, CA 94131

Business: 415-643-9428

Cell: 415-420-8449

Email: peck@bay2sierra.com

Website: www.bay2sierra.com

PROFESSIONAL SUMMARY

John Peck is an expert in complex constrained random design verification and board validation environments. He has developed and performed verification of memory controllers (SDRAM, DDR), on-chip interconnect networks, SoC integration, SPI controllers, and processor instruction set architectures (ISAs). He is knowledgeable about performance modeling and architectural validation of network processors and has designed high throughput software applications including search indexers and text extractors. He holds a bachelor's degree in Computer Engineering from Carnegie Mellon University and a master's degree in Computer Science from the University of California, Los Angeles.

John Peck has over 25 years of experience as a software and hardware engineer with over 11 years of experience as a technical analyst for hardware and software intellectual property cases. He has drafted numerous claim charts and prepared expert reports for software copyright, trade secret, and patent litigation. He is comfortable researching and expressing complex technical issues. He is experienced with source code review and able to explain its operation in ways that most people can understand.

EXPERIENCE

1/08 – Present: Bay2Sierra Silicon Services Inc., San Francisco, CA

Principal Engineer, Consulting and contracting services:

- Reads patents to understand infringement and validity issues
- Examines software source code for determining patent infringement
- Examines HDL code (e.g. Verilog and VHDL RTL) for determining patent infringement
- Examines schematics for determining patent infringement
- Writes expert reports
- Performance analysis and modeling
- ASIC/FPGA design verification, architecture, and design
- Complex SOC platform: IP design vetting/integration, and integration-level verification
- Design of memory controllers, system and device-level error management technologies
- Design verification/validation of FPGA designs implemented in PCIe platforms
- Design of hardware and software for high throughput applications
- Development of electronic design automation (EDA) tools
- Development of OpenSSL crypto-engine software integration for hardware devices including Atmel ATECC508A

12/03 – 12/07: Kazeon Systems, Mountain View, CA

Senior member of technical staff, office of the CTO:

- Responsible for system hardware and set performance goals and expectations
- Managed technical engagement with vendors and customers
- Designed high throughput search indexer based on Lucene
- Designed and implemented vectorized algorithms on x86 processors for text extraction at 1Gbps line rate throughputs

08/99 – 12/03: Redback Networks, San Jose (formerly with Siara Systems)

Senior member of technical staff: **Network processors:**

- Architectural validation using performance modeling methodologies: Implemented custom cycle-based model in C++.
- Functional verification of processor ISA, inter-chip interface blocks, performance measurement counters.
- RTL design of DDR SDRAM memory controller optimized for high throughput.

10/97 – 08/99: Advanced Micro Devices, Inc., Sunnyvale

Senior design engineer, Platforms Products Division: **K7 Northbridge chipset** design team responsibilities include: Design, writing RTL, verification, synthesis, and hand placement. Wrote verification testbench for SDRAM memory controller, implemented RTL/verified GART virtual address translation block, enhanced AGP block for 4x operation.

1994 – 1997: University of California, Los Angeles

Graduate Student Researcher on ARPA funded project (Supercomputer Supernetwork joint work of UCLA and the Jet Propulsion Laboratory). Co-Designed/Co-implemented Optical Channel Interface Gateway for Myrinet Network.

02/96 - 06/97: Consultant/Software Engineer (contractor w/Activision Studios), Los Angeles, CA

- Video Game Software Engineer for Win95 Platform. Product implemented is entitled “Dark Reign: The Future of War”
- Video Game Software Engineer for PC Platform. Product implemented is entitled “Mechwarrior 2: Mercenaries.”

05/95 – 07/95: Software Engineer Redline Games (contractor to Activision Studios), Los Angeles, CA

- Video Game Software Engineer for PC platform. Product implemented is entitled “Mechwarrior 2: 31st Century Combat.”

Summer 1994: Redline Games (under contract with Activision Studios)

Video Game Software Engineer for SNES platform (Nintendo game machine). Product implemented is entitled “Pitfall: The Mayan Adventure.”

Summer 1992: NCR Corporation E&M Atlanta

OS Development group: Programmer / Analyst device driver development DOS 5.0.

Summer 1991: Transarc Corporation

Product Support: Distributed system testing and debugging.

1988 - Summer 1993: Foxfire Technologies Corporation

Analysis and Computer Programming:

Inquiry system for real time shop floor control, Controller for real time data collection, Capacity planning and scheduling system, Finished goods inventory system, Shipping system, Control software for Rapistan automated conveyor sortation system, Router for real time data collection.

LEGAL CONSULTING

I have consulted on a total of 30 litigations. I have provided expert testimony 9 times at deposition and 3 times at trial. I have obtained permission to disclose the matters listed below. There is a total of 2 undisclosed matters that pertain to ongoing litigation with accused products including web-based software, and field programmable gate arrays (FPGAs).

02/2022 – Present: Safe Driving Technologies LLC v Ford Motor Company

Law Firm: Bailey Duquette P.C.

Client: Safe Driving Technologies LLC

Court: U.S. District Court, District of Delaware

Case: 21-cv-64-MN

- Alleged patent infringement
- Reviewed source code
- Submitted declarations for IPR
- Testified at deposition

01/2022 – 09/2022: MasterObjects, Inc. v. Facebook, Inc.

Law Firm: Hosie Rice LLP

Client: MasterObjects, Inc.

Court: U.S. District Court, Northern District of California

Case: 6:20-cv-00087-ADA

- Alleged patent infringement
- Submitted declarations on patent infringement
- Testified at deposition

12/2021 – Present: Express Mobile Inc. v Slack Technologies, Inc..

Law Firm: Feinberg Day Kramer Alberti Lim Tonkovich & Belloli LLP

Client: Express Mobile, Inc.

Court: U.S. District Court, Northern District of California

Case: 3:21-cv-02001-RS

- Alleged patent infringement
- Reviewed source code

10/2021 – 09/2022: MasterObjects, Inc. v. Amazon, Inc.

Law Firm: Hosie Rice LLP

Client: MasterObjects, Inc.

Court: U.S. District Court, for the Northern District of California San Francisco Division

Case: 5:20-cv-08103-WHA (KAW)

- Alleged patent infringement
- Submitted declaration on patent infringement
- Testified at deposition

07/2021 – Present: Express Mobile Inc. v Ebay Inc.

Law Firm: Feinberg Day Kramer Alberti Lim Tonkovich & Belloli LLP

Client: Express Mobile, Inc.

Court: U.S. District Court, Northern District of California

Case: 3:21-cv-02001-RS

- Alleged patent infringement
- Reviewed source code

08/2020 - Present: Rensselaer Polytechnic Institute and CF Dynamic Advances LLC v. Amazon.com, Inc.

Law Firm: Robins Kaplan

Client: Rensselaer Polytechnic Institute and CF Dynamic Advances LLC

Court: U.S. District Court, for the Northern District of New York

Case: 1:18-cv-549-BKS-CFH

- Alleged patent infringement
- Reviewed source code
- Assisted with preparing source code citations for infringement contentions
- Submitted declaration on source code analysis
- Testified at deposition

07/2020 – 11/2023: VB Assets, LLC v. Amazon.com Inc., et al.

Law Firm: Wilson Sonsini Goodrich & Rosati

Client: VB Assets, LLC.

Court: U.S. District Court, for the District of Delaware

Case: 1:19-cv-01410-MN

- Alleged patent infringement
- Reviewed source code
- Prepared infringement contentions pertaining to source code analysis
- Submitted expert report on source code analysis
- Testified at deposition
- Testified at trial

03/2020 – Present: VLSI Technology LLC. v. Intel Corporation.

Law Firm: Irell & Manella LLP

Client: VLSI Technology, LLC.

Court: U.S. District Court, Northern District of California San Jose Division

Case: 5:17-cv-05671-BLF

- Alleged patent infringement

- Reviewed HDL, C, and assembly language source code
- Assisted with response to claims of patent invalidity
- Assisted with writing of expert report

03/2020 – Present: VLSI Technology LLC. v. Intel Corporation.

Law Firm: Irell & Manella LLP

Client: VLSI Technology, LLC.

Court: U.S. District Court, for the District of Delaware

Case: 1:2018-cv-00966

- Alleged patent infringement
- Reviewed HDL, C, and assembly language source code
- Assisted with response to claims of patent invalidity

03/2020 – Present: VLSI Technology LLC. v. Intel Corporation.

Law Firm: Irell & Manella LLP

Client: VLSI Technology, LLC.

Court: U.S. District Court, for the Western District of Texas

Case: 6:19-cv-00254

- Alleged patent infringement
- Reviewed HDL, C, and assembly language source code
- Assisted with response to claims of patent invalidity

03/2020 – 07/2020: GPU++ LLC. v. Qualcomm Incorporated; Qualcomm Technologies, Inc.

Law Firm: Steptoe & Johnson LLP

Client: GPU++

Court: U.S. District Court, Western District of Texas

Case: 6:19-cv-00474-ADA

- Alleged patent infringement

11/2019 – 08/2020: Motorola Solutions, Inc. v. Hytera Communications Corporation, Ltd.

Law Firm: Shelston IP

Client: Hytera

Court: Federal Court of Australia

District Registry: New South Wales

Division: General

Case: NSD1283/2017

- Gave evidence at trial (Testified at trial)
- Submitted affidavits (expert reports)

08/2019 – 03/2022: Motorola Solutions, Inc. and Motorola Solutions Malaysia Sdn. Bhd. v. Hytera Communications Corp Ltd., Hytera America, Inc., and Hytera Communications America (West), Inc.

Law Firm: Steptoe & Johnson LLP

Client: Hytera

Court: U.S. District Court, Northern district of Illinois

Case: 1:17-cv-01973

- Alleged theft of trade secrets
- Testified at deposition
- Testified at trial
- Reviewed source code
- Wrote expert report

07/2019 – 12/2019: Neodron v. Samsung, Microsoft, Lenovo, Motorola, Dell, and HP

Law Firm: Russ August & Kabat

Client: Neodron

Court: U.S. International Trade Commission

Case: Inv. No. 337-TA-1162

- Alleged patent infringement of capacitive touch screens
- Reviewed source code
- Assisted with writing expert reports

05/2019 – Present: Alphonso Inc. v Free Stream Media Corp d/b/a Samba TV

Law Firm: Goodwin Procter LLP

Client: Alphonso, Inc.

Court: U.S. District Court, for the District of Delaware

Case: 1:18-cv-01205-CFC

- Alleged patent infringement
- Reviewed source code
- Prepared infringement contentions pertaining to source code analysis

04/2018 – 06/2019: Broadcom Corporation. vs. Toyota Motor Corporation, Japan, Toyota Motor North America, Inc. Toyota Motor Sales, U.S.A. Inc., Toyota Motor Engineering & Manufacturing North America, Inc., Toyota Motor Manufacturing, Indiana, Inc., Toyota Motor Manufacturing, Kentucky, Inc. Toyota Motor Manufacturing, Mississippi, Inc., Toyota Motor Manufacturing, Texas, Inc. Panasonic Corporation, Japan, Panasonic Corporation of North America, Denso Ten Limited, Japan, Denso Ten America, Limited., Renesas Electronics Corporation, Japan, Renesas Electronics America, Inc., Japan Radio Corporation, Japan.

Law Firm: Steptoe & Johnson LLP

Client: Broadcom Corporation

Court: U.S. International Trade Commission

Case: Inv. No. 337-TA-1119

- Alleged patent infringement of SoC integrated circuits in audiovisual products
- Examined HDL and firmware source code
- Wrote expert report pertaining to source code analysis

03/2018 – 12/2018: Blast Motion, Inc. vs. Diamond Kinetics, Inc.

Law Firm: Wilson Sonsini Goodrich & Rosati

Client: Blast Motion, Inc.

Court: U.S. District Court, Western District of Pennsylvania

Case: 2:17-cv-00733-MRH

- Alleged patent infringement of patents related to natural motion capture
- Examined source code including sensor firmware and mobile applications
- Prepared infringement contentions pertaining to source code analysis and assisted with depositions

06/2017 – 12/2017: Broadcom Corporation. vs. MediaTek Inc., MediaTek USA Inc., MStar Semiconductor Inc., Sigma Designs Inc., LG Electronics Inc., LG Electronics U.S.A., Inc., Funai Electric Company, Ltd., Funai Corporation Inc., P&F USA, Inc., Vizio, Inc.

Law Firm: Steptoe & Johnson LLP

Client: Broadcom Corporation

Court: U.S. International Trade Commission

Case: Inv. No. 337-TA-1047

- Alleged patent infringement of SoC integrated circuits in audiovisual products
- Examined HDL and firmware source code
- Wrote declaration regarding source code production
- Wrote expert report pertaining to source code analysis

06/2015 – 12/2017: Blast Motion, Inc. vs. Zepp Labs Inc.

Law Firm: Wilson Sonsini Goodrich & Rosati

Client: Blast Motion, Inc.

Court: U.S. District Court, Southern District of California

Case: 3:2015cv00700

- Alleged patent infringement of patents related to natural motion capture
- Examined source code including sensor firmware and mobile applications
- Wrote expert report pertaining to source code analysis and assisted with depositions

06/2017 – 06/2017: Avatar Integrated Systems

Law Firm: Davis Wright Tremaine LLP

Client: Avatar Integrated Systems

Project: Clean room re-development of EDA tool control language to copyright infringement

2/2017 – 5/2017: Blackberry Limited vs. BLU Products, Inc.

Law Firm: Cozen O'Connor LLP

Client: BLU Products

Court: U.S. District Court, Southern District of Florida

Case: 16-23535-CIV-MORENO

- Alleged patent infringement of standards essential patents
- Assisted with the writing of an inter partes review (IPR) declaration and an expert report

12/2016 – 2/2017: Papst Licensing v. Apple, LG Electronics, ZTE Corporation, Samsung Electronics, Lenovo, Motorola Mobility, Huawei Technologies

Law Firm: DiNovo Price Ellwanger

Client: Papst Licensing

Court: U.S. District Court, Eastern District of Texas, Tyler Division

Case: Civil Action No. 6:15-cv-1095

- Alleged patent infringement of mobile phones

09/2016 – 10/2016: Immersion Corporation v. Apple Inc., et al.

Law Firm: DLA Piper

Client: Apple Inc.

Court: U.S. International Trade Commission

Case: Inv. No. 337-TA-1004 and Inv. No 337-TA-990 (consolidated)

- Reviewed Apple product source code in support of rebuttal to infringement contentions

1/2016 – 6/2016: Verasonics v. Alpinion Medical Systems

Law Firm: Davis Wright Tremaine LLP

Client: Verasonics

Venue: American Arbitration, International Centre for Dispute Resolution

Case: 01-15-0002-9484

- Alleged trade secret theft of software for an ultrasound research platform
- Compared software source code using CodeSuite
- Assisted with the writing of a declaration and an expert report

09/2015 – 04/2016: Objectivision Pty Ltd. v. Visionsearch Pty Ltd and University of Sydney

Law Firm: Marque Lawyers

Client: Objectivision Pty Ltd.

Case: NSD2433 of 2013

- Alleged trade secret theft and copyright violation of software that comprises a medical diagnostic system relating to vision.

2/2015 – 1/2016: NNG, Kft. v. Ava Enterprises, Inc.

Law Firm: Lewis Roca Rothgerber Christie LLP

Client: NNG, Kft

Court: U.S. District Court, Central District of California

Case: 2:14-CV-00220

- Alleged copyright infringement

1/2015 – 6/2015: Salesforceone LLC. & PARS International Computer, Inc. v Salesforce.com, Inc.

Law Firm: Wilson Sonsini Goodrich & Rosati

Client: Salesforce.com, Inc.

Court: U.S. District Court, Northern District of California

Case: 4:2015-cv-02174

- Alleged trademark infringement

1/2015 – 6/2015: NVidia Corporation vs. Samsung Electronics, Qualcomm Inc.

Law Firm: Orrick LLP

Client: NVidia Corporation

Court: U.S. International Trade Commission
Case: Inv. No. 337-TA-932

- Alleged patent infringement of GPU integrated circuits
- Examined HDL code
- Submitted expert report pertaining to source code analysis

11/2012 – 5/2014: Intellectual Ventures v. Altera, Microsemi, Lattice Semiconductor, And Xilinx
Law Firm: Desmarais LLP

Client: Intellectual Ventures

Court: U.S. District Court, District of Delaware

Case: C.A. No. 10-1065-LPS

- Alleged patent infringement of FPGA and ASIC integrated circuits
- Examined schematics and Verilog code
- Assisted with writing expert reports and with depositions

EDUCATION

Masters Degree in Computer Science, University of California, Los Angeles, March 1995

Bachelor of Science in Computer Engineering, Carnegie Mellon University, May 1992

PUBLICATIONS

1. J. Cong, J. Peck, et. al., "The Supercomputer Supernet Testbed: A WDM-based Supercomputer Interconnect," *Journal of Lightwave Technology*, 1996.
2. J. Cong, J. Peck, Y. Ding, "RASP: A General Logic Synthesis Systems for SRAM-based FPGAs," *ACM/IEEE Inter. Symposium on FPGAs*, 1996.
3. J. Cong, J. Peck, et. al., "The Supercomputer Supernet (SSN): A High-Speed Electro-Optic Campus and Metropolitan Network," *Society of Photo-Optical Instrumentation Engineers*, 1996.
4. J. Cong, J. Peck, "On Acceleration of Logic Synthesis Algorithms using FPGA-based Reconfigurable Coprocessors," *IEEE Inter. Symposium on FPGAs for Custom Computing Machines*, 1997.

PATENTS

1. Peck, John; Davis, Gregory; Huynh Quoc, "Serial peripheral interface," U.S. Patent 8,589,717.
2. Novak; Stephen T., Peck, Jr.; John C., Waldron; Scott, "Method and apparatus for optimizing memory performance with opportunistic refreshing," U.S. Patent 6,046,952.
3. Novak; Stephen T., Peck, Jr.; John C., Waldron; Scott, "Method and apparatus for optimizing memory performance with opportunistic refreshing," U.S. Patent 6,147,921.
4. Novak; Stephen T., Waldron; Scott, Peck, Jr.; John C., "Queue-based memory controller," U.S. Patent 6,295,586.
5. Novak; Stephen T., Peck, Jr.; John C., "Method and apparatus for optimizing memory performance with opportunistic pre-charging," U.S. Patent 6,360,305.
6. Novak; Stephen T., Peck, Jr.; John C., "Queue-based control mechanism for queue-based memory controller," U.S. Patent 6,393,531.
7. Novak; Stephen T., Waldron; Scott, Peck, Jr.; John C., "Queue-based memory controller," U.S. Patent 6,496,906.
8. Peck, Jr.; John C., Subramanian; Sridhar P., "Optimized static sliding-window for ACK sampling," U.S. Patent 6,601,182.
9. Peck, Jr.; John C., Subramanian; Sridhar P., Waldron; Scott, "Optimizing the translation of virtual addresses into physical addresses using a pipeline implementation for least recently used pointer," U.S. Patent 6,686,920.
10. Peck, Jr.; John C., Subramanian; Sridhar P., Waldron; Scott, "Distributed translation look-aside buffers for graphics address remapping table," U.S. Patent 6,741,258.

SPECIAL KNOWLEDGE AND SKILLS

- FPGAs, ASICs, SoCs, Processors including GPUs, Chipsets
- HDL-Software Interfaces:
 - Programming Language Interface (PLI)
 - SystemVerilog Direct Programming Interface (DPI)
 - Verilog Procedural Interface (VPI)
- Operating systems:

- Linux
- UNIX
- Windows
- Programming languages:
 - C, C++
 - Assembly language, including x86 vector programming
 - Python, including embedded python
 - Perl
 - Java
- Scripting languages
 - Bash
 - Csh
- Synopsys Design Compiler (DC)
- Verilog
- System Verilog
- Verification methodologies
 - Universal Verification Methodology (UVM)

CERTIFICATIONS AND COURSE TRAINING

- Code Suite Certified by SAFE Corporation
- Computer Forensics CNIT 121
- Ethical Hacking and Network Defense CNIT 123
- Intellectual Property PLS 13

PROFESSIONAL ORGANIZATIONS

Senior member of the IEEE and IEEE Consultants' Network of Silicon Valley (CNSV)

Senior member of the Association for Computing Machinery (ACM)

Affiliate member of Professional & Technical Consultants Association (PATCA)