

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ARM LTD., MEDIATEK INC,

Petitioners,

v.

DAEDALUS PRIME LLC,

Patent Owner.

Case No.: IPR2025-00207

U.S. Patent No. 8,984,228

**DECLARATION OF NADER BAGHERZADEH, PH.D.
IN SUPPORT OF PATENT OWNER'S RESPONSE**

TABLE OF CONTENTS

I.	INTRODUCTION	- 1 -
II.	QUALIFICATIONS	- 2 -
III.	LEGAL STANDARDS	- 7 -
IV.	A PERSON OF ORDINARY SKILL IN THE ART ("POSITA").....	- 10 -
V.	OVERVIEW OF THE '228 PATENT	- 11 -
A.	Construction of a "Single Caching Agent".....	- 14 -
VI.	THE PETITION FAILS TO DEMONSTRATE THAT ANY CHALLENGED CLAIM IS UNPATENTABLE..	- 16 -
A.	The Challenged Claims are Not Unpatentable in View of Agarwal.....	- 17 -
1.	Overview of Agarwal	- 17 -
2.	The Single Caching Agent: Limitation of Claim 1 is Not Disclosed or Suggested by Agarwal.....	- 19 -
3.	Claim 2 is Not Obvious in View of Agarwal	- 34 -
4.	Claim 6 Is Not Obvious in View of Agarwal	- 42 -
5.	Claim 11 and Its Dependent Claims Are Not Obvious in View of Agarwal	- 44 -
6.	Claim 14 is Not Obvious in View of Agarwal	- 47 -
B.	Overview of Sinharoy	- 50 -

1.	Sinharoy Does Not Disclose or Suggest a Single Caching Agent as Recited in Claim 1	- 52 -
2.	Sinharoy Does Not Disclose or Suggest Distributed LLC Portions Each Associated with a Corresponding Core, as Recited in Claim 4	- 55 -
3.	Sinharoy Does Not Disclose or Suggest an Allocation Transaction Received from an IIO Module, as Recited in Claim 6	- 58 -
4.	Sinharoy Does Not Disclose or Suggest I/O Device-Originated Allocation, Forwarding, and Direct Storing, as Recited in Claim 11	- 59 -
C.	Claims 2, 3, and 16 Are Patentable Over Sinharoy and Blumrich	- 60 -
1.	Summary of Blumrich	- 60 -
2.	Claim 2 is Patentable Over Sinharoy and Blumrich..	- 62 -
3.	Claim 3 is Patentable Over Sinharoy and Blumrich..	- 66 -
4.	Claim 16 is Patentable Over Sinharoy and Blumrich-	- 71 -
D.	Claim 5 is Patentable Over Sinharoy and Sistla and Claim 19 is Patentable Over Sinharoy, Blumrich, and Sistla ...	- 72 -

I, Nader Bagherzadeh, declare as follows:

I. INTRODUCTION

1. I have been retained on behalf of Patent Owner, Daedalus Prime LLC (“Daedalus”), to provide my opinions in connection with this proceeding. In doing so, I have been asked to provide my independent review, analysis, and insights regarding the technology involved in U.S. Patent No 8,984,228 (the “’228 patent”) and how it compares with the art asserted in the petition challenging the patentability of its claims. In this Declaration, I set forth the independent opinions that I have reached and the basis for those opinions using the information currently available to me.

2. I am being compensated for my work in this matter at my standard hourly rate for consulting services. My compensation is not dependent on the outcome of the case and does not affect the substance of my statements in this Declaration. I have no financial interest in the matter or the patents at issue.

3. I have reviewed the specification and the claims of the ’228 patent. I am familiar with the technology involved therein, having worked in the field for many years.

4. To prepare this Declaration, I have considered the petition and the references cited therein, including the declaration of Dr. Hassoun, the ’228 patent, my own experience and knowledge, and the documents cited herein. I provide my

testimony from the perspective of a person of ordinary skill in the art at the time of the invention. I was at least a person of ordinary skill in the art at the time of the invention. I am familiar with the knowledge and skill level of these people at the time of the invention because I was involved in the industry at the time and had frequent interactions with such people at the relevant time. I have also been asked to construe the claims of the '228 patent, applying the plain and ordinary meaning of the terms thereof from the perspective of a person of ordinary skill in the art in light of the specification and other intrinsic evidence.

II. QUALIFICATIONS

5. Below is a summary of my education and experience. My *curriculum vitae*, included as **Appendix A** hereto, records my education, experience, and publications in greater detail.

6. I am currently a professor in the Department of Electrical Engineering and Computer Science at the University of California, Irvine {"UCI"). I have held this position since 2003. Previously, I held the position of Professor and Chair of Electrical Engineering and Computer Science (ECE) at UCI, from 1998 to 2002. Prior to that I was an Associate Professor in the ECE Department at UCI, from 1993 to 1997. Prior to that I was an Assistant Professor in the ECE Department at

UCI, from 1987 to 1992. In 2015, I received Professor of the Year for the Department of Electrical Engineering and Computer Science at UCI.

7. I earned a Bachelor of Science degree, with honors, in Electrical Engineering from the University of Texas at Austin in 1977. I earned a Master of Science degree in Electrical Engineering from the University of Texas at Austin in 1979. I earned a Ph.D. in Computer Engineering from the University of Texas at Austin in 1987.

8. I have over 45 years of experience in the Electrical Engineering and Computer Science field. That experience includes holding the position of Member of Technical Staff, No. 5 ESS Hardware/Software T1 Line Interface Design at AT&T Bell Labs from 1980 to 1984. In this position I received the Outstanding Achievement award from AT&T Bell Labs in 1984. In 2002, I received the Best Paper award in Institute of Electrical and Electronics Engineers (IEEE) Transactions on Very Large Scale Integration (VLSI) Design as well as the Best Student Paper award co-authored with my doctoral students in the proceedings of Asia and South Pacific Design Automation Conference (ASPDAC) '02. In 2020, I received the Best Paper award in the proceedings of the IEEE International Conference on Edge Computing (EDGE). In 2014, I received the Khwarizmi International Award and became an IEEE Fellow.

9. Over the past 45+ years my research has focused on hardware, software, and VLSI design of low-power and high-performance computer systems for applications in mobile systems, computer graphics processing, 3D integrated circuits, memory structures (e.g., static random-access memory (SRAM), dynamic random-access memory (DRAM), flash memory, and register files), Network-on-Chip (NoC), and integrated sensors. This includes proposing, with colleagues, an integrated approach that allows cross-layer verification of both a cache coherence protocol and a communication fabric in a multicore processor all at once.

10. Since joining UCI in 1987, I have been teaching, researching, and consulting regarding almost all aspects of high performance computer systems, including but not limited to VLSI design, computer architecture, parallel implementation of computationally critical applications such as signal processing and error correcting schemes, DRAMs, SRAMs, and interfacing flash memories.

11. Over the course of my career, I have supervised more than 40 doctoral students and 30 masters' students who have taken key positions in the electrical engineering and computer engineering sector. I have served as post-doctoral advisor for more than 10 scholars that worked with my research group.

12. In 2000, I cofounded a company called Morpho Technologies, which was focused on the design and development of low power and high-performance digital signal processors (DSPs) for mobile applications. As part of my duties as

the chief technologist for the company, I evaluated patents, technical reports, and presentations related to memory chip designs, DSPs, and parallel processing algorithms for mobile platforms.

13. For Morpho, I designed, developed, and tested the key memory block responsible for exchanging data between the host processor and the slave processor (array processor). This subcomponent was designed to meet the power and performance requirements of the architecture for mobile embedded applications. I also designed the main data storage of the microcontroller (a reduced instruction set computer, or “Tiny RISC”) that was used for storing temporary data between the Tiny RISC and the array processor, based on SRAM technology, which is a type of volatile memory. I was one of the primary contributors for the prototype board design of the Morpho chip, which utilized DRAMs, field-programmable gate arrays (FPGAs), flash memory and other prototype board components.

14. As part of my teaching and research experience, I have been responsible for developing circuits, electronic modules, and related architectures for a variety of memory systems. For example, I designed the memory block responsible for managing out-of-order execution of data registers corresponding to different threads in the multithreaded superscalar DSP that was developed at UCI. I also designed the original temporary data storage subcomponents for the VLIW integer processor (ViPER) and the original Tiny RISC processor. ViPER was

considered the very first ever VLIW processor designed using Metal Oxide Semiconductor Implementation Service (MOSIS) 1.2-micron technology that included memory and logic circuits. I have also done extensive work on the design of efficient data correction schemes using reconfigurable computing based on the Morpho parallel array architecture.

15. In 2006, I analyzed the hardware implementation of low-density paritycheck (LDPC) as a consultant for a local flash memory module design company. In this capacity, with the help of one of my doctoral students, we modeled the hardware and evaluated the computation complexity of using an application-specific integrated circuit (ASIC) for implementing LDPC for a flash memory-based board.

16. Currently, I am working on developing accelerators and optimization techniques for implementation of efficient machine learning platforms. As part of this work, I am taking sequential implementation of matrix vector computations that are the most demanding aspect of the Convolutional Neural Network (CNN) algorithms and implementing them on flash memories, using Ohm's equation and Kirchhoff's current law in the analog domain where the flash memories hold the corresponding weights for the convolutional layer of the CNN.

17. I am a technical expert in the subject matter areas relevant to this matter, including my expertise on cache architectures and cache coherency in

multicore processors. I developed a coherence-aware formal verification methodology that models realistic cache systems—including their queues, message flows, and protocol state machines—and uses invariants to mechanically prove correctness properties (like absence of deadlock) in complex multicore cache hierarchies. I extended prior verification frameworks to model realistic cache designs and coherence behaviors, particularly Modified Invalid (MI) coherence protocols, cache-to-cache transfers, DMA interactions, and queue-based interconnect constraints. As part of this work, I evaluated two extensive case studies: the verification of a directory-based MI protocol in a 2D mesh and the verification of a ring-based snoopy protocol in a 2D torus.

III. LEGAL STANDARDS

18. I have been informed by counsel of general principles concerning patent law and the analysis of whether a patent claim is invalid. Below I set forth my understanding of the legal principles that may be relevant to my opinions in this Declaration.

19. I understand that a party challenging a patent's patentability in an IPR proceeding bears the burden of proving invalidity by a preponderance of evidence.

20. I have been informed and understand that, under U.S. patent law, a patent may not issue "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 at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.” This has typically been referred to as the obviousness inquiry. I have been informed and understand that the obviousness inquiry requires that the claimed invention be considered as a whole and that in determining the differences between the prior art and the claims, the question is not whether the differences themselves would have been obvious, but whether the claimed invention as a whole would have been obvious.

21. I have been informed and understand that obviousness is a question of law based on underlying factual determinations, including: (1) the scope and content of prior art; (2) differences between prior art and claims; (3) the level of ordinary skill in the art; and (4) secondary considerations (or objective indicia) of non-obviousness, such as commercial success, long-felt but unmet needs, failure of others, unexpected results, and expert skepticism.

22. I have been informed and understand that a party asserting that a patent is obvious must demonstrate that a skilled artisan would have had reason to combine the teaching of the asserted prior art references to achieve the claimed invention, and that the skilled artisan would have had a reasonable expectation of success in doing so. I have been informed and understand that reasonable expectation of success is assessed as of the date of the invention described in the

patent. That the inventors were ultimately successful is irrelevant to whether one of ordinary skill in the art, at the time the invention was made, would have reasonably expected success. Any finding to the contrary represents impermissible use of hindsight— using the inventors’ success as evidence that the success would have been expected.

23. I have been informed that, as part of determining the scope and content of the prior art, it may be appropriate to consider whether there is evidence of a “teaching, suggestion, or motivation” to combine the prior art teachings in the prior art, the nature of the problem or the knowledge of a person having ordinary skill in the art. The question is not simply whether the prior art teaches the particular element of the invention. Knowledge of the goal does not render its achievement obvious.

24. I have been informed that one indicator of non-obviousness is when prior art “teaches away” from combining certain known elements. A prior art reference teaches away from the patent’s particular combination if it leads in a different direction or discourages that combination, recommends steps that would not likely lead to the patent’s result, or otherwise indicates that a seemingly inoperative device would be produced. If the prior art indicated that the invention would not have worked for its intended purpose or otherwise taught away from the invention, the invention is not obvious.

25. I have been informed that a patent that is composed of several elements is not made obvious merely because each of those elements was independently known in the prior art. Rather, the useful inquiry is whether there was a reason that would have prompted a person having ordinary skill in the art to combine the elements in the manner of the new invention. Thus, the challenger must show that at the time of the invention a person of ordinary skill in the art would have been motivated to combine the references, not just that the references could have been combined.

26. I have been informed that one must remain vigilant of the distortion caused by hindsight bias and must be cautious of arguments reliant upon *ex post* reasoning. The selective hindsight combination of references that show various elements of the claim generally does not suffice to establish obviousness.

IV. A PERSON OF ORDINARY SKILL IN THE ART (“POSITA”)

27. I understand that Petitioner has proposed, and the Board has agreed, at least preliminarily, that the level of ordinary skill in the art is someone with “(1) at least a bachelor’s degree in electrical engineering, computer engineering, physics, or a similar field, (2) approximately two years of industry experience in a field related to microelectronic architecture design, including experience with cache management in multicore systems-on-a-chip (SoCs)” Ex. 1003, ¶¶ 50-51. I have

adopted this definition of a person of ordinary skill in the art (“POSITA”) in connection with my analyses.

28. At the time of the invention recited in the ’228 patent, I met and exceeded this definition of a POSITA.

V. OVERVIEW OF THE ’228 PATENT

29. The ’228 patent describes a solution for particular “problems [that] arise once an IO component is integrated on the same chip with a multiprocessor.” Ex. 1001, 1:19-20. In such circumstances,

Traditional IO integration treats the IO component as a separate caching agent, meaning that dedicated logic is associated with the IO component to handle cache coherency operations. When an IO agent is performing read/write operations to main memory, it has to snoop the CPU side cache to maintain cache coherency. In MP systems, this becomes a major scaling problem.

Id., 1:20-28. The solution described in the ’228 patent employs “a CPU caching agent [] configured to support both CPU traffic and IO traffic as well, thereby resolving the scaling issue.” *Id.*, 1:56-58; *and see id.*, 2:16-26; 2:40 – 3:12. The single caching agent can be “distributed such that each of the different portions of the caching agent can be associated with a corresponding core and LLC [(last level cache)] bank or slice.” *Id.*, 3:1-3; *and see id.*, 6:20-39.

30. The use of a single caching agent to support both CPU and IO traffic is recited in independent claim 1 of the '228 patent:

An apparatus comprising:

a multicore processor including a plurality of cores, a shared cache memory, an integrated input/output (IIO) module to interface between the multicore processor and at least one IO device coupled to the multicore processor, and a caching agent to perform cache coherency operations for the plurality of cores and the IIO module, *the caching agent a single caching agent for the multicore processor* and including a plurality of distributed portions each associated with a corresponding one of the plurality of cores.

Id., 7:36-46 (emphasis added).

31. Figure 2 of the '228 patent illustrates an example of the recited “caching agent:”

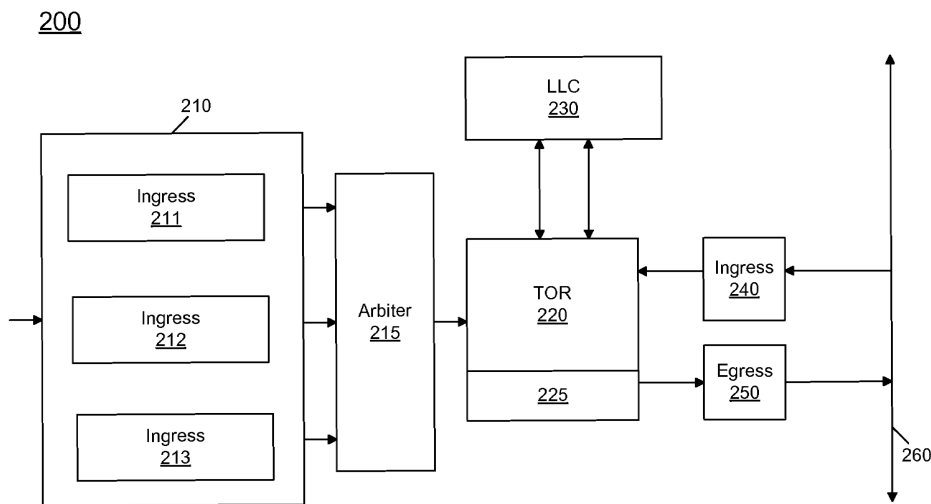


FIG. 2

According to the specification:

As shown in FIG.2, caching agent 200 may be logic interposed between one or more cores of a processor and a LLC 230. In addition, caching agent 200 may also provide an interface between an IIO module within a multi core processor and the LLC. As seen, caching agent 200 can include an ingress port 210 that includes multiple ingress queues 211-213. As seen, ingress port 210 may be coupled to an arbiter 215 which can arbitrate amongst incoming requests and provide them to a table of requests (TOR) 220. As seen, TOR 220 may be a buffer or other temporary storage for holding incoming requests. In the embodiment shown, TOR 220 may include at least one dedicated resource 225 for posted transactions. In one embodiment, dedicated resource 225 may be a single

entry of the queue, although the scope of the present invention is not limited in this regard.

Still referring to FIG. 2, TOR 220 interfaces with LLC 230. More specifically, this LLC may be a bank or other portion of the LLC associated with the caching agent. To provide access to other locations within the processor via a ring interconnect 260, an ingress queue 240 and an egress queue 250 may be provided. Thus via caching agent 200, cores of the processor and an IIO module can maintain coherency without the need for additional caching agents. Although not shown for ease of illustration, understand that caching agent 200 may further include various coherency logic to perform cache coherency operations on behalf of the cores and IIO module, and to enable coherent access with regard to both the core transactions and IO device transactions as described herein.

Id., 2:35-65.

A. Construction of a “Single Caching Agent”

32. I understand that when reviewing the patentability of a claim, the analysis requires that the claim must first be construed. And, I understand that, absent any specifically defined terms from the specification, the claims should be

construed consistent with their plain and ordinary meaning as understood by a person of ordinary skill in the art when read in the context of the specification and prosecution history of the patent. In forming my conclusions concerning the patentability of the claims I applied such constructions.

33. I understand the Board has preliminarily construed the “single caching agent” recited in claim 1) as “one caching agent for the processor, which may be formed from distributed portions.” Paper 15, Institution Decision, at p. 15. I agree a single caching agent means one caching agent for the processor. But I emphasize that this one caching agent is a logically unified entity, and would be understood as such by a person of ordinary skill in the art. By “logically unified entity,” I mean a single, cohesive caching agent where all distributed portions are interconnected, consistent, and function together in a rational or systematic manner, not a collection of independent caching agents that merely communicate with one another. This understanding is emphasized in the specification, which teaches that the *one* caching agent, irrespective of any distributed portions, acts for *all* the cores and the IIO module. Ex. 1001, 2:18-20 (“this caching agent may be *a combined caching agent* both for the CPU as well as for an integrated IO agent”); 2:58-60 (“[V]ia caching agent 200, cores of the processor and an IIO module can maintain coherency *without the need for additional caching agents.*”); 6:25-29 (“Note that while distributed caching agents are shown, understand that these distributed

portions *form a single caching agent*, and which is configured to handle cache coherency operations both for the cores as well as an IIO module 750.”); 7:5-7 (“*a single caching agent* shared by the IIO module and the cores of the processor.”) (emphases added).

VI. THE PETITION FAILS TO DEMONSTRATE THAT ANY CHALLENGED CLAIM IS UNPATENTABLE

34. I understand that the petition presents two groups of challenges to the claims of the '228 patent; one based primarily on the teachings of Agarwal (Ex. 1003), and the other based primarily on the teachings of Sinharoy (Ex. 1004):

Ground		References	Claims
1	§ 103	Agarwal ¹	1, 2, 4-7, 11-14
2A	§ 103	Sinharoy ²	1, 4, 6, 7, 11-14
2B	§ 103	Sinharoy in view of Blumrich ³	2, 3, 16-18
2C	§ 103	Sinharoy in view of Sistla ⁴	5

¹ Agarwal et al., U.S. Patent No. 7,805,575 (Ex. 1005).

² B. Sinharoy et al., IBM POWER7 Multicore Server Processor, IBM J. Res. & Dev., v. 5, no. 3 (May/June 2011) (Ex. 1006).

³ Blumrich et al., US PGPUB 2011/0219188 (Ex. 1007).

⁴ Sistla, et al., US PGPUB 2008/0216750 (Ex. 1008).

2D	§ 103	Sinharoy in view of Sistla and Blumrich	19
----	-------	--	----

Pet., 5.

A. The Challenged Claims are Not Unpatentable in View of Agarwal

1. Overview of Agarwal

35. Agarwal describes a tiled multicore processor architecture in which each “tile” includes a processor core, a cache, and a directory controller responsible for managing cache coherence for a specific region of memory. Ex. 1005, 3:7-14, 21-24. The system implements a distributed directory protocol, where each directory controller operates autonomously for its assigned memory region. *Id.*, 3:39-41, 55-61; 4:9-12; 40-45. The assignment of memory regions to directory controllers is accomplished through the concept of a “home tile.” For any given memory address, a deterministic function (such as a hash or address mapping table) designates a particular tile as the “home tile” for that address. The directory controller on the home tile is solely responsible for maintaining coherence for its assigned memory region. *Id.*, 4:40-45, 55-57; 17:42-57; 23:51–53.

36. When a processor core or an IO module wishes to access a memory location, the request is routed to the home tile for that address. The directory controller on the home tile examines its directory state to determine the appropriate

coherence action—such as supplying data, invalidating other caches, or updating state. Each directory controller acts independently for its own region. *Id.*, 16:25-45; 19:3 – 20:25 (discussing Figs. 6A & 6B); 27:46 – 28:8.

37. Agarwal extends this protocol to IO transactions through its “ProxyIO” coherence mechanism. *Id.*, 27-44-46. When an IO module (integrated on the multicore chip) initiates a transaction—such as a DMA write or read—the IO module uses the same address mapping logic to determine the home tile for the target memory address. The IO module then sends the transaction directly to the home tile’s directory controller, which processes the IO request in the same manner as a request from a processor core. The directory controller updates its directory state, manages cache lines, and ensures coherence for the memory region under its control. The IO module itself does not participate in coherence management beyond initiating the transaction; all coherence actions are handled by the home tile’s directory controller. *Id.*, 29:5 – 30:4.

38. In summary, Agarwal’s architecture is characterized by a set of distributed and independent directory controllers, each managing coherence for a distinct memory region as the “home tile,” and handling both processor and IO module transactions for their assigned regions.

2. The Single Caching Agent: Limitation of Claim 1 is Not Disclosed or Suggested by Agarwal

a) ARM Conflates Agarwal's "Distributed Directory Controllers" with the Claimed "Single Caching Agent"

39. Claim 1 requires “a single caching agent ... including a plurality of distributed portions each associated with a corresponding one of the plurality of cores.” Ex. 1001, 7:36-46. The '228 patent specification makes clear that this is not a mere collection of independent cache controllers, but a logically unified agent with distributed portions, each tied to a core, that performs coherency for both the cores and the IIO module. *Id.*, 2:66 – 3:3 (“Note that although a single structure is shown in FIG. 2 for ease of illustration, understand that a caching agent can be distributed such that each of different portions of the caching agent can be associated with a corresponding core and LLC bank or slice.”); 6:25–29 (“[T]hese distributed portions form a single caching agent, and which is configured to handle cache coherency operations both for the cores as well as an IIO module 750.”); 7:5-7 (“[A] single caching agent shared by the IIO module and the cores of the processor.”).

40. The specification of the '228 patent explains that “caching agent 200 may be logic interposed between one or more cores of a processor and a LLC 230. In addition, caching agent 200 may also provide an interface between an IIO module within a multicore processor and the LLC. ... Thus via caching agent 200,

cores of the processor and an IIO module can maintain coherency without the need for additional caching agents. *Id.*, 2:36-65. By employing the “single caching agent” a separate, dedicated IO caching agent is avoided. Instead, “the IIO module proxies through the CPU caching agent to access memory or other IO devices.” *Id.*, 3:4-5. This approach also “reduces the amount of snoop traffic needed since a reduced number of caching agents per system can be realized.” *Id.*, 3:8-9.

41. Agarwal’s Proxy Cache Coherence (“PCC”) approach, by contrast, involves a system where each tile (core) has its own directory controller and directory storage, operating independently for their assigned memory regions. Ex. 1005, 16:30-38; 18:1-10. There is no teaching or suggestion in Agarwal that these controllers are logically unified into a “single caching agent” as required by claim 1. Instead, Agarwal’s architecture is an example of a distributed directory protocol, where each directory controller is responsible for its own partition of memory, and there is no central or logically unified agent managing coherency for the entire multicore processor. *Id.*, 16:26–45, 23:51–53.

42. Petitioner’s and Dr. Hassoun’s assertion that Agarwal’s distributed directory controllers “collectively” manage coherence, *see* Ex. 1003, ¶ 64, does not satisfy the claim’s requirement for a single caching agent. Dr. Hassoun testified in his deposition that the controllers are “operating collectively, but they’re doing different functions as they need to. ... But collectively, they perform the cache

coherency operation.” Ex. 2007, 83:5–10. This “collective” action that Dr. Hassoun assigns to the directory controllers is nothing more than message passing. Agarwal explains that in the event of a cache miss on a tile, the “miss goes as a message through the network to the home tile.” Ex. 1005, 19:11-12. The message encodes various information, but it is the directory controller at the home tile—that is, the tile for which its associated directory controller has responsibility—that is responsible for cache coherence operations with respect to the associated address. *Id.*, 19:30-38. There is no “collective” cache coherence operation. And, inasmuch as each home tile’s directory controller is performing the same actions for its assigned memory region, there is no “single caching agent” that is performing cache coherency operations *for the plurality of cores*—a distinction critical to claim 1.

43. The “single caching agent” required by the ’228 patent is a logically unified entity—irrespective of any distributed portions—that acts as one agent for all cores and the IIO module. Ex. 1001, 2:66–3:3, 6:22–29. In Agarwal’s system, while the individual directory controllers may interact via protocol-level messaging, they are not unified with one another as a single agent. Instead, each directory controller is autonomously responsible for its own assigned memory region, making independent coherence decisions for that memory region. Ex. 1005, 16:26–45; 19:7 – 20:25; 23:51–53. The controllers are not unified as a single entity

that makes coherency decisions for all of the cores (tiles); rather the directory controller of each home tile manages its own assigned memory region independently. The distinction is between Agarwal's approach, which relies on collaboration among peer tiles, and an alternative approach that enforces coherence through a single, unified control point.

44. Put differently, in Agarwal the actions of the different home tile directory controllers (as independent agents) involve a group of autonomous entities operating according to their own decision-making processes. They interact through messages passed to one another, but they do not operate as a single unit. The system's outcome emerges from these individual operations. A single agent with distributed portions, in contrast, refers to one unified entity whose components are spread out but operate under centralized control. The "portions" work in concert under a common directive. By way of analogy, imagine a company with several independent managers. Each runs their own department ("collective" management) while periodically informing the others about their decisions. Contrast this with a single manager with deputies in each department ("single agent with distributed portions"). The former is not the latter, even if the individual department managers inform one another of some of their activities.

b) The Association of Distributed Portions with Cores Is Not Disclosed

45. Claim 1 requires that each distributed portion of the single caching agent is “associated with a corresponding one of the plurality of cores.” Ex. 1001, 7:44-45. The petition argues, “Agarwal teaches ‘[a]t least some of the directory controllers are associated with a processor core. . . . Similarly, Agarwal teaches ‘[e]ach of at least some of the processor cores includes a directory controller...’ and ‘at least some of the storage for directory state.’ Each directory controller 603 / directory storage 604 pair (‘distributed portion [of the caching agent]’) is ‘associated with’ a processor 600 within each tile (i.e., ‘corresponding one of the plurality of cores’).” Pet. at 23-24 (citations omitted). In other words, the petition argues that because each tile’s directory controller is architecturally paired with a core, this constitutes the “association” recited in the claim. But this argument fails.

46. Agarwal’s directory controllers serve memory regions, not cores. *See* Ex. 1005, 19:7-29. The assignment of memory regions to directory controllers is determined by a hash or mapping function on the memory address, not by core identity. Any core or IO module can generate a request for any memory address, and the request is routed to the home tile for that address. As a result, a directory controller on Tile X handles requests originating from any core, including cores other than Tile X, depending on memory access patterns and address mapping. The controller’s responsibility is to the memory region, not to the core on its tile. *Id.*,

16:31-45; 17:50-51; 23:51-53. This is true even in Agarwal’s shim cache coherence (“SCC”) embodiment. *Id.*, 14:60 – 15:12.

47. Contrast this with the ’228 patent’s association between *cores* and portions of the caching agent. Claim 1 explicitly provides that each portion of the caching agent is associated with one of the plurality of cores. Ex. 1001, 7:44-46. Given this context, a POSITA would understand that the distributed portion is not merely co-located with a core, but as the component that performs cache coherency operations for that core. And it is a one-to-one correspondence between portion of the caching agent and a core. The claim language ties each “distributed portions” to the *function* of performing cache coherency operations for its associated core. The specification reinforces this, stating that “a caching agent can be distributed such that each of different portions of the caching agent can be associated with a corresponding core and LLC bank or slice,” and that these distributed portions “form a single caching agent ... configured to handle cache coherency operations both for the cores as well as an IIO module.” *Id.*, 2:66–3:3, 6:22–29. This one-to-one functional association does not exist in Agarwal, which instead describes one tile’s directory controller performing cache coherency operations for a request originating at another tile’s core. Ex. 1005, 19:7-38.

48. A practical example based on Agarwal illustrates the distinction: Suppose a core A on Tile 1 accesses a memory address whose home tile is Tile 5.

The directory controller on Tile 5, not Tile 1, manages coherence for that address, even though it was a core on Tile 1 that initiated the request. The same is true if a core B on Tile 2 accesses that same memory address. Once again it is the directory controller on Tile 5 that manages coherence for the address. In no meaningful way, in the context of claim 1, is the directory controller on Tile 5 associated with *a core*, as it must manage coherence for requests made by any and all cores of the multiprocessor whenever those cores access one of the directory controller's assigned memory addresses. Accordingly, in my opinion, Agarwal does not teach or suggest at least this "association" requirement of claim 1.

c) The "Single Caching Agent" Must Perform Coherency for Both Cores and the IIO Module

49. Claim 1 requires that the single caching agent perform coherency operations for both the cores and the IIO module. Ex. 1001, 7:36-46. Agarwal's "ProxyIO" extension allows an I/O module to participate in the directory protocol, but this is accomplished by sending requests to the appropriate directory controller(s) for the relevant memory region(s). There is no single agent that manages coherencies for all the cores and the I/O module; rather, each directory controller manages its own assigned memory region, whether the request comes from a core or an I/O module. Ex. 1005, 29:5-28.

50. According to Agarwal, in an extension of the PCC approach, called ‘ProxyIO coherence,’ a home tile—that is, a directory controller of a home tile—manages cache coherence in the presence of an I/O module (‘I/O coherence’). *Id.*, 27:44-46; 29:5-31. Coherence processing at the home tile for memory access requests from an I/O module includes a set of actions that are similar to those for PCC. *Id.* Agarwal’s Figure 9C illustrates an I/O module sending a write transaction to the home tile, which then performs coherence processing.

51. As I explained above, each home tile’s directory controller is an independent agent, responsible only for its assigned memory region. Agarwal’s architecture is intentionally decentralized in this regard. Consequently, when a home tile’s directory controller manages coherence for its assigned memory region, even when the request comes from an I/O module, other directory controllers are not involved in the coherence operation. This means that for requests by the I/O module that implicate different memory regions, coherency operations associated with those requests will be handled, autonomously, by different directory controllers according to the respective memory region assigned to each directory controller. In short, there is no single agent responsible for coherency for all cores and the I/O module together.

d) A POSITA Would Not be Motivated to Modify Agarwal to Arrive at the Claimed Invention

52. Even if Agarwal's distributed directory controllers could be considered "portions" of a single agent (which they cannot), the petition does not provide a reasoned motivation for a POSITA to modify Agarwal's architecture to arrive at a logically unified single caching agent with distributed portions as claimed. Any assertion that "it would have been obvious" to treat Agarwal's distributed directory controllers as a single agent is unsupported by technical rationale or evidence.

53. In a tile-based system like Agarwal's, it would neither be practical nor possible to implement a "single caching agent" as recited in the '228 patent without fundamentally changing the architecture. The reasons for this are rooted in scalability, physical distribution, and the design philosophy of tiled multicore systems.

54. Physical distribution and scalability: Tile-based systems that employ directory-based cache coherence approaches, such as Agarwal's, are expressly designed for scalability and parallelism. *See id.*, 2:21-24. Each tile (core) is a self-contained unit with its own local resources, including cache and directory controller. Introducing a single, centralized caching agent would create a bottleneck as the number of tiles increases, undermining the main advantage of the specified approach. Moreover, routing all cache coherence traffic routed through a

single agent would introduce significant wiring complexity and latency, especially as chip sizes grow.

55. Distributed directory protocols are fundamental to Agarwal: Agarwal's PCC protocol is built on the principle that each tile is responsible for coherence for a subset of memory (its "home region"). *Id.*, 4:9-12, 40-45; 7:65 – 8:13. This localizes coherence operations, reduces global communication, and enables the system to scale. A single caching agent, by contrast, would necessarily require all coherence operations—reads, writes, invalidations, etc.—to be funneled through one logical entity, which is fundamentally at odds with the distributed, parallel nature of tiled systems.

56. Logical vs. physical unity: The '228 patent's "single caching agent with distributed portions" is a logically unified agent, even if physically distributed. In Agarwal, the system is made up of independent directory controllers, each acting autonomously for its assigned memory region. To implement a true "single caching agent" in Agarwal's system would require unifying all directory controllers, which would necessitate architectural changes not taught or suggested by Agarwal.

57. Performance and power: Routing all coherence traffic through a single agent would create a performance bottleneck and increase power consumption due to longer communication paths and higher contention.

e) Sharing of Cache Lines in Agarwal Does Not Require Logical Coordination of Caching Agents

58. Sharing of cache lines across tiles in Agarwal does not mean there is logical unity or a “single caching agent” present. While such sharing does require protocol-level communication between Agarwal’s independent directory controllers, that is not logical coordination in the sense of a single caching agent as recited in claim 1.

59. Distributed Directory Protocols: Local Autonomy, Protocol Communication: In Agarwal, each home tile’s directory controller is solely responsible for coherence decisions for its assigned memory region. *Id.*, 3:39-41; 4:9-12; 23:51-53. When a cache line is shared (i.e., cached in multiple tiles), the home tile’s directory controller tracks which tiles have copies (using pointers, sharer lists, or bits). *Id.*, 4:19-22; 14:33-36; 16:58-66. If another tile wants to write to a shared line, it must communicate with the home tile’s controller, which then issues invalidations or updates as needed. *Id.*, 22:27-29.

60. Protocol Communication $\neq$ Logical Unity: The controllers exchange messages as part of the coherence protocol (e.g., sending invalidates, acknowledgments), but each acts autonomously for its region. *Id.*, 19:7 – 20:25. There is no single caching agent maintaining global state and no single point of

cache coherence operations—each controller acts independently for its own allocated memory region. *Id.* This is fundamentally different from a “single caching agent,” whether with distributed portions or otherwise, which describes a unified logic and global view. *See, e.g.,* Ex. 1001, 2:36-65.

61. No Unified Coherency for All Cores and IIO: Even when cache lines are shared, all coherence actions (e.g., invalidation, update) for a given line are managed by the home tile’s controller for that line. Other directory controllers are not involved in coherence decisions for that line—they only participate for lines for which they are the home. Ex. 1005, 19:7 – 20:25. Agarwal describes the protocol as “distributed,” with each home tile acting as a “proxy” for its assigned region. *Id.*, 7:65 – 8:28. There is no suggestion that the controllers are just portions of a single agent; rather, they are independent agents.

62. To better appreciate this distinction, consider of a group of independent referees, each assigned to a different playing field. If a player moves from one field to another, the referees may communicate about the player’s status, but each referee is only in charge of their own field. There is no “single referee” for all fields—just a protocol for communication between referees. In Agarwal, sharing of cache lines requires similar message-passing between the independent directory controllers, but this does not create logical unity or a single caching agent. Each controller is autonomous for its region. Communication is necessary

for correctness but does not amount to unified control or logic. This is fundamentally different from the “single caching agent” of the ’228 patent.

f) The Protocol-Driven Cooperation Among Directory Controllers in Agarwal Does Not Mean They are Parts of a Single Caching Agent

63. Protocol-driven cooperation in Agarwal reflects distributed autonomy, not the logical unity of a single caching agent as recited in claim 1.

64. In Agarwal, directory controllers cooperate by following a distributed coherence protocol, e.g., sending invalidation messages, acknowledgments, etc. *Id.* Dr. Hassoun describes this as a “collective” approach to coherency operations that establishes Agarwal as teaching a single caching agent. *See* Ex. 1003, ¶ 64. In my opinion, this is not correct.

65. The communication that is present between Agarwal’s directory controllers is necessary for correctness, but it is message passing, not centralized control. Said differently, Agarwal’s coherence protocol ensures system-wide consistency of memory, not logical unity of controllers. Each directory controller operates autonomously for its assigned memory region. The protocol simply defines how these independent agents interact. It does not somehow transform these autonomous agents into a single, unified decision-making entity.

66. If anything, Agarwal’s design philosophy is one of “distributed coordination.” Each home tile’s directory controller, and only that controller, makes coherence decisions for its assigned memory region. Ex. 1005, 19:7 – 20:25. When a memory line is assigned to a tile (e.g., via a hash function) the tile’s controller is fully responsible for coherence of that line. Other tiles do not participate in the decision-making for that line; they only send and/or receive protocol messages when necessary. In other words, the controllers do not share a common control structure or act as portions of a single entity. They are independent agents. When they communicate with one another they do so as peers, not as subcomponents of a unified agent. This philosophy emphasizes scalability and parallelism. It avoids bottlenecks inherent in a centralized control scheme and allows each tile to operate independently for its region.

67. Contrast this with the ’228 patent. The ’228 patent requires a single caching agent with distributed portions, meaning both logical unity (one agent, even if physically distributed) and unified control (distributed portions acting under unified logic). Think of the distinction like this:

68. In one scenario (the ’228 patent methodology), a master scoreboard in a sports tournament shows all teams’ scores (global state). This means decisions (e.g., ranking, tie-breaks, etc.) can be made centrally because all of the necessary information is aggregated in one place. This mirrors the ’228 patent’s single

caching agent with distributed portions which maintains coherence across all cores and the IIO module as one logical entity.

69. In a second scenario (Agarwal, distributed coordination) each team only knows its own score and communicates it (or, more generally, updates regarding the score) occasionally. There is no single “master” scoreboard. The messages exchanged in the second scenario do not create or imply a central authority or a unified global state. Instead, there are only “local” scoreboards, and the messages serve to maintain correctness through distributed consensus, where each scoreboard updates its own local state. This reflects Agarwal’s independent directory controllers, which cooperate via a distributed protocol but remain autonomous for their assigned memory regions.

70. The ’228 patent states that distributed portions “form a single caching agent” configured to handle coherencies for all cores and the IIO module. *See, e.g.*, Ex. 1001, 6:22–29. This means that there is one logical entity that knows the complete coherence status of every memory line across all participants. Decisions about cache coherence are made with awareness of the entire system’s state. Agarwal never suggests such unity. Each directory controller only knows about its own memory region. There is no single agent or mechanism that aggregates coherence information across all tiles. Each controller acts independently, and system-wide consistency is achieved through protocol-driven messaging, not

centralized or unified logic. While Agarwal’s protocol ensures a unified *view of memory*, that does not make the controllers parts of a single logical agent. They remain independent entities following common rules, not portions of one agent.

71. For all of the above reasons, it is my opinion that the petition fails to establish that Agarwal discloses or renders obvious the “single caching agent ... including a plurality of distributed portions each associated with a corresponding one of the plurality of cores” required by claim 1 of the ’228 patent. Agarwal’s distributed directory controllers are not a “single caching agent,” and there is no teaching, suggestion, or motivation to modify Agarwal to arrive at the claimed invention.

3. Claim 2 is Not Obvious in View of Agarwal

72. Claim 2 reads:

“The apparatus of claim 1, wherein the caching agent includes a table of requests to store pending requests for the plurality of cores and the IIO module.”

Ex. 1001, 7:47-49.

73. I understand that because of its dependency from claim 1, all of the above reasons for claim 1’s patentability over Agarwal apply equally to claim 2.

74. The recited “table of requests” (TOR) within the caching agent, which stores pending requests for both the cores and the IIO module, is described as a buffer or temporary storage for holding incoming requests from both the cores and the IIO module and is part of the single caching agent’s logic for managing and arbitrating requests. *Id.*, 2:44–53, 2:66 – 3:3.

75. The petition argues that Agarwal discloses or renders obvious the recited “table of requests” because Agarwal’s directory controller in each tile maintains a buffer or queue of outstanding transactions. Pet., 24–25; Ex. 1003, ¶¶ 94–97, 204–213. The petition also references Agarwal’s mention of a “small buffer that contains a list of outstanding transactions with an acknowledgement counter associated with each [outstanding transaction].” Pet., 24 (citing Agarwal, 27:21–25).

a) The “Table of Requests” Limitation Is Not Disclosed or Suggested by Agarwal

1. The ’228 Patent’s “Table of Requests” (TOR) Is a Specific, Centralized Structure

76. The ’228 patent describes the TOR as a centralized structure within the caching agent that stores pending requests from both the plurality of cores and the IIO module. Ex. 1001, 2:38–53; Fig. 2, els. 220, 225. Understanding the TOR as a “centralized structure,” is consistent with the patent’s description of the TOR

as a single, shared resource within the caching agent and not a collection of separate, per-core or per-module buffers. As illustrated in Figure 2, the TOR is a single, unified table within the caching agent, downstream of the arbiter (215) and upstream of the LLC (230), that arbitrates and manages requests from both the processor cores and the IIO (integrated input/output) module.

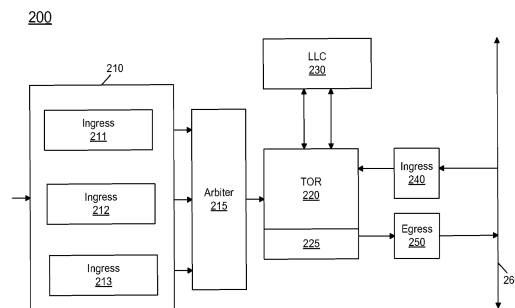


FIG. 2

All requests from the cores and the IIO module are funneled through this structure. Claim 2 makes clear the nature of this shared resource and the claim does *not* say “each core includes a table of requests” or “each portion [of the single caching agent] includes its own table.” Instead, it says *the caching agent* (singular) includes *a table of requests* (singular) for all sources, which implies a unified, centralized structure within the caching agent that stores all incoming requests.

2. Agarwal’s Buffers Are Not the Claimed “Table of Requests”

77. Agarwal’s “small buffer” is described as a per-tile structure that tracks outstanding transactions for that tile’s directory controller. Ex. 1005, 27:21–25.

There is no disclosure in Agarwal of a centralized table or buffer that stores pending requests from both all the processor cores and the I/O module, nor is there any suggestion that the per-tile buffers are shared resources in the manner required by claim 2.

78. Agarwal's architecture is fundamentally distributed: each tile manages its own directory state and outstanding transactions. *See, e.g., id.*, Fig. 2, Fig. 6A/B, and the accompanying text at 16:26–45; 19:7 – 20:25; 27:21–25. There is no teaching or suggestion of a single, centralized table (within a caching agent or otherwise) that stores pending requests from both the plurality of cores and the I/O module. The buffers described are local to each tile and are not shared among tiles or between cores and I/O modules, and the architecture is explicitly described as distributed, with each tile responsible for its own directory and transaction management.

3. The Petition Conflates Distinct Structures

79. The petition and Dr. Hassoun equate Agarwal's per-tile buffer with the '228 patent's TOR, but this is a misreading.

80. The Background section of the '228 patent notes that in systems where both cores and I/O modules issue requests, deadlocks can occur if posted requests (e.g., writes) are blocked by non-posted requests (e.g., reads), especially

when both types of requests are funneled through a shared resource. Ex. 1001, 19-30. The invention introduces the TOR within the caching agent, and it includes at least one dedicated resource (queue entry) for posted transactions. *Id.*, 2:44–53. By reserving a dedicated entry for posted requests, the TOR ensures that posted requests are not blocked by non-posted requests in the queue. This prevents deadlocks that would otherwise occur if a posted request (which must proceed for the system to make progress) is stuck behind non-posted requests. The TOR, as part of the caching agent, thus enforces the required ordering and resource reservation to comply with PCIe and similar protocols.

81. In Agarwal’s architecture, each tile (core) has its own directory controller, directory storage, and buffer for outstanding transactions. Ex. 1005, 16:34-45. There is no centralized queue or table that receives requests from all cores and I/O modules. Each tile manages its own requests and directory state independently. The petition’s argument that Agarwal’s per-tile buffer is equivalent to the ’228 patent’s TOR ignores the requirement that the TOR is a shared structure for *all* sources, not a local buffer. There is no teaching or suggestion in Agarwal of a unified table that stores requests from all cores and I/O modules.

b) The Petition’s Reliance on “Obviousness” Is Misplaced

1. “Obvious to Include a Buffer” Is Not Enough

82. The petition argues that it would be “obvious” to include a buffer or queue for outstanding requests. Pet. 24. But this is not the same as the claimed “table of requests” that is shared, centralized, and arbitrates among both core and IIO requests. In my opinion, the mere existence of buffers in prior art does not render the specific, centralized, multi-source arbitration structure of the ’228 patent obvious.

83. In computer architecture, it is common and well-known to use buffers or queues to track outstanding requests—each processor, core, or module may have its own local buffer for this purpose. These are typically simple, per-source structures. Indeed, that is precisely what Agarwal describes. Ex. 1005, 27:21–25. But the “table of requests” in the ’228 patent is not just *any* buffer. It is a specific, shared structure within the caching agent that is designed to store pending requests from both the plurality of cores and the IIO module. It is not per-core or per-module, but a unified table. It is part of the caching agent, not just a generic architectural feature. And it is structured to handle requests from multiple sources in a unified way. Ex. 1001, 2:35-65.

84. I am advised that obviousness requires that the prior art, either alone or in combination, would have led a POSITA to the claimed invention as a whole—not just to a generic concept (like “use a buffer”), but to the specific structure and function claimed. The ’228 patent’s TOR is designed (as explained in

the context of the invention) to address system-level problems such as deadlock and ordering between posted and non-posted requests from multiple sources. A generic buffer or queue for outstanding requests does not, by itself, solve these problems or provide the same system-level coordination.

2. No Evidence of a Need for or Benefit of a Unified Table Handling Both

85. Agarwal's architecture does not require, suggest, or benefit from a unified table of requests as claimed.

86. First, Agarwal's architecture is fundamentally distributed. Each tile (core) in Agarwal's system has its own directory controller, directory storage, and buffer for outstanding transactions. *Id.*, Figs. 2, 6A/B; 16:38–45 (“A directory storage 604 for directory state can be included within the same memory that provides the cache 602, or can be included in a separate memory module on the tile. ... A directory controller 603 is coupled to the directory storage 604.”). Outstanding requests are tracked locally. *Id.*, 27:21–25 (“The tile can also keep a small buffer that contains a list of outstanding transactions along with an acknowledgement counter associated with each.”). Such an architecture does not require or suggest a unified table. Because each tile is responsible for its own memory region (Agarwal's “home tile” concept), and each tile only manages requests for its own assigned memory region, there is no architectural need for a

single, unified table that tracks requests from all cores and I/O modules. Requests are routed directly to the appropriate tile based on address mapping (hash function, TLB, etc.), and each tile processes only those requests relevant to its assigned memory region. As such, Agarwal would not benefit from a unified table as it would only introduce scalability issues. Each tile can already process requests independently and in parallel, maximizing throughput. Having to route requests through a centralized table would defeat the very purpose of this distributed approach that allows the system to scale to many cores. A POSITA would not deliberately handicap the system in such a fashion.

3. The Petition's Use of Secondary References (e.g., Blumrich) Is Misguided

87. The petition argues a buffer would be beneficial and cites Blumrich for the idea of a request queue. Pet., 25. But even if such a buffer/queue were added to operate Agarwal's directory controller, that still would not meet the requirements of claim 2.

88. A request queue (buffer) as in Blumrich is typically used to track requests for a particular directory controller or cache and is not inherently a shared structure for multiple sources. According to Blumrich, request queue 302 is attached to each L2 slice, so it is not centralized. Ex. 1007, [0128], [0130]. Hence, if a POSITA adopted a buffer as described by Blumrich to modify Agarwal's

directory controller, each tile in Agarwal would have a buffer for tracking outstanding requests handled by that tile's directory controller. While such a buffer might help that tile manage its own requests (from its own core or from remote sources directed to that tile), the buffer would still be local to each tile, there would not be a single, unified table within a caching agent that stores pending requests from both the plurality of cores and the IIO module, and the architecture would remain distributed, with each tile managing its own requests.

89. Thus, modifying Agarwal to include a per-tile buffer (even if inspired by Blumrich) does not create a shared, unified table as required by claim 2.

4. Claim 6 Is Not Obvious in View of Agarwal

90. Claim 6 reads:

“The apparatus of claim 1, wherein the caching agent is to receive an allocation transaction from the IIO module, the allocation transaction received in the IIO module from an IO device coupled to the multicore processor.

Ex.1001, 7:64-67.

91. An “allocation transaction,” in the context of the '228 patent, refers to a specific type of transaction initiated by an IO device (via the IIO module) that causes data to be directly stored (allocated) into the shared cache memory of the multicore processor, rather than first being written to system memory. Ex.1001,

3:45–58; Fig. 3. The “allocation transaction” is central to the ’228 patent’s approach for improving IO performance: it allows IO data to be placed directly in the processor’s cache, making it quickly accessible to the processor cores, and reducing memory bandwidth usage.

92. The petition argues that Agarwal discloses or renders obvious this limitation because Agarwal’s I/O module can receive requests from IO devices and forward these requests to the caching agent, which then allocates data in the shared cache. Pet., 28-30.

a) Agarwal Does Not Disclose the Required Transaction Chain

93. Agarwal describes two cases for I/O module transactions: a “memory coherence case,” in which an I/O module writes data to main memory, and a “cache coherence case,” in which an I/O module writes data directly into cache. In the memory coherence case, the process is initiated from a tile (i.e., a core), the I/O module writes to memory, and cache lines, if present, are updated accordingly. In the cache coherence case, the I/O module writes data into a home tile’s cache, and other caches and memory are updated according to the coherence protocol. Ex. 1005, 28:9–32; 29:5 – 30:11. It is the latter, “cache coherence case” that the petition equates to the “allocation transaction” of claim 6. Pet., 29-30.

94. A careful reading, however, reveals that Agarwal does not disclose the full chain of I/O device to IIO module to single caching agent required by the claim. According to Agarwal, the I/O module (912) “includes an I/O interface 913 such as a network port . . . and circuitry 915 to enable the module 912 to process direct memory access (DMA) transactions initiated from a tile, for example, or to otherwise handle a transaction to write data into off-chip main memory via a memory interface 914 (for example, into main memory location A).” Ex. 1005, 28:10–22. That is, Agarwal is directed to core-initiated DMA transactions, not transactions initiated by external IO devices. Claim 6’s requirement that the allocation transaction be “received in the IIO module *from an IO device* coupled to the multicore processor” is missing.

95. Neither the petition nor Dr. Hassoun’s declaration address this gap. Both rely on the “cache coherence case” in Agarwal, but do not explain how Agarwal’s system handles transactions from external IO devices.

5. Claim 11 and Its Dependent Claims Are Not Obvious in View of Agarwal

96. Claim 11 recites a method that includes:

[11a] “*receiving an allocation transaction in an integrated input/output (IIO) module ... from an IO*

device ... the allocation transaction ... a request to directly store data ... into a shared cache memory”

[11b] *“forwarding the allocation transaction to a caching agent ... the caching agent shared by a plurality of cores ... and the IIO module”*

[11c] *“directly storing the data of the allocation transaction into the shared cache memory”*

Ex. 1001, 8:12-23.

97. Similar to claim 6, “allocation transaction” semantics are baked into claim 11’s steps. As noted above, an “allocation transaction” is a request from an IO device, received at the IIO, that is specifically for directly storing data into the LLC, bypassing system memory. *Id.*, 3:45–58. This is more than “a write that ultimately causes allocation;” it’s a transaction type and dataflow the ’228 patent describes explicitly.

98. Petitioner points to Agarwal’s example where the I/O module delivers a write to the “home tile,” whose directory controller then does normal coherence and can place data in the L2/L3 cache. Pet., 32. But, as detailed above with respect to claim 6, Agarwal does not teach an *I/O-originated “allocation transaction”* in the ’228 patent sense. Petitioner treats any I/O write that ends up in cache as an “allocation transaction,” which over-reads the term as used in the ’228 patent.

99. Further, the “forwarding ... to a caching agent shared by a plurality of cores and the IIO module,” step [11b], is not shown in Agarwal. Even without the “single” caching agent limitation found in claim 1, claim 11 still requires forwarding the allocation transaction to “a caching agent” that is shared by multiple cores and the IIO. In the ’228 patent, the IIO proxies through the CPU caching agent—i.e., the IIO forwards the allocation transaction to that agent, which services both CPU and IIO traffic. That’s the express dataflow in Fig. 3 and its accompanying text. Ex. 1001, 3:45 – 4:31.

100. Petitioner’s mapping of Agarwal equates Agarwal’s distributed directory controller and directory storage (per-tile, address-hashed “home tile”) with the ’228 patent’s “caching agent,” asserting that sending an I/O write to the home tile counts as “forwarding ... to a caching agent” that’s “shared by a plurality of cores and the IIO module.” Pet., 33. The problem with Petitioner’s argument is that in Agarwal, the I/O module delivers the transaction to a home tile chosen by a hash; the directory controller on that tile runs coherence. Agarwal does not describe this flow as “forwarding to a caching agent” that is a shared CPU/IIO agent; it’s just normal directory-based coherence at the home tile. By contrast, the ’228 patent teaches that IIO traffic is proxied through the CPU caching agent (a shared CPU/IIO coherency agent), and the IIO explicitly forwards the allocation

transaction to that agent. Ex. 1001, 3:45 – 4:31. That proxying step (and the identity of the recipient) matters for [11b].

101. In short, Agarwal’s home-tile directory controller is not the “caching agent” contemplated by [11b], which is shared by a plurality of cores and the I/O module, and Agarwal never teaches the I/O “forwarding the allocation transaction” to such a shared agent. Moreover, claim 11 requires receiving and directly storing “the data of the allocation transaction” in a shared cache. Agarwal discusses writes reaching the cache but never teaches an I/O-originated “allocation transaction” with the specific semantics and flow the ’228 patent requires.

6. Claim 14 is Not Obvious in View of Agarwal

102. Claim 14 recites:

“The method of claim 12, further comprising directly storing the data of the allocation transaction into the shared cache memory without communicating with the system memory.

Ex. 1001, 8:33-36.

103. The claim recites a prohibition of communicating with the system memory at the time of the direct store. This is narrower than merely “not sending a transaction” during later core accesses (claim 12). It speaks to the store path itself.

The specification teaches that the invention bypasses system memory bandwidth during this direct store and writes back to memory only upon eviction. *Id.*, Fig. 3, steps 330 → 350/360; 3:45 – 4:11.

104. The petition treats claim 14 as riding on earlier mapping for claims 7, 11 and 12, citing Agarwal’s “cache coherence case” (Fig. 9C) to assert that I/O writes go “directly into the shared L3 cache.” Pet., 37. It does not identify any explicit teaching in Agarwal that this direct write occurs “without communicating with the system memory.” Dr. Hassoun treats claim 14 in a similar summary fashion, cross-referencing his analysis for claims 7/11/12, but does not identify where Agarwal affirmatively teaches the “without communicating” constraint, nor does he provide a reasoned motivation why a POSITA would implement a no-memory-communication store path. Ex. 1003, ¶¶ 286-288.

105. The failures of the petition and Dr. Hassoun’s declaration are not surprising because Agarwal does not disclose “without communicating with the system memory” at the time of the direct cache store. While Agarwal’s ProxyIO “cache coherence case” (Fig. 9C) does involve the I/O module “writ[ing] data directly into the L2/L3 cache,” Agarwal does not say this occurs without any communication with system memory, as claim 14 expressly requires. Citing a “write into L3” does not equal a disclosure that it is done without communicating with memory.

106. Further, claim 14’s timing matters. The “without communicating” constraint attaches to the store step, not only to later accesses. Petitioner points to claim 12-type behavior (multiple core accesses without sending a memory transaction) and then infers claim 14. Pet., 37. That conflates different timing constraints. While claim 12 is concerned with subsequent core reads/writes, claim 14 is concerned with the moment of direct store of the allocation transaction’s data into the shared cache. The petition does not identify any passage where Agarwal affirmatively characterizes the direct I/O-to-cache write as entirely non-communicative with system memory.

107. Finally, reliance on alternative I/O paths in Agarwal (memory-write case vs. cache-injection case) does not cure the petition’s deficiency with respect to claim 14. Agarwal presents two I/O paths: (1) writes to main memory and (2) writes into cache (Fig. 9B vs. 9C). Ex. 1005, 28:10 – 29:28. The existence of a cache-injection alternative does not establish that the cache-injection path itself is implemented “without communicating with the system memory.” The petition assumes that because data lands in cache, no memory communication occurs, but it does not cite any statement in Agarwal tying the cache-injection path to the no-communication constraint that claim 14 explicitly imposes.

B. Overview of Sinharoy

108. Sinharoy describes the IBM POWER7 processor, an eight-core chip with a three-level cache hierarchy and integrated I/O controllers. Each core has private L1 and L2 caches, and all cores share a logically unified 32MB L3 cache, physically implemented as eight 4MB local L3 regions, each with its own centralized controller. The chip also includes two integrated I/O controllers, supporting DMA cache injection, and a high-bandwidth on-chip coherence fabric that connects all cores, caches, memory controllers, and I/O controllers. Ex. 1006, 1-2, 16-21, 22-23.

109. B. Coherency Mechanism: Peer-to-Peer Broadcast Snoop: In POWER7, cache coherency is enforced by a broadcast-based protocol. When a cache miss or coherence event occurs, the request is broadcast over the coherence fabric and snooped by all relevant agents: L2 and L3 controllers for each core, the memory controller, and the I/O controllers. Each controller acts as an independent peer, responding to coherence events for its own region or domain. *Id.*, 19, 23. There is no single, unified caching agent that manages coherency for both the processor cores and the I/O module. Instead, each controller (L2, L3, I/O) maintains and enforces coherence for its own region, and the coherence fabric arbitrates and delivers messages among them. *Id.*

110. I/O Cache Injection: No Proxying Through a CPU Caching Agent: Sinharoy describes a feature called “cache injection,” whereby an I/O controller can write DMA data directly into a local L3 region if that region already owns the targeted cache line. If not, the data is written to memory. *Id.*, 17, 22. The I/O controller initiates and manages these transactions as a full peer on the coherence fabric; there is no teaching or suggestion that I/O requests are forwarded to or proxied through a CPU caching agent. The I/O controller is not a portion of a CPU caching agent, nor does any single agent perform coherency for both the cores and the I/O module.

111. Distributed Controllers, Not a “Single Caching Agent.” While Sinharoy’s L3 cache is physically distributed into eight regions, each with its own controller, *id.*, 19, these are not described as distributed portions of a single caching agent. Rather, each L3 region controller operates independently, and the coherence protocol is enforced collectively by all peer agents via the broadcast fabric. *Id.* The architecture does not present a unified control plane or a single agent responsible for all coherency actions for both CPU and I/O traffic. The I/O controller participates as a peer, not as a subordinate or proxy of a CPU caching agent. *Id.*, 22-23.

1. Sinharoy Does Not Disclose or Suggest a Single Caching Agent as Recited in Claim 1

112. Claim 1 requires a single caching agent for a multicore processor, implemented in a plurality of distributed portions each associated with a corresponding core. That single caching agent performs cache coherency operations for all the cores and an IIO module. Ex. 1001, 7:36-36.

113. As described above, a POSITA reading the '228 patent would understand the recited caching agent as a single coherency controller that (1) performs all cache-coherency operations for both the plurality of CPU cores and the integrated I/O (IIO) module, (2) is physically partitioned into distributed portions (one associated with each core/LLC slice) that together constitute that one agent, (3) presents one uniform control plane (shared arbitration/state-tracking resources and policies) for both CPU and I/O traffic, and (4) proxies IIO transactions through that same agent, thereby reducing the number of caching agents in the system and reducing snoop traffic. *See, e.g., id.*, 2:33-65 (ingress/arbiter/TOR); 2:58–65 (coherency logic “to perform cache coherency operations on behalf of the cores and IIO module”); 3:10-11 (“a system can include a single caching agent per multicore processor socket”); 3:4–12 (“the IIO module proxies through the CPU caching agent to access memory or other IO devices”); 3:45 – 4:4; 4:15–54 (IIO allocation transactions forwarded to the caching agent, which then directly stores into the LLC, and the agent sends “drop ownership”

indications when needed); 6:22-29 (“while distributed caching agents are shown, ... these distributed portions form a single caching agent” that handles coherency for the cores and the I/O); 6:25-29; 1:19–30; 3:4–12 (contrasting patented approach with “traditional” designs that treat integrated I/O as a separate caching agent, which doesn’t scale due to extra snoops).

114. Sinharoy’s POWER7 architecture is fundamentally different: Sinharoy employs multiple independent controllers (for L2, L3, and I/O), each acting as a peer on a broadcast coherence fabric, with no single agent managing coherency for both CPU and I/O domains. Ex. 1006, 1-2, 16-21, 22-23. Each L3 region has its own controller and resources, and each L3 region’s controller provides its own dispatch ports and pools of machines—confirming per-region controllers, not one agent spread across slices. *Id.*, 19; Ex. 1003, ¶ 326. Coherency is enforced by a broadcast coherence fabric that all agents (memory controller, L2/L3 caches, I/O controllers) snoop. Ex. 1006, 19, 23. The integrated I/O controllers participate in the same broadcast/snoop coherence as the core-side directory controllers and “support cache injection, which enables DMA write traffic to be written directly into a cache, instead of to memory.” *Id.*, 17. In other words, the I/O is a full peer, not a subordinate portion of a CPU caching agent. It participates in coherence on equal footing, and Sinharoy does not characterize the I/O controller as a portion of a CPU caching agent, nor describe the CPU agent

performing coherency for I/O. The coherence protocol is enforced collectively by all participants, with each agent responsible for its own coherence actions.

115. The contrast can be understood by analogy: The '228 patent's architecture is like a single control tower managing all flights in and out of an airport. Sinharoy's architecture, on the other hand, involves a number of independent towers, each responding to messages of the others.

116. This same analogy highlights why Dr. Hassoun's position, that the set of L2/L3 controllers acting cooperatively constitutes a "single caching agent," is incorrect. As I described above, message passing via a broadcast interconnect (the coherence fabric) does not make multiple, independent controllers one agent. The fabric broadcasts information; but each cache region's controller still performs coherency operations independently. The fabric does not maintain cache-line state for all cores/I/O.

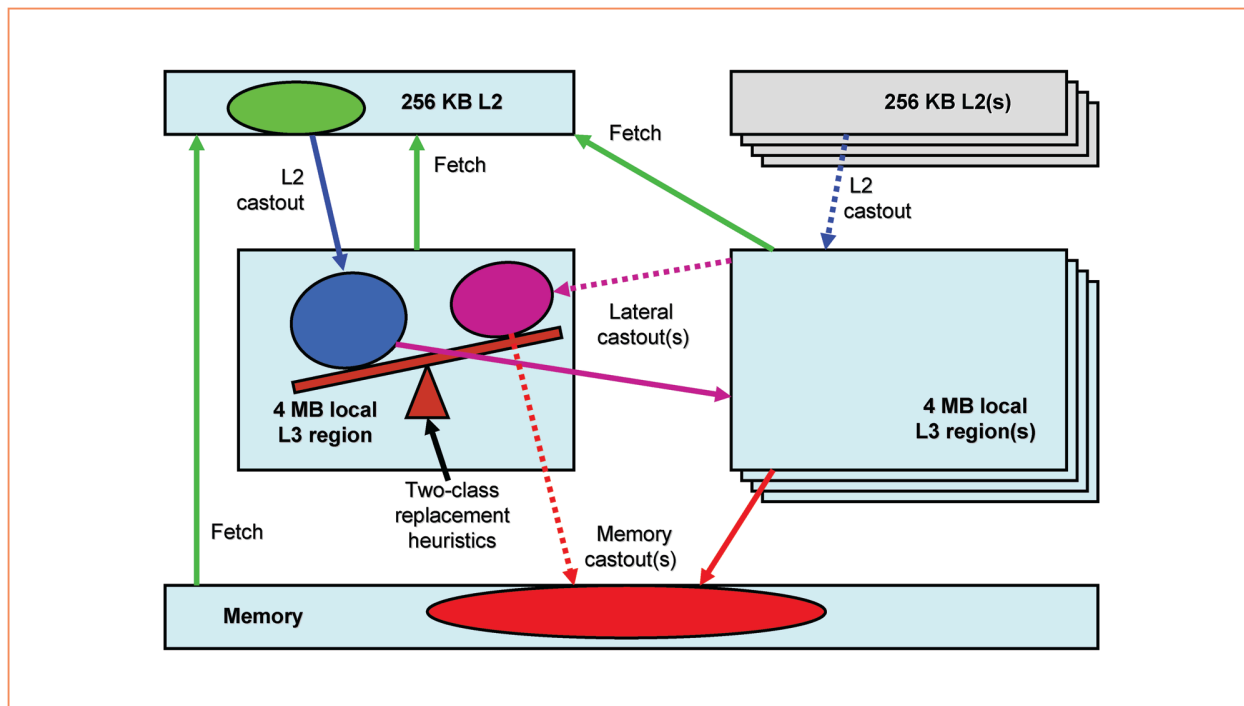
117. Thus, Sinharoy does not disclose or suggest a single caching agent—distributed or otherwise—that manages coherency for both CPU and I/O domains as required by claim 1. This distinction is dispositive and underscores the patentability of the challenged claims.

2. Sinharoy Does Not Disclose or Suggest Distributed LLC Portions Each Associated with a Corresponding Core, as Recited in Claim 4

118. Claim 4 requires a shared cache memory (LLC) that includes distributed portions, each associated with a corresponding core. Ex. 1001, 7:53-56.

119. Sinharoy fails to disclose this association:

120. First, POWER7's L3 regions are not "associated with a corresponding core." Sinharoy notes that the L3 cache is divided into eight regions, but these are not tied one-to-one to specific cores. Instead, each region has its own centralized controller, with a core/L2 dispatch port and a lateral L3 region dispatch port (plus dual snoop dispatch ports). Ex. 1006, 19. This is a per-region organization—not a per-core association. The presence of a lateral ingress path means each L3 region is architected to accept non-local requests (i.e., from other cores), *Id.* ("[O]perations that miss the 4-MB local L3 region but hit in the remaining 28 MB of the L3 cache access the cache via the snoop dispatch ports. L2 cast-out operations access the 4-MB local L3 region via the core/L2 dispatch port, whereas lateral cast-outs access a given 4-MB L3 region via the lateral L3 region dispatch port."). This is consistent with a shared (not per-core bound) LLC, and inconsistent with the claimed per-core association; and see Fig. 10 (showing a fetch from a non-local L3 region).



121. Second, there is no motivation to repartition POWER7’s L3 into per-core portions. Petitioner provides no reasoned explanation or expectation of success for modifying POWER7’s address-homed, shared L3 into a per-core association as required by the claim. Petitioner may argue that because a 4-MB “local” L3 region is “tightly coupled” to a core’s private L2 this implies at least a motivation for per-core association. But proximity/latency optimization is not the same as the association recited in Claim 4. The lateral port in each POWER 7 L3 cache region exists precisely to handle non-local accesses, showing that any core can use any L3 region when the address maps there. Adopting the per-core

association of Claim 4 would require eliminating this access, directly contradicting the principle of operation articulated in Sinharoy. Such a modification is therefore non-obvious.

122. In my experience, in designs like POWER7, address bits, not core IDs, determine the “home” L3 region. Sinharoy’s architectural elements—the lateral region port, broadcast snoop buses, and independent region controllers—all align with an address-homed shared cache. For example, the local L3 regions primarily act as a victim cache for their associated L2 cache. Data evicted from a core’s private L2 cache is sent to its local L3 region. If that region is full or if the data needs to be accessed by another core, the cache line can be routed to other L3 regions or main memory based on heuristics. *Id.*, 19-20. Hence, any core accessing the same physical address will be sent to the same L3 cache location. This system allows for efficient data sharing and maintains cache coherence across cores within the chip (and even across multiple chips in a system through high-speed SMP links). *See id.*, 24. This arrangement is inconsistent with claim 4’s per-core association of LLC portions. Petitioner points to physical locality but never shows core-based association of L3 regions as the claim demands.

3. Sinharoy Does Not Disclose or Suggest an Allocation Transaction Received from an IIO Module, as Recited in Claim 6

123. Claim 6 requires the caching agent receive an allocation transaction from the IIO module. The allocation transaction is received in the IIO module from an I/O device. Ex. 1001, 7:60-64.

124. Sinharoy does not disclose this message flow:

125. In Sinharoy there is no I/O device-originated “allocation transaction” forwarded to a caching agent. The ‘228 patent defines an allocation transaction as a request to directly store data into the LLC (i.e., to allocate a line in the LLC for I/O data), sent from the I/O device to the IIO, and forwarded by the IIO to the caching agent. *See id.*, 3:45 – 4:4. Sinharoy instead describes “cache injection,” that is, directing DMA traffic to the correct physical L3 slice based on the address-homing function. Ex. 1006, 17. This is a controller-side optimization, not a distinct transaction type issued by the I/O device and delivered to a coherency agent. Further, in Sinharoy’s cache injection there is no two-hop message flow, I/O Device → IIO → Caching Agent, required by Claim 6. In Sinharoy, the I/O controllers implement cache injection by writing into an L3 region when that region already owns the line. There is no intermediate recipient that functions as a “single” caching agent to which the IIO forwards a distinct allocation request. *Id.*, 17, 22–23. In other words, the IIO acts as a coherent requester on the fabric, possessing the intelligence to target the correct L3 portion directly.

4. Sinharoy Does Not Disclose or Suggest I/O Device-Originated Allocation, Forwarding, and Direct Storing, as Recited in Claim 11

126. Claim 11 requires (1) receiving an allocation transaction in an IIO module from an I/O device, (2) forwarding the allocation transaction to a caching agent shared by the cores and IIO module, and (3) directly storing the data into a shared cache memory. Ex. 1001, 8, 11-21.

127. Sinharoy does not disclose these steps, arranged as claimed. As discussed above with respect to Claim 6, Sinharoy does not describe a two-hop, device-originated allocation transaction that is forwarded to a caching agent. Claim 11 defines the incoming message: an allocation transaction comprising a request to directly store into LLC. The same allocation transaction that arrived in the IIO is forwarded to the single caching agent. In Sinharoy's cache injection, DMA writes initiated by the I/O device are direct into an L3 region (when that region already owns the line). Ex. 1005, 17, 22–23. There is no intermediate recipient that functions as a “single” caching agent to which the IIO forwards a distinct allocation request. Instead, the I/O controller implements cache injection locally, based on cache line ownership.

C. Claims 2, 3, and 16 Are Patentable Over Sinharoy and Blumrich

1. Summary of Blumrich

128. Blumrich describes a multiprocessor system with a cache-based point of coherence, focusing on speculative execution, transactional memory (TM), and thread-level speculation (TLS). Ex. 1007, Abst., [0018]-[0027], [0080]-[0081], [0092]-[0098]. The system includes:

- a. Centralized Speculation Management: A central unit (L2 CENTRAL) manages speculation IDs, tracks thread states, allocates and commits IDs, and coordinates conflict detection and resolution across multiple processor cores and cache slices. *See id.*, [0077]-[0081].
- b. Speculation IDs and Domains: Each speculative thread is assigned a unique speculation ID, which tags all memory requests from that thread. IDs are managed in domains, allowing independent groups of threads to execute in different speculative modes (TLS, TM, rollback). *Id.*
- c. Cache as Point of Coherence: The L2 cache is the point of coherence for all processors, maintaining memory consistency and handling speculative reads and writes. Speculative writes are held in the cache and not written to main memory until the thread is committed. *Id.*, [0119] *et seq.*

- d. Conflict Detection and Resolution: The system maintains detailed records of speculative reads and writes in the cache directory. Conflict checking is performed on directory lookup, and conflicts are recorded and signaled to the responsible processor. *See, e.g.*, [0242]-[0243].
The system supports multiple versions of a cache line; each associated with a different speculative writer. *Id.*, [0152].
- e. Version Aggregation: When speculative threads are committed, the cache aggregates multiple versions of a cache line into a single committed version, as directed by the central command execution unit. *Id.*, Abst., [0232]; *and see* claim 16.
- f. Directory-Based Coherence: The system uses directory-based coherence, with the cache directory tracking which cores have copies of each line and which lines have been modified speculatively. *Id.*, [0121], [0125].
- g. Transactional Memory and Rollback: The system supports transactional memory, allowing atomic execution of critical sections, and rollback mode for error recovery. The central unit manages mode switching and ensures proper commit and invalidation of speculative states. *Id.*, [0183] *et seq.*

2. Claim 2 is Patentable Over Sinharoy and Blumrich

129. Claim 2 reads, “The apparatus of claim 1, wherein the caching agent includes a table of requests to store pending requests for the plurality of cores and the IIO module.” Ex. 1001, 7:47-49.

130. In the proposed Sinharoy-Blumrich combination, Petitioner relies on Sinharoy for POWER7’s L2/L3 controllers, which provide pools of “read/claim (RC) machines,” “write machines,” and related resources; and also mentions queues such as a “load reorder queue,” “store reorder queue,” etc., Pet., 66, but those are core-internal (L1/L2 path), and the RC/write machines are controller pipelines, not a unified table within the caching agent.

131. The ’228 patent identifies the table of requests (“TOR”) as “a buffer or other temporary storage for holding incoming requests,” with dedicated posted-transaction capacity. Ex. 1001, 2:44-52. In the embodiment of Fig. 2, ingress queues feed an arbiter, which then inserts incoming (core/IIO) transactions into the TOR before the LLC/controller logic acts on them. *Id.*, 2:53-65. This is an ingress storage element, not an execution engine. The practical upshot of this is that Claim 2 requires a data structure that stores pending requests, i.e., a queue/buffer at the front door of the caching agent shared by the requestors (cores and IIO).

132. For its part, Sinharoy describes each L3 controller as having a “pool of operational resources”—eight read machines, 24 lightweight RC (read/claim) machines, and ten write machines, which perform the controller’s operations (managing read hits, store misses, prefetches, writes, etc.). Ex. 1006, 19. An “RC machine” (or “write machine”) in Sinharoy is a state machine within a cache controller to manage sequences of cache coherence operations. That is, it is what a controller uses to execute a transaction after it has been scheduled. RC machines handle read requests and the process of claiming ownership of a cache line when a read request is initiated; write machines handle write requests and manage the process of modifying a cache line. Instead of one monolithic controller handling all possible 13 coherence states (see Table 2, p. 17) simultaneously, the POWER7 design uses these specialized, independent “machines” to manage different transaction types (read/claim vs. write). Importantly, these state machines are not described as storing a population of pending transactions, and so is not the TOR recited in claim 2.

133. In Blumrich, the L2 is the point of coherence and contains a “request queue 302” that “buffers incoming read and write requests,” with a directory pipe and conflict-checking for speculative/TM operation. Ex. 1007, [0128]. This is a per-slice L2 queue, not an I/O queue nor an L3 caching-agent structure.

Nevertheless, Petitioner argues that a POSITA would have been motivated to

modify Sinharoy's L3 controller to include a request queue like Blumrich's, and that this queue would serve as the claimed TOR. Pet., 67-68.

134. I disagree. Blumrich's request queue is not a table for mixed core/IIO requests in a caching agent. Blumrich's request queue 302 is part of the L2 slice controller in a system where the L2 is the point of coherence. It buffers incoming read and write requests from the processor cores and is used for speculative execution and transactional memory management. Ex. 1007, [0128]-[0135]. Blumrich does not disclose or suggest that this queue is used to buffer or store requests from an IIO module (i.e., from DMA or external I/O sources). Its function is to manage speculative and non-speculative memory requests from the processor threads, not I/O traffic.

135. Even if one considers Sinharoy's L3 controller as handling both core and IIO traffic, via cache injection, it does not disclose a unified ingress queue or table that stores pending requests from both sources before execution. The assertion that a POSITA would "obviously" implement Sinharoy's L3 controller with a request queue like Blumrich's is one that neither reference suggests nor motivates. Sinharoy does not suggest the need for a unified ingress queue for both core and IIO requests at the L3 controller level. Its "pool of operational resources" (RC/write machines) are execution resources, not ingress buffers, and are not described as holding pending requests from both sources. Blumrich does not teach

or suggest that its request queue is suitable for, or should be adapted to, handle both core and IIO requests in a single table at the L3 controller level. Its queue is L2-centric and focused on speculative execution management.

136. Further even if combined, a Sinharoy-Blumrich system would not have a TOR for both cores and an IIO module. Blumrich's queue is not shown to handle IIO requests and adapting it to do so would require significant architectural changes not suggested by either reference. Moreover, modifying Sinharoy's L3 controller to track all pending requests, including those from the IIO module is not a simple substitution. Sinharoy's architecture does not require or suggest a single ingress queue for both types of requests; its execution resources are provisioned to handle requests as they arrive, and there is no disclosure of a unified pending request table for both sources.

137. The petition's alleged motivation to combine is based on general principles (alleging it would be obvious to use a queue or buffer to track outstanding requests), Pet., 67, but does not address the specific technical challenges or architectural differences between L2 speculative request management (Blumrich) and L3 cache injection/multisource traffic (Sinharoy). There is no evidence that a POSITA would have reasonably expected that simply adopting a Blumrich-like queue as part of Sinharoy's L3 controller would result in a system

that meets the claim—i.e., a single caching agent with a table of requests for both cores and IIO module.

138. Thus, in my opinion, Claim 2 is not taught or suggested by Sinharoy and Blumrich. Even when considered together as the petition proposes, Sinharoy and Blumrich do not suggest a caching agent that includes a table of requests to store pending requests for the plurality of cores and the IIO module. Certainly, the combination is not expressly or implicitly disclosed, and the alleged rationale for the combination is generic and not grounded in the specific architectures or teachings of the references.

3. Claim 3 is Patentable Over Sinharoy and Blumrich

139. Claim 3 reads: “The apparatus of claim 2, wherein the table of requests includes at least one dedicated entry to store a posted transaction.” Ex. 1001, 7:50-52.

140. This is an enhancement to the TOR recited in claim 2. The specification explains that the TOR includes “at least one dedicated resource 225 for posted transactions,” i.e., an entry allocated for one-way, non-completion I/O writes (“posted”)—a behavior with material architectural implications. *Id.*, 2:49-50. The specification emphasizes that this dedicated entry is to ensure that posted

transactions are not blocked by non-posted requests, which is important for deadlock avoidance and system performance. *Id.*, 5:41-45.

141. As noted above, Sinharoy and Blumrich do not teach or render obvious a caching agent that includes a table of requests to store pending requests for the plurality of cores and the IIO module. Furthermore, the combination of these references does not teach or suggest the specific TOR with a dedicated entry specifically for posted transactions (i.e., write operations that do not require a completion/acknowledgment), as recited in Claim 3.

142. Blumrich's request queue 302 is a general-purpose buffer for incoming read and write requests from processor cores, used for speculative execution and transactional memory management. Ex. 1007, [128]. Blumrich does not disclose or suggest that any entry in the queue is "dedicated" for posted transactions, nor does it discuss posted I/O semantics at all. All entries are available for any type of request, and there is no mention of reserving or dedicating an entry for a particular class of transaction (such as posted I/O writes).

143. Sinharoy's "cache injection," where a DMA write from an IIO module may target the cache instead of memory, does not rely on or even make use of a table of requests, let alone one with a dedicated entry for posted transactions. Ex. 1006, 17. Sinharoy's L3 controller uses pools of execution resources (RC/write machines), but these are not described as a queue or table, and certainly not one

with reserved entries for posted transactions. Indeed, there is no disclosure of a mechanism to ensure that posted transactions always have a dedicated slot in an RC/write machine, even if it were considered a request table.

144. In asserting obviousness, the petition conveniently overlooks these shortcomings of the references. Further, the petition offers no specific motivation to combine, and instead merely states the conclusion of obviousness as the premise for same. Pet., 68-69 (arguing it would be obvious to include a dedicated write queue because there are write transactions). The petition does not address the specific technical challenges or architectural differences between L2 speculative request management (Blumrich) and L3 cache injection/multisource traffic (Sinharoy) that a POSITA would face in even attempting the proposed combination.

145. First, the POSITA would need to make architectural changes in Sinharoy to provide a unified ingress buffer (the TOR) within the L3 controller that receives and temporarily stores all incoming requests from both the processor cores and the IIO module. This buffer would have to be accessible to both types of requestors and must precede execution by the controller's operational resources (RC/write machines).

146. Second, the POSITA would have to implement a data structure (e.g., a queue or array) that can hold multiple pending requests, with metadata indicating

the source (core or IIO), request type, and status. It would further need at least one entry specifically reserved for posted transactions (i.e., transactions that are issued and do not require immediate completion or acknowledgment, such as posted writes from IIO/DMA). Neither reference describes how that would be implemented and the petition points to no general knowledge in the art that would fill this gap.

147. Third, the system would have to be able to identify posted transactions (from IIO or cores) at ingress and route them to the dedicated entry in the queue. The petition offers no direction on how a POSITA could or would modify ingress logic to recognize posted transactions and ensure they are stored in the dedicated entry, rather than being handled like ordinary requests.

148. Fourth, any controller and arbitration logic would require modification. For example, the arbiter, which schedules requests for execution, would have to be updated to read from the TOR, including the dedicated posted transaction entry, and to prioritize or handle posted transactions according to system policy. The POWER 7 controller's operational resources (RC/write machines) would have to be coordinated with the TOR so that requests are only executed after being entered into the table, maintaining the pending status until execution. Again, the petition offers no comments as to how this would be accomplished.

149. Finally, a series of additional modifications to integrate with an I/O module and provide coordinated recordkeeping and dedicated entry guarantees would be required. The I/O module would need to be connected to the controller's ingress logic so that its posted transactions are routed into the TOR, not bypassed or handled separately. And both core and IIO requests would have to be handled in a unified manner, with the TOR serving as the single point of entry for pending requests. These involve more than mere firmware changes and would directly impact the design of the hardware of the POWER7. As to recordkeeping, the TOR would have to track the status of each request (pending, executing, completed), especially for posted transactions which may have different completion semantics, and the system would have to ensure that the presence of posted transactions in the TOR is visible to other system components (e.g., for flow control, error handling, or reporting) while still guarantying that at least one entry in the TOR was always available/reserved for posted transactions, as required by the claim.

150. None of this is straightforward and the petition has not shown it is within the general knowledge of a POSITA. Sinharoy's architecture does not have a unified ingress buffer or TOR; its execution resources are not designed to hold pending requests from both cores and IIO. Blumrich's request queue is per-slice and focused on speculative execution, not on posted transactions or unified core/IIO traffic. Combining the two would require significant redesign of the

controller's ingress, arbitration, and request tracking logic, as well as new data structures and coordination mechanisms. The combined teachings of the references fails to include teachings or even suggestions as to why or how to make these changes, and in light of such failings a general, unspecified motivation to "avoid deadlocks or improve performance" provides an insufficient basis to conclude such modifications would be obvious.

151. Thus, even when considered together, as the petition proposes, Sinharoy and Blumrich do not teach or render obvious a table of requests that includes at least one dedicated entry to store a posted transaction, as recited in Claim 3.

4. Claim 16 is Patentable Over Sinharoy and Blumrich

152. Claim 16 recites a system with two multicore processors, one of which includes a caching agent including a TOR with multiple entries for core or IIO pending requests, including at least one dedicated entry for a posted transaction, plus a peripheral controller. Ex. 1001, 8:42-64. Accordingly, all of the reasons for patentability of Claims 2 and 3 discussed above apply equally to Claim 16.

D. Claim 5 is Patentable Over Sinharoy and Sistla and Claim 19 is Patentable Over Sinharoy, Blumrich, and Sistla

153. Petitioner challenges Claim 5 of the '228 patent as being obvious in view of a combination of Sinharoy and Sistla, US PG PUB 2008/0126750 A1 (Ex. 1008) and Claim 19 as being obvious in view of Sinharoy, Blumrich, and Sistla. Pet., 74 *et seq.* These challenges rely on Sistla's teachings regarding the use of a ring interconnect. *Id.*

154. Sistla describes a system and method for aggregating core-cache clusters in a scalable manner to produce multi-core processors. Ex. 1008, Abst. The key architectural feature is the use of a scalability agent, which operates as an interface between an on-die interconnect and both multiple processor cores and memory. The scalability agent protocol enables multiple core-cache clusters to appear as one caching agent to external devices, handles local cache coherence, and partitions work among clusters. *Id.*, [0018]-[0019].

155. Claim 5 reads, "The apparatus of claim 4, wherein the plurality of distributed portions of the caching agent are coupled via a ring interconnect." Ex. 1001, 7:57-59.

156. Claim 19 reads, "The system of claim 16, wherein the first caching agent is a single caching agent for the first multicore processor and includes a plurality of distributed portions each associated with a corresponding one of the first plurality of cores, the first shared cache memory comprises a last level cache

(LLC) including a plurality of distributed portions each associated with a corresponding one of the first plurality of cores, and wherein the plurality of distributed portions of the first caching agent are coupled via a ring interconnect.” *Id.*, 9:9-17.

157. In its challenges to Claims 5 and 19, Petitioner asserts that Sistla’s disclosure concerning a “ring interconnect” would be an obvious feature to include in the architecture of Sinharoy and Sinharoy-Blumrich, respectively. Pet., 75-78. This is not so.

158. Sistla’s “ring” couples clusters, not distributed portions of a caching agent. Ex. 1008, [0018]-[0019], [0030]. Sistla shows an on-die interconnect 320 that “is configured as a ring interconnect (but may be a 2D mesh)” to connect core-cache clusters (310...310_n). The scalability agent (SA) is associated with each cluster and address-partitioned into separate “SA0...SA3” units. *Id.*, [0030], [0034].

159. Sistla’s SA (i) issues on-die snoops, (ii) aggregates snoop responses, and (iii) performs address conflict checks so a socket “appears as one caching agent” to an external interconnect. *Id.*, [0040]. Thus, it is a coordination/snoop-aggregation layer, not an entity “that performs cache coherency operations for the plurality of cores and the IIO module.” In other words, Sistla’s SA is not a “caching agent” as recited in the challenged claims.

160. This lack of identity is confirmed by considering that for write-class transactions the “SA only sequences the address portion” and “does not contain any data buffers.” *Id.*, [0065]. This underscores that Sistla’s SA is not a cache-controlling “caching agent” as claimed.

161. Petitioner’s theory of obviousness depends on conflating Sistla’s SA with the caching agent recited in Claims 1 and 19. The reference itself does not support such an interpretation; hence, Petitioner’s theory collapses.

162. Claim 5 and 19 are not obvious in view of Sinharoy and Sistla, or Sinharoy, Blumrich and Sistla, respectively, for at least these reasons.

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

Executed this 16th Day of December, 2025 in Irvine, CA.

By:  _____

Nader Bagherzadeh, Ph.D.

APPENDIX A

Nader Bagherzadeh, Ph.D.

Expertise

- Digital Signal Processing Architectures
 - Processor Architectures (VLIW, Superscalar)
 - VLSI Design
 - Computer Networks & Digital Telecommunication
 - Computer Graphics
 - Memory System Architecture
 - DRAM, SRAM, and Flash memory
 - Automotive Computing
 - Network-on-Chip (NoC)
 - Error Detection and Correction
 - Machine Learning Accelerators
-

Contact Information:

Cell: (949) 400 5167

Office: (949) 824 8720

Email: nader@uci.edu

Office Address:

EH 4209

Electrical Engineering and Computer Science Department

University of California, Irvine

Irvine, CA 92697

Education

<u>Year</u>	<u>College/University</u>	<u>Degree</u>
1987	The University of Texas at Austin	Ph.D., Computer Engineering
1979	The University of Texas at Austin	MS, Electrical Engineering
1977	The University of Texas at Austin	BS (with honors), Electrical Engineering

Employment History

From:	1987	University of California at Irvine
To:	Present	Irvine, CA
Position:		<i>2003- Present: Professor, Department of Electrical Engineering and Computer Science</i>
		<i>1998-2002: Professor and Chair, ECE/EECS Department</i>
		<i>1993-1997: Associate Professor, ECE Department</i>
		<i>1987-1992: Assistant Professor, ECE Department</i>

Consultant Curriculum Vitae

From: 1980 **AT&T Bell Labs**
To: 1984 Holmdel, NJ
Position: *Member of Technical Staff, No.5 ESS Hardware/Software T1Line Interface Design*

Awards and Honors

1973-77 Dean's honor roll for six semesters; UT Austin
1984 Outstanding achievement award; AT&T Bell Labs
2002 Best Paper award in IEEE Transactions on VLSI Design
2002 Best Student Paper award in the proceedings of ASPDAC'02
2014 Khwarizmi International Award
2014 IEEE Fellow
2015 Professor of the Year for the EECS Department, UCI
2020 Best Paper award in the proceedings of IEEE EDGE Computing

Consulting History

From: 2006 **STEC**
To: 2008 Irvine, CA
Duties: Flash memory CODEC design and evaluation

From: 2006 **Tialinx**
To: 2008 Newport Beach, CA
Duties: Radar DSP board design and evaluation

From: 2000 **Morpho Technologies**
To: 2005 Irvine, CA
Duties: Co-Founder; DSP design for communication and multimedia systems

Litigation Support Experience

Date: 2024 **Nixon Peabody LLP**
Case: *Network System Technologies LLC ("NST") vs. Qualcomm*
Project: Expert Witness for NST; provided declarations and depositions
Status: IPR
Contact: Jennifer Hayes
300 S. Grand Avenue, Suite 4100
Los Angeles, CA 90071-3151
Direct Dial: (213) 629 6179
jenhayes@nixonpeabody.com

Consultant Curriculum Vitae

- Date: 2024 **Kirkland & Ellis LLP**
Case *TCL Electronics Holdings Ltd. vs. Intellectual Ventures I LLC*
Project: Expert Witness for TCL; provided declarations;
Status: IPR
Contact: Russell Levine
300 North LaSalle
Chicago, IL 60654
Direct Dial: (312) 862 2466
Russell.levine@kirkland.com
- Date: 2024 **Covington & Burling LLP**
Case *Redstone Logics LLC v. Samsung Electronics Co., Ltd.*
Project: Expert Witness for Samsung Electronics Co., Ltd. Patent analysis (No Reports/Declarations)
Status: Settled
Contact: Ali Mojibi
1999 Avenue of the Stars
Los Angeles, CA 90067-4643
(424) 332 4803
amojibi@cov.com
- Date: 2023 **McKool Smith**
Case *TexasLDPC Inc. v Broadcom Inc., LSI Corporation, Avago Technologies U.S., Inc.; C.A. No. 18-1966-SB*
Project: Expert Witness for Broadcom; provided declaration and was deposed
Status: Dismissed without prejudice
Contact: Steven Rizzi
One Manhattan West
395 9th Avenue, 50th Floor
New York, NY 10001
(212) 402-9809
srizzi@McKoolSmith.com

Consultant Curriculum Vitae

Date: 2023 **Mintz**
Case: Daedalus v. Qualcomm Inc. and Samsung Electronics Co.
Project: Expert Witness for Daedalus Prime LLC
Status: ITC, provided technical support without rendering an opinion in a report.
Case is closed.
Contact: Peter F. Snell
pfsnell@mintz.com
212 692 6850
Chrysler Center
666 Third Avenue
New York, NY 10017
212 935 3000
mintz.com

Date: 2023 **Caldwell Cassidy & Curry**
Case: BeSang Inc. v. Intel and/or Micron, CIVIL ACTION NO. 2:23-CV-00028-JRG-RSP and Case No. 3:23-cv-00113-HZ
Project: Expert Witness for BeSang; patent analysis only
Status: Stay
Contact: Bjorn A. Blomquist
2121 North Pearl Street, Suite 1200
Dallas, Texas 75201
Office Line: (214) 888-4960
Mobile: (608) 790-0902
bblomquist@caldwellcc.com

Date: 2022 **Arnold & Porter**
Case: Rex Computing, Inc. v. Cerebras Systems Inc.
Project: Expert Witness for Rex Computing, IPR Petition by Cerebras
Status: Board sided with Rex and did not grant the petition; provided declaration and was deposed
Contact: Nicolas Lee,
44th Floor, 777 South Figueroa Street
Los Angeles, CA 90017
Direct Tel.: (213) 243-4156
Company: (213) 243-4000
nicholas.lee@arnoldporter.com

Consultant Curriculum Vitae

- Date: 2022 **Quinn Emanuel Urquhart & Sullivan, LLP,**
Case MediaTek Inc. v. NXP Semiconductors N.V.
Project: Expert Witness for Media Tek; provided patent analysis only
Status: Settled
Contact: Jordan B. Kaericher
 865 S Figueroa St 10th Floor
 Los Angeles, CA 90017
 Direct Tel.: (213) 443-3843
 Company: (213) 443-3000
 jordankaericher@quinnemanuel.com
- Date: 2022 **Bartlit Beck LLP**
Case Viasat vs. Kioxia and Western Digital, Patent Litigation Consulting
Project: Expert Witness for Viasat. Provided declaration and was deposed
Status: IPR; Board decided in favor of Viasat
Contact: Scott McBride
 54 West Hubbard Street
 Chicago, IL 60654
 Direct Tel.: (312) 494-4436
 scott.mcbride@bartlitbeck.com
- Date: 2021 **Kilpatrick Townsend, LLP**
Case Autel Intelligent Tech. Corp., LTD vs. Orange Electronic Co., LTD
Project: Expert Witness for Autel. Provided IPR declaration
Status: Closed
Contact: David A. Reed,
 Suite 2800, 1100 Peachtree Street NE
 Atlanta, GA 30309-4528
 Direct Tel.: (404) 745-2548
 Company: (404) 815-6500
 dreed@kilpatricktownsend.com

Consultant Curriculum Vitae

- Date: 2021 **Desmarais LLP**
Case: AMD, QCM, and STM vs. Monterey Research
Project: Expert Witness for Monterey. Provided IPR declarations and was deposed.
Status: Settled
Contact: Jordan Malz,
230 Park Avenue
New York, NY 10169
Direct Tel.: (212) 351-5497
Company: (212) 351-3400
JMalz@desmaraisllp.com
www.desmaraisllp.com
- Date: 2020 **Devlin Law Firm LLC**
Case: Engagement for Assistance with Certain Patents Asserted by Flash-Control, LLC
Project: Expert Witness for Flash Control. Provided IPR declaration and was deposed.
Status: Pending IPR Decision
Contact: Jim Lennon,
1526 Gilpin Avenue
Wilmington, Delaware 19806
Tel.: (302) 449-9010
Cell: (302) 593-4486
www.devlinlafirm.com
- Date: 2019 **The Honorable David O. Carter**
Case: Limestone Memory Systems vs. Micron Technology, et al.
Project: Special Master for Judge Carter; provided a report for Judge Carter
Status: Settled
Contact: Judge David O. Carter,
Ronald Reagan Federal Building, United States Courthouse
411 West Fourth Street, Courtroom 9D
Santa Ana, CA 92701-4516

Consultant Curriculum Vitae

Publications

Journals Refereed

- j1 P. Suhler, N. Bagherzadeh, M. Malek, and N. Iscoe, "Software authorization systems," *IEEE Software*, pp. 34-41, September 1986.
- j2 A. Guzman, E. Krall, P. McGehearty, and N. Bagherzadeh, "Performance of symbolic applications on a parallel architecture," *International Journal of Parallel Programming*, vol. 16, no.3, pp. 183-214, 1987.
- j3 Y. Moon, N. Bagherzadeh, and J. Sklansky, "A macropipelined multicomputer architecture for image analysis," *Journal of the Optical Society of America*, pp. 951-962, 1989.
- j4 D. Blough and N. Bagherzadeh, "Near-optimal message routing and broadcasting in faulty hypercubes," *International Journal of Parallel Programming*, vol. 19, no. 5, pp. 405-423, 1990.
- j5 N. Bagherzadeh, S. Heng, and C. Wu, "A parallel asynchronous garbage collection algorithm for distributed systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 3, no. 1, pp. 100-107, March 1991.
- j6 A. Kavianpour and N. Bagherzadeh, "Application of three-valued logic for distributed termination detection," *International Journal of Computers and Electrical Engineering*, vol. 17, no. 2, pp. 65-74, 1991.
- j7 A. Kavianpour and N. Bagherzadeh, "Finding circular shapes in an image on a pyramid architecture," *Pattern Recognition Letters*, vol. 13, no. 12, pp. 843-848, December 1992.
- j8 N. Bagherzadeh and K. Hawk, "Parallel implementation of the auction algorithm on the Intel hypercube," *Journal of Computer and Software Engineering*, vol. 1, no. 1, pp. 17-32, 1993.

Consultant Curriculum Vitae

- j9 A. Abnous, C. Christensen, J. Gray, J. Lenell, A. Naylor, and N. Bagherzadeh, "Design and implementation of the 'Tiny RISC' microprocessor," *Journal of Microprocessors and Microsystems*, vol. 16, no. 4, pp.187-193, 1992.
- j10 A. Abnous and N. Bagherzadeh, "Pipelining and bypassing in a VLIW processor," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 6, pp. 658-664, June 1994.
- j11 A. Kavianpour and N. Bagherzadeh, "A systematic approach for mapping application tasks in hypercubes," *IEEE Transactions on Computers*, vol. 42, no. 6, pp. 742-746, June 1993.
- j12 A. Kavianpour and N. Bagherzadeh, "Parallel algorithms for line detection on a pyramid architecture," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, no.1, pp. 337-349, 1994.
- j13 C. Bowen and N. Bagherzadeh, "Image processing applications in C++ on a hypercube multicomputer," *International Journal of Computers and Electrical Engineering*, vol. 19, no. 5, pp. 351-364,1993.
- j14 A. Kavianpour, S. Shoari, and N. Bagherzadeh, "A new approach for circle detection on multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 20, no. 2, pp. 256-260, February 1994.
- j15 S. Latifi and N. Bagherzadeh, "Incomplete star: An incrementally scalable network based on the star graph," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 1, pp. 97-102, January 1994.
- j16 N. Bagherzadeh, N. Nassif, and S. Latifi, "A routing and broadcasting scheme on faulty star graphs," *IEEE Transactions on Computers*, Vol 42, no. 11, pp 1398-1403, November 1993.
- j17 J. Gray, A. Naylor, A. Abnous, and N. Bagherzadeh, "VIPER: a VLIW integer microprocessor," *IEEE Journal of Solid States and Circuits*, Vol. 28, no. 12, pp 1377-1383, December 1993.
- j18 J. Lenell and N. Bagherzadeh, "A performance comparison of several superscalar processor models with a VLIW processor," *Journal of Microprocessors and Microsystems*, Vol 18, no. 3, pp 232-242, 1994.
- j19 A. Kavianpour, N. Bagherzadeh, and S. Shoari, "Finding elliptical shapes in an image using a pyramid architecture, " *International Journal of Computers and Electrical Engineering*, vol. 21, no. 1, pp. 69-75, Jan 1995.

Consultant Curriculum Vitae

- j20 S. Latifi, M. M. de Azevedo, and N. Bagherzadeh, "A star-based I/O-bounded network for massively parallel systems," *IEE Proceedings-Computers and Digital Techniques*, Vol. 142, No. 1, January 1995.
- j21 M. M. de Azevedo, N. Bagherzadeh, and S. Latifi, "Broadcasting algorithms for the star-connected cycles interconnection network," *Journal of Parallel and Distributed Computing*, 25, 209-222 (1995).
- j22 S. Wallace and N. Bagherzadeh, "Performance issues of a superscalar microprocessor," *Microprocessors and Microsystems*, Vol. 19, No. 4, pp 187-199 (1995).
- j23 A. Abnous and N. Bagherzadeh, "Architectural design and analysis of a VLIW processor," *International Journal of Computers and Electrical Engineering*, Vol 21, No. 2, pp 119-142 (1995).
- j24 T. Kato and N. Bagherzadeh, "A novel superscalar architecture using scheduling patterns and performance analysis," *The Transactions of the Institute of Electronics, Information, and Communication Engineers D-I*, Vol.J78-D-I, No. 9, pp 809-819 (1995).
- j25 N. Bagherzadeh and M. Dowd, "Computation in faulty stars," *IEEE Transactions on Reliability*, Vol 44, No. 1, pp 114-119 (1995).
- j26 N. Bagherzadeh, M. Dowd, and S. Latifi, "A well-behaved enumeration of star graphs," *IEEE Transactions on Parallel and Distributed Systems*, Vol 6, No. 5, pp 531-535, May (1995).
- j27 M. M. de Azevedo, N. Bagherzadeh, M. Dowd, and S. Latifi, "Some topological properties of star connected cycles," *Information Processing Letters*, 58, pp 81-85 (1996).
- j28 N. Bagherzadeh, M. Dowd, and N. Nassif, "Embedding an arbitrary binary tree into the star graph," *IEEE Transactions on Computers*, Vol 45, No. 4, pp 475-481 (1996).
- j29 N. Bagherzadeh and M. Dowd, "Problems on routing bounded distance assignments in hypercubes," *Computer Systems-Science and Engineering*, Vol 11, No. 4, pp 221-226, July (1996).
- j30 S. Latifi and N. Bagherzadeh, "On cluster-star graph and its properties," *Computer Systems-Science and Engineering*, Vol 11, No. 3, pp 145-149, May (1996).
- j31 S. Shoari and N. Bagherzadeh, "Computation of the fast fourier transform on the star-connected cycle networks," *International Journal of Computers and Electrical Engineering*, Vol 22, No. 4, pp 235-246 (1996).

Consultant Curriculum Vitae

- j32 N. Nassif and N. Bagherzadeh, "A grid embedding into the star graph for image analysis solutions," *Information Processing Letters*, Vol. 60, No. 5, pp 255-260 (1996).
- j33 S. Latifi, N. Bagherzadeh, and R. Gajjala, "Fault-tolerant embedding of linear arrays and rings in the star graph," *International Journal of Computers and Electrical Engineering*, Vol 23, No. 2, pp 95-107 (1997).
- j34 S. Latifi and N. Bagherzadeh, "On embedding rings into a star-related network," *Information Sciences*, Vol. 99, No. 1-2, pp 21-35 (1997)
- j35 S. Shoari, N. Bagherzadeh, T. Milner, and J. S. Nelson, "Moment algorithms for blood vessel detection in infrared images of laser-heated skin," *International Journal of Computers and Electrical Engineering*, Vol 23, No. 5, January (1998).
- j36 M. M. de Azevedo, N. Bagherzadeh, M. Dowd, and S. Latifi, "Average distance and routing algorithms in the star-connected cycles interconnection network," *International Journal of Science and Technology-Special Issue of Computer in Scientia Iranica*, Vol. 3, No. 4, (1997).
- j37 N. Bagherzadeh, M. Dowd, and S. Latifi, "Faster Column Operations in Star Networks," *Telecommunication Systems*, Vol. 10, Nos. 1, 2 (1998).
- j38 M. M. de Azevedo, N. Bagherzadeh, and S. Latifi, "Low expansion packings and embeddings of hypercubes into star graphs: a performance-oriented approach," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 9, No. 3, (1998).
- j39 S. Shoari, N. Bagherzadeh, D. Goodman, T.E. Milner, D. J. Smithies, and J.S. Nelson, "A parallel algorithm for pulsed laser infrared tomography," *Pattern Recognition Letters*, 19 (1998).
- j40 S. Wallace and N. Bagherzadeh, "A scalable register file architecture for superscalar processors," *Microprocessors and Microsystems*, Vol. 22, No. 1, (1998).
- j41 S. Wallace and N. Bagherzadeh, "Modeled and measured instruction fetching performance for superscalar microprocessors," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 9, No. 6, (1998).
- j42 N. Nassif and N. Bagherzadeh, "Image component labeling on the star graph using divide and conquer," *IEE Proceedings-Computers and Digital Techniques*, Vol. 145, No. 1, (1998).
- j43 M. Lee, G. Lu, H. Singh, N. Bagherzadeh, F. Kurdahi, E. Filho, and V. Castro Alves, "MorphoSys A Reconfigurable Computing Chip," *IEEE Circuits and Systems Newsletter*, Vol. 10, No 1, March/April 1999.

Consultant Curriculum Vitae

- j44 M. Lee, H. Singh, G. Lu, N. Bagherzadeh, F. Kurdahi, E. Filho, and V. Castro Alves, "Design and Implementation of the MorphoSys Reconfigurable Computing Processor," *Journal of Signal Processing Systems for Signal, Image, and Video technology*, Vol. 24, No. 2/3, pp. 147-164, March 2000.
- j45 H. Singh, M. Lee, G. Lu, F. Kurdahi, N. Bagherzadeh, and E. Filho, "MorphoSys: An Integrated Reconfigurable System for Data-Parallel and Computation-Intensive Applications," *IEEE Transactions on Computers*, Vol. 49, No. 5, pp. 465-481 (2000).
- j46 R. Maestre, F. Kurdahi, M. Fernandez, R. Hermida, N. Bagherzadeh, and H. Singh, "Kernel scheduling techniques for efficient solution of space exploration in reconfigurable computing," *EUROMICRO Journal of Systems Architecture on Modern Methods and Tools in Digital System Design*, 2000.
- j47 R. Maestre, F. Kurdahi, M. Fernandez, R. Hermida, N. Bagherzadeh, and H. Singh, "A Formal Approach to Context Scheduling for Multi-context Reconfigurable Architectures," *IEEE Transactions on VLSI Design-Special Issue on Reconfigurable Computing*, Vol. 9, Issue 1, pp. 173-185, Feb 2001.
- j48 R. Maestre, F. Kurdahi, M. Fernandez, R. Hermida, N. Bagherzadeh, and H. Singh, "A Framework for Reconfigurable Computing: Task Scheduling and Context Management," *IEEE Transactions on VLSI Design-Special Issue on System Level Design*, Vol. 9, Issue 6, pp. 858-873, Dec 2001 (Best Paper Award).
- j49 P. Chou, J. Liu, D. Li, and N. Bagherzadeh, "IMPACCT: Methodology and Tools for Power-Aware Embedded Systems," *Kluwer Design Automation of Embedded Systems*, pp. 205-232, October 2002.
- j50 J. Liu, P. Chou, and N. Bagherzadeh, "Power-Aware Task Motion for Enhancing Dynamic Range of Embedded Systems with Renewable Energy Sources in Proceedings of the Workshop on Power Aware Computer Systems," in *Lecture Notes in Computer Science (LNCS)*, Springer-Verlag, pp. 84--99. 2002.
- j51 R. Maestre, F. Kurdahi, M. Fernandez, R. Hermida, N. Bagherzadeh, and H. Singh, "A Framework for Reconfigurable Computing: Task Scheduling and Context Management-a Summary," *Circuits and Systems Magazine*, IEEE, Vol. 2, Issue 4, pp. 48-51, 2002.
- j52 M. Sanchez-Elez, H. Du, N. Tabrizi, Y. Lung, N. Bagherzadeh, and M. Fernandez, "Algorithm Optimizations and Mapping Scheme for Interactive Ray Tracing on a Reconfigurable Architecture," *Computers & Graphics*, 2003
- j53 G. Venkataramani, W. Najjar, F. Kurdahi, N. Bagherzadeh, W. Bohm and J. Hammes, "Automatic Compilation to a Coarse-grained Reconfigurable System-on-Chip," *ACM Trans. on Embedded Computing Systems*, November 2003.

Consultant Curriculum Vitae

- j54 B. Gorjiara, N. Bagherzadeh, and P. Chou, "Ultra-fast and Efficient Algorithm for Energy Optimization by Gradient-based Stochastic Voltage and Task Scheduling," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, Volume 12 , Issue 4 (September 2007)
- j55 J. Bahn, S. Lee, and N. Bagherzadeh, "Design of a Router for Network-on-Chip," *International Journal of High Performance Systems Architecture*, 2007 - Vol. 1, No.2 pp. 98 - 105
- j56 A. Niktash, H. Parizi, and N. Bagherzadeh, "A Reconfigurable Processor for Forward Error Correction," *ARCS 2007*, LNCS 4415, pp. 1-13, 2007
- j57 J. Bahn and N. Bagherzadeh, "Design of Simulation and Analytical Models for a 2D-meshed Asymmetric Adaptive Router," *Computers & Digital Techniques, IET*, Volume 2, Issue 1, January 2008 Page(s):63 - 73
- j58 F. Rivera, M. Sanchez-Elez, R. Hermida, and N. Bagherzadeh, "Scheduling methodology for conditional execution of kernels onto multicontext reconfigurable architectures," *Computers & Digital Techniques, IET*, Volume 2, Issue 3, May 2008 Page(s):199 - 213
- j59 J. Bahn, S. Lee, Y. Yang, J. Yang, and N. Bagherzadeh, "On Design and Application Mapping of a Network-on-Chip (NoC) Architecture," *Parallel Processing Letters (PPL)*, Volume: 18, Issue: 2 (June 2008), pp. 239 - 255
- j60 N. Tabrizi and N. Bagherzadeh, "An ASIC Design and Formal Analysis of a Novel Pipelined and Parallel Sorting Accelerator," *Integration, The VLSI Journal*, Volume 41, Issue 1 (January 2008), Pages 65-75
- j61 S. Lee, J. Bahn, Y. Yang, and N. Bagherzadeh, "A Generic Network Interface architecture for a Networked Processor Array (NePA)," *ARCS 2008*, LNCS 4934, pp. 247–260, 2008
- j62 J. Bahn, J. Yang, W. Hu, and N. Bagherzadeh, "Parallel FFT Algorithms on Network-on-Chips," *Journal of Circuits, Systems, and Computers* 18(2): 255-269, 2009.
- j63 F. Bagci, F. Kluge, T. Ungerer, and N. Bagherzadeh, "Optimisations for LocSens - an Indoor Location Tracking System Using Wireless Sensors," *International Journal of Sensor Networks (IJSNET)*, Vol. 6, Nos. 3/4, 2009.
- j64 J. Bahn and N. Bagherzadeh, "On-Chip Interconnection Network with an Efficient Parallel Buffer Structure and Generic Traffic," *Scientia Iranica, Computer Science & Engineering and Electrical Engineering*. Vol. 16, No. 2, pp. 104-118, 2009.

Consultant Curriculum Vitae

- j65 M. Sanchez-Elez, N. Bagherzadeh, and R. Hermida, "A Framework for Low Energy Data Management in Reconfigurable Multi-Context Architectures," *Journal of Systems Architecture*, 2009, Pages 127-139
- j66 S. Lee and N. Bagherzadeh, "A Variable Frequency Link for a Power-Aware Network-on-Chip (NoC)," *INTEGRATION, the VLSI Journal*, 42 (2009) 479–485
- j67 S. Lee and N. Bagherzadeh, "A High Level Power Model for Network-on-Chip (NoC) Router," *Computers and Electrical Engineering*, 35 (2009) 837–845
- j68 Y. Yang, J. Bahn, S. Lee, J. Yang, and N. Bagherzadeh, "Parallel Processing for Block Ciphers on a Fault Tolerant Networked Processor Array," *International Journal of High Performance Systems Architecture*, Vol. 2, Nos. 3/4, 2010
- j69 J. Yang, S. Lee, C. Chen, and N. Bagherzadeh, "Ray Tracing on a Networked Processor Array," *International Journal of Electronics*, 2010, 97(10):1193-1205
- j70 F. Bagci, T. Ungerer, and N. Bagherzadeh, "Multi-level Security in Wireless Sensor Networks," *International Journal on Advances in Software*, Vol. 4, No. 6, 2010.
- j71 C. Wang, W. Hu, S. Lee, and N. Bagherzadeh, "Area and Power-Efficient Innovative Congestion-Aware Network-on-Chip," *Journal of Systems Architecture*, Volume 57 Issue 1, Pages 24-38, January, 2011
- j72 A. Hatanaka and N. Bagherzadeh, "A Scheduling Approach for Distributed Resource Architectures with Scarce Communication Resources," *Int. J. High Performance Systems Architecture*, Vol. 3, No. 1, Pages 12-22, 2011
- j73 C. Wang and N. Bagherzadeh, "High-throughput Differentiated Service Provision Router Architecture for Wireless Network-on-Chip," *Int. J. High Performance Systems Architecture*, Vol. 4, No. 1, Pages 38-56, 2012
- j74 F. Bolanos, J. Aedo, F. Rivera, and N. Bagherzadeh, "Mapping and Scheduling in Heterogeneous NoC through Population-Based Incremental Learning," *Journal of Universal Computer Science*, vol. 18, no. 7 (2012), 901-916
- j75 W. Hu, C. Chen, J. Bahn, and N. Bagherzadeh, "Parallel Low-Density Parity Check Decoding on a Network-on-Chip-Based Multiprocessor Platform," *IET Computers & Digital Techniques*, Vol. 6, Issue. 2, pp. 86–94, 2012
- j76 A. Hatanaka and N. Bagherzadeh, "A Software Pipelining Algorithm of Streaming Applications with Low Buffer Requirements," *Transactions D: Computer Science & Engineering and Electrical Engineering*, Vol. 19, No. 3, Pages 627-634, 2012
- j77 H. Abdullah, H. A. Elsadek, H. E. ElDeeb, and N. Bagherzadeh, "Fractional Derivatives Based Scheme for FDTD Modeling of nth-Order Cole–Cole Dispersive

Consultant Curriculum Vitae

- Media," *IEEE Antennas and Wireless Propagation Letters*, Vol. 11, Pages 281-284, 2012
- j78 A. Alhussien, C. Wang, and N. Bagherzadeh, "Design and Evaluation of a High Throughput Robust Router for Network-on-Chip," *IET Computers & Digital Technique*, Vol. 6, Issue. 3, pp. 173-179, 2012
- j79 C. Wang, W. Hu, and N. Bagherzadeh, "A Load-Balanced Congestion-Aware Wireless Network-on-Chip Design for Multi-Core Platforms," *Microprocessors and Microsystems*, Vol. 36, Issue. 7, pp. 555-570, 2012
- j80 S. Azampanah, A. Khademzadeh, N. Bagherzadeh, M. Janidarmian, and R. Shojaei "Contention-Aware Selection Strategy for Application-Specific Network-on-Chip," *IET Computers & Digital Technique*, Vol. 7, Issue. 3, pp. 105-114, 2013
- j81 F. Bolanos, F. Rivera, J.E. Aedo, and N. Bagherzadeh, "From UML Specifications to Mapping and Scheduling of Tasks into a NoC, with Reliability Considerations," *Journal of Systems Architecture*, Vol. 59, Issue 7, pp. 429-440, 2013
- j82 X. Wang, M. Yang, Y. Jiang, M. Palesi, P. Liu, T. Mak, and N. Bagherzadeh, "Efficient Multicast Schemes for 3-D Networks-on-Chip," *Journal of Systems Architecture*, Vol. 59, Issue 9, pp. 693-708, 2013
- j83 A. Alhussien, F. Verbeek, B. van Gastel, N. Bagherzadeh, and J. Schmaltz, "Fully Reliable Dynamic Routing Logic for a Fault-Tolerant NoC Architecture," *Journal of Integrated Circuits and Systems*, Vol. 8, Issue. 1, pp. 43-53, 2013
- j84 C. Wang, W. Hu, and N. Bagherzadeh, "Scalable Load Balancing Congestion-Aware Network-on-Chip Router Architecture," *Journal of Computer and System Sciences*, Vol. 79, Issue. 4, pp. 421-439, 2013
- j85 A. Demiriz, N. Bagherzadeh, and A. Alhussein, "Using Constraint Programming for the Design of Network-on-Chip Architectures," *Computing (Springer)*, DOI 10.1007/s00607-013-0359-4, 2013
- j86 C. Wang and N. Bagherzadeh, "Design and Evaluation of a High Throughput QoS-Aware and Congestion-Aware Router Architecture for Network-on-Chip," *Microprocessors and Microsystems*, Vol. 38, Issue. 4, pp. 304-315, 2014
- j87 A. Del Barrio, N. Bagherzadeh, and R. Hermida, "Ultra-Low-Power Adder Stage Design for Exascale Floating Point Units," *ACM Transactions on Embedded Computing Systems(TECS)*, Vol. 13, Issue. 3, Article No. 105, March 2014
- j88 R. Mirzaee, K. Navi, and N. Bagherzadeh, "High-Efficient Circuits for Ternary Addition," *VLSI Design*, Volume 2014, Article ID 534587, pp.1-15, 2014

Consultant Curriculum Vitae

- j89 S. Angizi, E. Alkaldy, N. Bagherzadeh, and K. Navi, "Novel Robust Single Layer Wire Crossing Approach for Exclusive OR Sum of Products Logic Design with Quantum-Dot Cellular Automata," *Journal of Low Power Electronics*, Vol. 10, No. 2, pp. 259-271, 2014
- j90 A. Demiriz, N. Bagherzadeh, and O. Ozturk, "Voltage Island Based Heterogeneous NoC Design Through Constraint Programming," *Computers and Electrical Engineering (Elsevier)*, 40 (2014) 307-316
- j91 W. Hu, C. Wang, and N. Bagherzadeh, "Design and Analysis of a Mesh-Based Wireless Network-on-Chip," *Journal of Supercomputing (Springer)*, DOI 10.1007/s11227-014-1341-4, 22 November, 2014
- j92 F. Sharifi, M. H. Moaiyeri, K. Navi, and N. Bagherzadeh, "Quaternary Full Adder Cells Based on Carbon Nanotube FETs," *Journal of Computer. Electronics (Springer)*, DOI 10.1007/s10825-015-0714-0, 2015
- j93 M. Khayambashi, P. Yaghini, A. Eghbal, and N. Bagherzadeh, "Analytical Reliability Analysis of 3D NoC under TSV Failure," *ACM Journal on Emerging Technologies in Computing Systems*, Vol. 11, No. 4, Article 43, April 2015
- j94 H. Karkhaneh, J-M. Liu, A. Ghorbani, and N. Bagherzadeh, "Fiber Dispersion Effects in Injection-Locked Optical OFDM Systems," *Optical and Quantum Electronics (Springer)*, DOI 10.1007/s11082-015-0197-z (2015)
- j95 P. Yaghini, A. Eghbal, S. Yazdi, N. Bagherzadeh, and M. Green, "Capacitive and Inductive TSV-to-TSV Resilient Approaches for 3D ICs," *IEEE Transactions on Computers*, Vol. 65, No. 3, pp. 693-705, March 2016, Digital Object Identifier no. 10.1109/TC.2015.2498550
- j96 A. Demiriz, N. Bagherzadeh, and A. Alhussein, "Using Constraint Programming for the Design of Network-on-Chip Architectures," *Computing (Springer)*, (2015) 97:579–592, DOI 10.1007/s00607-013-0359-4
- j97 P. Yaghini, A. Eghbal, M. Khayambashi, and N. Bagherzadeh, "Coupling Mitigation in 3-D Multiple-Stacked Devices," *IEEE Transactions on VLSI Systems*, Vol. 23, Issue: 12, 2015
- j98 S. Angizi, F. Danehdaran, S. Sarmadi, N. Bagherzadeh, and K. Navi, "An Ultra-High Speed and Low Complexity Quantum-Dot Cellular Automata Full Adder," *Journal of Low Power Electronics*, Vol. 11, No. 2, pp. 173-180(8), 2015
- j99 A. Eghbal, P. Yaghini, N. Bagherzadeh, and M. Khayambashi, "Analytical Fault Tolerance Assessment and Metrics for TSV-Based 3D Network-on-Chip," *IEEE Transactions on Computers*, Vol. 64, No. 12, pp. 3591-2604, December (2015); Digital Object Identifier no. 10.1109/TC.2015.2401016

Consultant Curriculum Vitae

- j100 S. Angizi, M. Moaiyeri, S. Farrokhi, K. Navi, and N. Bagherzadeh, "Designing Quantum-dot Cellular Automata Counters with Energy Consumption Analysis," *Microprocessors and Microsystems*, 2015, DOI: <http://dx.doi.org/10.1016/j.micpro.2015.07.011>
- j101 S. Angizi, S. Sayedsalehi, A. Roohi, N. Bagherzadeh, and K. Navi, "Design and Verification of New n-Bit Quantum-Dot Synchronous Counters Using Majority Function-Based JK Flip-Flops," *Journal of Circuits, Systems, and Computers*, Vol. 24, No. 10 (2015) 1550153 (17 pages); DOI: 10.1142/S0218126615501534
- j102 R. Mirzaee, M. Jasemi, A. Valizadeh, and N. Bagherzadeh, "On the Design of Fully Symmetrical Bridge-Style Circuits," *IETE Journal of Research*, (2015); DOI: 10.1080/03772063.2015.1065712
- j103 M. Jasemi, R. Mirzaee, K. Navi, and N. Bagherzadeh, "Voltage Mirror Circuit by Carbon Nanotube Field Effect Transistors for Mirroring Dynamic Random Access Memories in Multiple-Valued Logic and Fuzzy Logic," *IET Circuits, Devices, and Systems*, Vol. 9, Iss5, pp. 343-352, (2015); DOI: 10.1049/iet-cds.2014.0295
- j104 F. Sharifi, M. H. Moaiyeri, K. Navi, and N. Bagherzadeh, "Robust and Energy Efficient Carbon Nanotube FET-Based MVL Gates: A Novel Design Approach," *Microelectronics Journal (Elsevier)*, 46 (2015) 1333-1342.
- j105 P. Yaghini, A. Eghbal, S. Yazdi, N. Bagherzadeh, "On the Design of Hybrid Routing Mechanism for Mesh-Based Network-on-Chip," *INTEGRATION, the VLSI journal*, 50 (2015) 183-192, <http://dx.doi.org/10.1016/j.vlsi.2014.12.006>
- j106 M. Moaiyeri, N. Khastoo, M. Nasiri, K. Navi, and N. Bagherzadeh "An Efficient Analog-to-Digital Converter Based on Carbon Nanotube FETs," *Journal of Low Power Electronics*, Vol. 12, 150-157, 2016. doi:10.1166/jolpe.2016.1432
- j107 F. Sharifi, M. Moaiyeri, K. Navi, and N. Bagherzadeh, "Ultra-Low-Power Carbon Nanotube FET-Based Quaternary Logic Gates," *International Journal of Electronics*, Vol. 103, No. 9, pp. 1524-1537, 2016, <http://dx.doi.org/10.1080/21681724.2016.1138506>
- j108 M. Agyeman, A. Ahmadiania, and N. Bagherzadeh, "Performance and Energy Aware Inhomogeneous 3D Networks-on-Chip Architecture Generation," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 27, No. 6, pp. 1756-1769, June 2016, Digital Object Identifier no. 10.1109/TPDS.2015.2457444
- j109 H. Shabani, A. Roohi, A. Reza, M. Reshadi, N. Bagherzadeh, and R. DeMara, "Loss-Aware Switch Design and Non-Blocking Detection Algorithm for Intra-Chip Scale Photonic Interconnection Networks," *IEEE Transactions on Computers*, Vol. 65, No. 6, pp. 1789-1801, June 2016, Digital Object Identifier no. 10.1109/TC.2015.2458866

Consultant Curriculum Vitae

- j110 M. Daliri, K. Navi, R. Mirzaee, S. Daliri, and N. Bagherzadeh "A New Approach for Designing Compressors with a New Hardware-Friendly Mathematical Method for Multi-Input XOR gates," *IET Circuits, Devices, and Systems*, Vol. 11, Iss. 1 pp. 46-57, 2016, Doi: 10.1049/iet-cds.2016.0041
- j111 M. Daliri, R. Mirzaee, K. Navi, and N. Bagherzadeh "Ternary Cyclic Redundancy Check by a New Hardware-Friendly Ternary Operator," *Microelectronics Journal*, pp. 126-137, 2016, <http://dx.doi.org/10.1016/j.mejo.2016.04.018>
- j112 R. Salamat, M. Khayambashi, M. Ebrahimi, and N. Bagherzadeh "A Resilient Routing Algorithm with Formal Reliability Analysis for Partially Connected 3D-NoCs," *IEEE Transactions on Computers*, Vol. 65, No. 11, pp. 3265-3279, 2016, Digital Object Identifier no. 10.1109/TC.2016.2532871
- j113 F. Sharifi, A. Panahi, H. Sharifi, K. Navi, N. Bagherzadeh, and H. Thaliyal, "Design of Quaternary 4–2 and 5–2 Compressors for Nanotechnology," *Computers and Electrical Engineering*, pp. 64-74, 2016, <http://dx.doi.org/10.1016/j.compeleceng.2016.11.006>
- j114 A. Bozorgmehr, M. Moaiyeri, K. Navi, and N. Bagherzadeh "Ultra-Efficient Fuzzy Min/Max Circuits Based on Carbon Nanotube FETs," *IEEE Transactions on Fuzzy Systems*, 2017, DOI 10.1109/TFUZZ.2017.2697404
- j115 M. Daliri, R. Mirzaee, K. Navi, and N. Bagherzadeh "High-Performance Ternary Operators for Scrambling," *INTEGRATION the VLSI Journal*, 59 (2017), <http://dx.doi.org/10.1016/j.vlsi.2017.03.010>
- j116 H. Zhao, N. Bagherzadeh, and J. Wu, "A General Fault-Tolerant Minimal Routing for Mesh Architectures," *IEEE Transactions on Computers*, Vol. 66, No. 7, pp. 1240-1246, 2017, Digital Object Identifier no. 10.1109/TC.2017.2651828
- j117 Z. Ghaderi, M. Ebrahimi, Z. Navabi, E. Bozorgzadeh, and N. Bagherzadeh, "SENSIBLE: A Highly Scalable SENSOR DeSIGN for Path-Based Age Monitoring in FPGAs," *IEEE Transactions on Computers*, Vol. 66, No. 5, pp. 919-926, 2017, Digital Object Identifier no. 10.1109/TC.2016.2622688
- j118 M. Bagherian, M. Daliri, A. Abdoli, K. Navi, and N. Bagherzadeh, "A 3D Universal Structure Based on Molecular-QCA and CNT Technologies," *Journal of Molecular Structure*, MOLSTR 22439, (2016), 10.1016/j.molstruc.2016.04.025
- j119 A. Noghondar, M. Rishadi, and N. Bagherzadeh, "Reducing bypass-based network-on-chip latency using priority mechanism," *IET Computers & Digital Techniques*, pp. 1-8, 2017, doi: 10.1049/iet-cdt.2016.0161

Consultant Curriculum Vitae

- j120 M. Bagherian, M. Moaiyeri, S. Angizi, N. Bagherzadeh, and K. Navi, “Quantum-dot cellular automata circuits with reduced external fixed inputs,” *Microprocessors and Microsystems*, 50 (2017) 154-163, <http://dx.doi.org/10.1016/j.micpro.2017.03.009>
- j121 M. Berarzadeh, S. Mohammadyan, K. Navi, and N. Bagherzadeh, “A Novel Low Power Exclusive-OR via Cell Level-based Design Function in Quantum Cellular Automata,” *Journal of Computational Electronics (Springer)*, DOI 10.1007/s10825-017-0986-7, April, 2017
- j122 S. Tabrizchi, A. Panahi, F. Sharifi, K. Navi, and N. Bagherzadeh, “Method for Designing Ternary Adder Cells Based on CNFETs,” *IET Circuits, Devices & Systems*, doi: 10.1049/iet-cds.2016.0443, (2017)
- j123 F. Verbeek, P. Yaghini, A. Eghbal, and N. Bagherzadeh, “Deadlock Verification of Cache Coherence Protocols and Communication Fabrics,” *IEEE Transactions on Computers*, Vol. 66, No. 2, Digital Object Identifier no. 10.1109/TC.2016.2584060, February 2017
- j124 Z. Rouhani, S. Angizi, M. Taheri, K. Navi, N. Bagherzadeh, “Towards Approximate Computing with Quantum-Dot Cellular Automata,” *Journal of Low Power Electronics*, Vol. 13, No. 1, pp 29-35, doi:10.1166/jolpe.2017.1475, 2017
- j125 Z. Ghaderi, N. Bagherzadeh, and A. Albaqsami, “STABLE: Stress-Aware Boolean Matching to Mitigate BTI-Induced SNM Reduction in SRAM-Based FPGAs,” *IEEE Transactions on Computers*, Vol. 67, No. 1, Digital Object Identifier no. 10.1109/TC.2017.2725952, 2017
- j126 M. Bagherian, M. Moaiyeri, K. Navi, and N. Bagherzadeh, “An Energy and Cost Efficient Majority-based RAM Cell in Quantum-dot Cellular Automata,” *Results in Physics (Elsevier)*, <https://doi.org/10.1016/j.rinp.2017.08.067>, 2017
- j127 A. Arasteh, M. Moaiyeri, M. Taheri, K. Navi, and N. Bagherzadeh, “An Energy and Area Efficient 4:2 Compressor Based on FinFETs,” *Integration the VLSI Journal*, DOI: 10.1016/j.vlsi.2017.09.010, (60) October 2017
- j128 S. Angizi, Z. He, N. Bagherzadeh, and D. Fan, “Design and Evaluation of a Spintronic In-Memory Processing Platform for Nonvolatile Data Encryption,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 37. No. 9, Digital Object Identifier 10.1109/TCAD.2017.2774291, September 2018
- j129 F. Khubrani, N. Bagherzadeh, and N. Meshkati, “Importance of Safety Culture at Pre-Operational Phase to Prevent Nuclear Plant Accidents,” *International Journal of Atomic and Nuclear Physics*, 2017
- j130 R. Salamat, M. Khayambashi, M. Ebrahimi, and N. Bagherzadeh, “LEAD: An Adaptive 3D-NoC Routing Algorithm with Queuing-theory Based Analytical

Consultant Curriculum Vitae

- Verification,” *IEEE Transactions on Computers*, Vol. 67, No. 8, Digital Object Identifier no. 10.1109/TC.2018.2801298, August 2018
- j131 R. Mirzaee, M. Daliri, K. Navi, and N. Bagherzadeh, “A Single Parity-Check Digit for One Trit Error Detection in Ternary Communication Systems: Gate-Level and Transistor-Level Designs,” *Journal of Multiple-Valued Logic and Soft Computing*, 29 3-4), August 2017
- j132 M. Agyeman, A. Ahmadiania, and N. Bagherzadeh, “Energy and Performance-aware Application Mapping for Inhomogeneous 3D Networks-on-Chip,” *Journal of Systems Architecture* 89 (2018) 103-117, <https://doi.org/10.1016/j.sysarc.2018.08.002>
- j133 F. Khodaparast, M. Reshadi, and N. Bagherzadeh, “Application Partitioning and Mapping for Bypass Channel Based NoC,” *Computers and Electrical Engineering (Elsevier)*, 71 (2018) 676-691, <https://doi.org/10.1016/j.compeleceng.2018.08.016>
- j134 F. Danehdaran, M. Bagherian, K. Navi, and N. Bagherzadeh, “Design and Power Analysis of New Coplanar One-Bit Full-Adder Cell in Quantum-Dot Cellular Automata,” *Journal of Low Power Electronics*, Vol. 14, 1-11, 2018
- j135 A. Alqahtani, Z. Ren, J. Lee, and N. Bagherzadeh, “System-Level Analysis of 3D ICs with Thermal TSVs,” *ACM Journal on Emerging Technologies in Computing Systems*, Vol. 14, No. 3, Article 37. Pub. date: October 2018, <https://doi.org/10.1145/3264736>
- j136 A. Haghighathoseini, H. Bobarshad, F. Saghafi, M. Rezaei, and N. Bagherzadeh, “Hospital Enterprise Architecture Framework (Study of Iranian University Hospital Organization),” *International Journal of Medical Informatics*, <https://doi.org/10.1016/j.ijmedinf.2018.03.009>
- j137 Z. Ghaderi, A. Alqahtani, and N. Bagherzadeh, “AROMA: Aging-Aware Deadlock-Free Adaptive Routing Algorithm and Online Monitoring in 3D NoCs,” *IEEE Transactions on Parallel and Distributed Systems*, Vol. 29, No. 4, April 2018, Digital Object Identifier no. 10.1109/TPDS.2017.2780173
- j138 A. Charif, A. Coelho, M. Ebrahimi, N. Bagherzadeh, and N. Zergainoh, “First-Last: A Cost-Effective Adaptive Routing Solution for TSV-Based Three-Dimensional Networks-on-Chip,” *IEEE Transactions on Computers*, Vol. 67, No. 10, October 2018, Digital Object Identifier no. 10.1109/TC.2018.2822269
- j139 M. Kim, A. Del Barrio, L. Tavares Oliveira, R. Hermida, and N. Bagherzadeh, “Efficient Mitchell’s Approximate Log Multipliers for Convolutional Neural Networks,” *IEEE Transactions on Computers*, Vol. 68, No. 5, May 2019, Digital Object Identifier no. 10.1109/TC.2018.2880742
- j140 A. Shahkarami, H. Bobarshad, and N. Bagherzadeh, “A Stream-Sensitive Distributed Approach for Configuring Cascaded Classifier Topologies in Real-Time Large-Scale

Consultant Curriculum Vitae

- Stream Mining Systems,” *SN Applied Sciences*, (2019), <https://doi.org/10.1007/s42452-019-0565-6>
- j141 M. Torabzadekashi, S. Rezaei, A. Heydari, H. Bobarshad, V. Alves, and N. Bagherzadeh, “Computational Storage: an Efficient and Scalable Platform for Big Data and HPC Applications,” *Journal of Big Data*, (2019) 6:100, <https://doi.org/10.1186/s40537-019-0265-5>
- j142 A. Mehranzadeh, A. Khademzadeh, N. Bagherzadeh, and M. Rishadi, “DICA: Destination Intensity and Congestion-aware Output Selection Strategy for Network-on-Chip Systems,” *IET Computers & Digital Techniques*, doi: 10.1049/iet-cdt.2018.5154, 2020.
- j143 S. Tabrizchi, M. Taheri, K. Navi, and N. Bagherzadeh, “Novel CNFET Ternary Circuit Techniques for High-Performance and Energy-Efficient Design,” *IET Circuits, Devices & Systems*, doi: 10.1049/iet-cds.2018.5036
- j144 M. Zahmatkesh, S. Tabrizchi, S. Mohammadyan, K. Navi, and N. Bagherzadeh, “Robust Coplanar Full Adder Based on Novel Inverter in Quantum Cellular Automata,” *International Journal of Theoretical Physics* (2019) 58:639–655, <https://doi.org/10.1007/s10773-018-3961-6>
- j145 M. Soleymani, M. Reshadi, and N. Bagherzadeh, “CLBM: Controlled Load-Balancing Mechanism for Congestion Management in Silicon Interposer NoC Architecture,” *Journal of Systems Architecture*, <https://doi.org/10.1016/j.sysarc.2019.07.003>, 98 (2019) 102-113
- j146 F. Danehdaran, S. Angizi, M. Bagherian, K. Navi, and N. Bagherzadeh, “A combined three and five inputs majority gate-based high performance coplanar full adder in quantum-dot cellular automata,” *Int. j. inf. tecnol.*, <https://doi.org/10.1007/s41870-019-00365-z>
- j147 L. Garcia Morales, J.E. Aedo Cobo, and N. Bagherzadeh, “A New Approach to the Population-Based Incremental Learning algorithm using Virtual Regions for Task Mapping on NoCs,” *Journal of Systems Architecture*, <https://doi.org/10.1016/j.sysarc.2019.01.013>, p. 443-454, 2019
- j148 M. Toulabinejad, M. Taheri, K. Navi, and N. Bagherzadeh, “Toward Efficient Implementation of Basic Balanced Ternary Arithmetic Operations in CNFET Technology,” *Microelectronics Journal* (Elsevier), 90 (2019) 267-277, <https://doi.org/10.1016/j.mejo.2019.05.010>
- j149 S. Nikbakht and N. Bagherzadeh, “Divisible load scheduling of image processing applications on the heterogeneous star and tree networks using a new genetic algorithm,” *Concurrency and Computation Practice and Experience* (Wiley), (2019), <https://doi.org/10.1002/cpe.5498>

Consultant Curriculum Vitae

- j150 P. Sadri, M. Wahiduzzaman, S. Isklam, I. Krivorotov, M. Nilsson, N. Bagherzadeh, and O. Boyraz, "Array of Symmetric Nanohole Dimers with High Sensitivity for Detection of Changes in an STT-RAM Ultrathin Dielectric Layer," *Journal of the Optical Society of America B*, Vol. 36 No. 11, (2019), <https://doi.org/10.1364/JOSAB.36.003090>
- j151 M. Moghadam, E. Masoumi, N. Bagherzadeh, D. Ramsingh, G.P. Li, and Z. Kain, "A Machine-Learning Approach to Predicting Hypotensive Events in ICU Settings," *Computers in Biology and Medicine*, <https://doi.org/10.1016/j.combiomed.2020.103626>, 2020.
- j152 H. Zhao, N. Bagherzadeh, Q. Wang, and Y. Wang, "A Fine-Grained Source-Throttling Method for Mesh Architectures," *IEEE Access, Special Section on Artificial Intelligence in Parallel and Distributed Computing*, Volume 8, 2020, Digital Object Identifier 10.1109/ACCESS.2020.2967459
- j153 A. Albaqsami, M. Hosseini, M. Jasemi, and N. Bagherzadeh, "Adaptive HTF-MPR: An Adaptive Heterogeneous TensorFlow Mapper Utilizing Bayesian Optimization and Genetic Algorithms," *ACM Trans. Intell. Syst. Technol.*, 11, 5, Article 55 (August 2020), <https://doi.org/10.1145/3396949>
- j154 A. Bozorgmehr, M. Khaleqi, M. Moaiyeri, K. Navi, and N. Bagherzadeh, "A high-performance fully programmable membership function generator based on 10 nm gate-all-around CNTFETs," *International Journal of Electronics and Communications*, <https://doi.org/10.1016/j.aeue.2020.153293>
- j155 E. Montoya, J. Chen, R. Ngelale, H. Lee, H. Tseng, L. Wan, E. Yang, P. Braganca, O. Boyraz, N. Bagherzadeh, M. Nilsson, and I. Krivorotov, "Immunity of Nanoscale Magnetic Tunnel Junctions with Perpendicular Magnetic Anisotropy to Ionizing Radiation," *Scientific Reports (Nature Research)*, <https://doi.org/10.1038/s41598-020-67257-2>, (2020)
- j156 M. Jasemi, S. Hessabi, and N. Bagherzadeh, "Reliable and Energy Efficient MLC STT-RAM Buffer for CNN Accelerators," *Computers and Electrical Engineering*, <https://doi.org/10.1016/j.compeleceng.2020.106698>
- j157 J. Talafy, F. Zokaee, H. Zarandi, and N. Bagherzadeh, "A High Performance Multi-Bit Output Logic-in-Memory Adder," *IEEE Transactions on Emerging Topics in Computing*, DOI 10.1109/TETC.2020.2982951,
- j158 P. Sadri, M. Wahiduzzaman, M. Shafiqul Islam, E. Montoya, I. Krivorotov, N. Bagherzadeh, and O. Boyraz, "Effect of Magnesium Oxide Adhesion Layer on Resonance Behavior of Plasmonic Nanostructures," *Applied Physics Letters*, 116, 241601 (2020); <https://doi.org/10.1063/5.0008665>

Consultant Curriculum Vitae

- j159 Z. Ren, A. Alqahtani, N. Bagherzadeh, and J. Lee, "Thermal TSV Optimization and Hierarchical Floorplanning for 3D Integrated Circuits," *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, <https://doi.org/10.1109/TCPMT.2020.2970382>, Volume: 10, Issue: 4, April 2020
- j160 A. Bozorgmehr, M. Khaleqi, M. Moaiyeri, K. Navi, and N. Bagherzadeh, "A Novel Digital Fuzzy System for Image Edge Detection Based on Wrap-Gate Carbon Nanotube Transistors," *Computers and Electrical Engineering*, 87 (2020) 106811, <https://doi.org/10.1016/j.compeleceng.2020.106811>
- j161 M. Jasemi, S. Hessabi, and N. Bagherzadeh, "Enhancing Reliability of Emerging Memory Technology for Machine Learning Accelerators," *IEEE Transactions on Emerging Topics in Computing*, DOI 10.1109/TETC.2020.2984992, (2020).
- j162 S. Hosseini, M. Reshadi, and Nader Bagherzadeh, "Flow Mapping on Mesh-Based Deep Learning Accelerator," *Journal of Parallel Distributed Computing*, 144 (2020) 80-97, <https://doi.org/10.1016/j.jpdc.2020.04.011>
- j163 A. Tajari, H. Zarandi, and N. Bagherzadeh, "IRHT: An SDC Detection and Recovery Architecture Based on Value Locality of Instruction Binary Codes," *Microprocessors and Microsystems* [Volume 77](https://doi.org/10.1016/j.micpro.2020.103159), September 2020, 103159, <https://doi.org/10.1016/j.micpro.2020.103159>
- j164 S. Hosseini, M. Reshadi, N. Bagherzadeh, and A. Khademzadeh, "Data Scheduling and Placement in Deep Learning Accelerator," *Cluster Computing*, <https://doi.org/10.1007/s10586-021-03355-8>, (2021)
- j165 M. Kim, A. Del Barrio, H. Kim, and N. Bagherzadeh, "The Effects of Approximate Multiplication on Convolutional Neural Networks," *IEEE Transactions on Emerging Topics in Computing*, Digital Object Identifier 10.1109/TETC.2021.3050989, (2021)
- j166 M. Taheri, N. Shafiee, F. Sharifi, M. Moaiyeri, K. Navi, and N. Bagherzadeh, "Energy Efficient Hybrid Full Adder Design for Digital Signal Processing in Nanoelectronics," *Analog Integrated Circuits and Signal Processing*, (2021), <https://doi.org/10.1007/s10470-021-01831-0>
- j167 R. Murillo, A. Del Barrio, G. Botella, M. S. Kim, H. Kim, and N. Bagherzadeh, "PLAM: A Posit Logarithm-Approximate Multiplier," *IEEE Transactions on Emerging Topics in Computing*, Digital Object Identifier, DOI 10.1109/TETC.2021.3109127, (2021)
- j168 M. Hosseini, M. Ebrahimi, P. Yaghini, and N. Bagherzadeh, "Near Volatile and Non-Volatile Memory Processing in 3D Systems," *IEEE Transactions on Emerging Topics in Computing*, Digital Object Identifier, <https://doi.org/10.1109/TETC.2021.3115495>, (2021)

Consultant Curriculum Vitae

- j169 S. Nabiee, M. Harding, J. Hersh, and N. Bagherzadeh, "Hybrid U-Net: Semantic Segmentation of High-resolution Satellite Images to Detect War Destruction," *Machine Learning with Applications*, <https://doi.org/10.1016/j.mlwa.2022.100381>, (2022)
- j170 H. Kim, M. Alnemari, and N. Bagherzadeh, "A Storage-Efficient Ensemble Classification Using Filter Sharing on Binarized Convolutional Neural Networks," *PeerJ Computer Science*. (2022), Vol. 8. <http://dx.doi.org/10.7717/peerj-cs.924>
- j171 A. Ding, Y. Qiao, and N. Bagherzadeh, "BNN an Ideal Architecture for Acceleration with Resistive in Memory Computation," *IEEE Transactions on Emerging Topics in Computing*, DOI 10.1109/TETC.2023.3237778, (2023)
- j172 M. Moghadam, E. Masoumi, S. Kendale, and N. Bagherzadeh, "Predicting hypotensive events in the ICU settings using patient's short-term physiological history and contextual data," *Computer Methods and Programs in Biomedicine* (2023) <https://doi.org/10.1016/j.cmpbup.2023.100100>

Journals (other)

- jo1 F. Mehdipour and N. Bagherzadeh, "New memory technology supports parallel processing," *Defense Electronics*, pp. 54-58, November 1990.
- jo2 S. Shoari, A. Kavianpour, and N. Bagherzadeh, "Pyramid simulation of image processing Applications," *Image and Vision Computing*, Vol. 12, No. 8, pp. 523-529, October 1994.
- jo3 F. Kurdahi, N. Bagherzadeh, P. Athanas, and J. Muniz, "Guest Editors' Introduction: Configurable Computing," *IEEE Design and Test of Computers: Special Issue on Configurable Computing*, March/April 2000.
- jo4 N. Bagherzadeh, "Guest Editor for Special Issue: Reconfigurable and Scalable High Performance Architectures," *International Journal of High Performance Systems Architecture*, Vol. 1, No. 2, 2007.
- jo5 H. Sarbazi-Azad and N. Bagherzadeh: Editorial notes: Special Issue on On-Chip Parallel and Network-Based Systems. *Microprocessors and Microsystems - Embedded Hardware Design* 36(7): 529-530 (2012)
- jo6 H. Sarbazi-Azad and N. Bagherzadeh: Multicore Computing Systems: Architecture, Programming Tools, and Applications, *Journal of Computer and Systems Sciences*, 79(4): 403-405 (2013)

Patents

Consultant Curriculum Vitae

- p1 A. Eghbal, P. Yaghini, and N. Bagherzadeh, "Three-Dimensional NoC Reliability Evaluation," US 11,093,673 B2, Date of Patent: August 17, 2021

Book Chapters

- b1 D. Blough and N. Bagherzadeh, "An efficient fault-tolerant routing algorithm for hypercube systems," *Advances in Parallel Computing*, Vol. 2, New York: North-Holland Press, 1989.
- b2 M. Villa-Uriol, F. Kuester, N. Bagherzadeh, A. Perez, and J. M. McCarthy, "Chapter on Humanoids and Biomedical Applications, Kinematics Synthesis of Avatar Skeletons from Visual Data," *On Advances in Robot Kinematics*, Kluwer Academic Publishers, Sestri Levante, Italy, pp. 171-180, 2004.
- b3 M. Sanchez-Elez, M. Fernandez, N. Bagherzadeh, R. Hermida, F. Kurdahi, and R. Maestre, "A Coarse-Grain Dynamically Reconfigurable System and Compilation Framework," *Vassliadis and Soudris, Eds.*, pp. 181-215, Springer, 2007.
- b4 S. Lee and N. Bagherzadeh, "Chapter 9: Energy/Power Issues in Network-on-Chip," *Network-on-Chips: Theory and Practice*, CRC Press, 2009, ISBN: 9781420079784.
- b5 F. Hosseinzadeh, N. Bagherzadeh, A. Khademzadeh and M. Janidarmian, "Fault-Tolerant Optimization for Application-Specific Network-on-Chip Architecture," *IAENG Transactions on Engineering Technologies, Lecture Notes in Electrical Engineering (Springer)*, Chapter 26, pp. 363-381, 2014.
- b6 F. Bolanos, F. Rivera, J. Aedo, and N. Bagherzadeh, "Task Mapping in Heterogeneous NoC by Means of Population-Based Incremental Learning," *Advances in Intelligent Systems: Reviews, Book Series, Vol. 1*, 2017
- b7 B. Barhami, M. Jamali, N. Bagherzadeh, and G. Morales, "Wireless Network-on-Chip Architecture A Comprehensive Review," *Horizons in Computer Science Research*, Nova Science Publishers, Inc., Volume 16, 2018.

Conferences Refereed

- c1 M. Lee, J. Fiene, C. Wu, N. Bagherzadeh, and G. Brown, "Network facility for a reconfigurable computer architecture," *Proceedings of 1985 Distributed Computing Symposium*, pp. 264-271, 1985.
- c2 C. Wu, M. Lee, C. Sudtikitpisan, J. Moaddeb, G. Brown, W. Lin, N. Bagherzadeh, and D. Vaughn, "Prototype of star architecture; a status report," *AFIPS Proceedings, NCC*, pp. 191-201, 1985.

Consultant Curriculum Vitae

- c3 N. Bagherzadeh, K. Tseng and C. Wu, "Erratic non-uniform traffic behavior of a multiple minicomputer network," *Proceedings of the International Symposium Mini and Microcomputer and Their Applications*, pp. 39-42, November 1986.
- c4 N. Bagherzadeh and C. Wu, "Distributed synchronous garbage collection using broadcasting," *Proceedings of the 20th Hawaii International Conference on System Sciences*, pp. 362-369, January 1987.
- c5 N. Bagherzadeh and C. Wu, "Annotations for parallelism," *Proceedings of the 6th Phoenix Conference on Computers and Communications*, pp. 346-350, February 1987.
- c6 N. Bagherzadeh, T. Kerola, W. Leddy, and R. Brice, "On parallel execution of the traveling salesman problem on a neural network model," *First International Conference on Neural Networks*, San Diego, California, pp. 317-324, June 21-24, 1987.
- c7 N. Bagherzadeh, B. Kandu, S. Heng, and C. Wu, "Network simulation of a synchronous garbage collection algorithm," *Symposium on the Simulation of Computer Networks*, Colorado Springs, Colorado, pp. 215-222, August 1987.
- c8 A. Guzman, E. Krall, P. McGehearty, and N. Bagherzadeh, "Effects of Applications on the performance of a parallel architecture," *HICSS-21*, January 1988.
- c9 Y. Moon, N. Bagherzadeh, and J. Sklansky, "Assignment of tasks to multiple processors for image processing," *Pattern Recognition for Advanced Missile System Conference*, Huntsville, Alabama, November 1988.
- c10 Y. Moon, N. Bagherzadeh, and J. Sklansky, "Macropipelined multicomputer systems for image analysis," *SPIE OE/LASE '89 Conference on High Speed Computing II*, Los Angeles, California, January 1989.
- c11 D. Blough, N. Bagherzadeh, and R. Sehgal, "A new fault-tolerant routing algorithm for hypercube systems," *Proceedings of the Third Annual Parallel Processing Symposium*, pp. 130-137, March 1989.
- c12 N. Nassif and N. Bagherzadeh, "A new method for distributed termination detection," *Proceedings of the Third Annual Parallel Processing Symposium*, pp. 300-310, March 1989.
- c13 D. Blough and N. Bagherzadeh, "A new fault-tolerant routing algorithm for hypercube systems," *International Workshop on Hardware Fault Tolerance in Multiprocessors*, University of Illinois, pp. 52-54, June 1989.
- c14 A. Mendez, N. Bagherzadeh, and F. Mehdipour, "Three dimensional mass memory compatible with parallel processing architectures," *Proceedings of the Fourth Annual Parallel Processing Symposium*, Fullerton, California, pp. 34-42, April 1990.

Consultant Curriculum Vitae

- c15 K. Tsai, K. Kim, and N. Bagherzadeh, "A hierarchical mesh architecture," *Proceedings of the Fourth Annual Parallel Processing Symposium*, Fullerton, California, pp. 459-477, April 1990.
- c16 A. Kavianpour and N. Bagherzadeh, "A three coloring algorithm for distributed termination detection of non-FIFO distributed systems," *Proceedings of the Fourth Annual Parallel Processing Symposium*, Fullerton, California, pp. 708-718, April 1990.
- c17 N. Bagherzadeh, "Broadcasting on faulty hypercubes," *Proceedings of the Fourth Annual Parallel Processing Symposium*, Fullerton, California, pp. 730-732, April 1990.
- c18 A. Abnous and N. Bagherzadeh, "Special features of a VLIW architecture," *Proceedings of the Fifth Annual Parallel Processing Symposium*, Fullerton, California, pp. 224-227, 1991.
- c19 A. Kavianpour and N. Bagherzadeh, "Parallel Hough Transform for image processing on a pyramid architecture," *Proceedings of the International Conference on Parallel Processing*, pp. 395-398, August 1991.
- c20 A. Abnous, R. Potasman, N. Bagherzadeh, and A. Nicolau, "A percolation based VLIW architecture," *Proceedings of the International Conference on Parallel Processing*, pp. 144-148, August 1991.
- c21 K. Tsai, Y. Kim, and N. Bagherzadeh, "Hypermesh: a combined quad tree and mesh network for parallel processing," *Proceedings of the International Conference on Parallel Processing*, pp. 696-697, August 1991.
- c22 N. Bagherzadeh and K. Hawk, "Parallel implementation of the auction algorithm on the Intel Hypercube," *Proceedings of the Sixth Annual Parallel Processing Symposium*, pp. 443-447, March 1992.
- c23 A. Abnous, C. Christensen, J. Gray, J. Lenell, A. Naylor and N. Bagherzadeh, "VLSI design of the Tiny RISC microprocessor," *Proceedings of the Custom Integrated Circuits Conference*, pp. 30.4.1-5, May 1992.
- c24 A. Abnous and N. Bagherzadeh, "Pipelining and bypassing in a VLIW processor," *Proceedings of the International Symposium on Computer Architecture*, p. 434, May 1992.
- c25 S. Latifi, S. Zheng, and N. Bagherzadeh, "Optimal ring embedding in hypercubes with faulty links," *Proceedings of the International Fault-Tolerant Computing Symposium*, pp. 178-184, July 1992.

Consultant Curriculum Vitae

- c26 T. Kato, K. Suginuma, and N. Bagherzadeh, "On design and performance analysis of a superscalar architecture," *Proceedings of the International Conference on Parallel Processing*, August 1992.
- c27 J. Lenell, S. Wallace and N. Bagherzadeh, "A 20MHz CMOS Reorder Buffer for a Superscalar Microprocessor," *Proceedings of the NASA VLSI Conference*, 1992.
- c28 T. Kato, T. Ono and N. Bagherzadeh, "Performance Analysis and Design Methodology for a Scalable Superscalar Architecture," *Proceedings of the MICRO-25 Conference*, Portland, Oregon, 1992.
- c29 T. Kato, T. Ono and N. Bagherzadeh, "A Novel Superscalar Architecture with Predetermined Scheduling Patterns," *Proceedings of the International Conference on Parallel and Distributed Systems*, Taiwan, 1992.
- c30 T. Kato, T. Ono and N. Bagherzadeh, "Data Cache Architecture of the Superscalar by Scheduling Patterns," *HICSS'26*, 1993.
- c31 J. Gray, A. Naylor, A. Abnous and N. Bagherzadeh, "VIPER: A 25-MHz, 100-MIPS Peak VLIW Microprocessor," *Proceedings of the Custom Integrated Circuits Conference*, 1993.
- c32 S. Latifi, M. de Azevedo and N. Bagherzadeh, "The Star Connected Cycles: A Fixed-Degree Network for Parallel Processing," *International Conference on Parallel Processing*, St. Charles, IL, August 1993.
- c33 A. Fijany, G. Kwan, and N. Bagherzadeh, "A fast algorithm for parallel computation of multibody dynamics on MIMD parallel architectures," *Proceedings of Computing in Aerospace 9 AIAA*, San Diego, California, October 1993.
- c34 S. Shoari, A. Kavianpour, and N. Bagherzadeh, "Pyramid simulation of image processing Applications," *Proceedings of Computing in Aerospace 9 AIAA*, San Diego, California, October 1993.
- c35 A. Kavianpour, N. Bagherzadeh, and S. Shoari, "Finding elliptical shapes in an image using a pyramid architecture," *International Symposium on Speech, Image Processing, and Neural Networks*, Hong Kong, 1994.
- c36 M. M. de Azevedo, S. Latifi, and N. Bagherzadeh, "On packing and embedding hypercubes into star graphs," *Proceedings of the VI Brazilian Symposium on Computer Architecture and High Performance Computing (VI SBAC-PAD)*, Caxambu, Minas Gerais, Brazil, August 1-5, pp. 3-19, (1994).
- c37 M. M. de Azevedo, N. Bagherzadeh, and S. Latifi, "Fault-diameter of the star-connected cycles interconnection network," *Proceedings of the 28th Annual Hawaii*

Consultant Curriculum Vitae

International Conference on System Sciences (HICSS-28), Vol. II (Software technology), Maui, Hawaii, January 3-6, pp. 469-478, 1995.

- c38 M. M. de Azevedo, N. Bagherzadeh, and S. Latifi, "Embedding meshes in the star-connected cycles interconnection network," *10th International Conference on mathematical and computer modeling and Scientific Computing* Boston, July 5-8 (1995).
- c39 S. Wallace, N. Dagli, and N. Bagherzadeh, "Design and implementation of a 100 Mhz Reorder Buffer," *37th Midwest Symposium on Circuits and Systems*, 1994.
- c40 M. M. de Azevedo, S. Latifi, and N. Bagherzadeh "Low expansion packings and embeddings of hypercubes into star graphs," *Proceedings of the IEEE 15th Annual International Phoenix Conference on Computers and Communications*, Phoenix, March 27-29, pp. 115-122 (1996).
- c41 M. M. de Azevedo, N. Bagherzadeh, and S. Latifi, "Variable-dilation embeddings of hypercubes into star graphs: performance metrics, mapping functions, and routing," *Euro-Par*, Lyon, France, 1996.
- c42 N. Nassif and N. Bagherzadeh, "Image analysis solutions and grid embedding on the star," *Proceedings of the 1996 SCS Simulation Multiconference*, New Orleans, April 8-11, pp. 396-402 (1996).
- c43 M. Gulati and N. Bagherzadeh, "Performance study of a multithreaded superscalar microprocessor," *Proceedings of the Second International Symposium on High-Performance Computer Architecture*, San Jose (California), February, pp. 291-301 (1996).
- c44 S. Wallace and N. Bagherzadeh, "Instruction fetching mechanism for superscalar microprocessors," *Euro-Par*, Lyon, France, 1996.
- c45 S. Wallace and N. Bagherzadeh, "A scalable register file architecture for dynamically scheduled processors," *PACT*, Boston, 1996.
- c46 M. Loikkanen and N. Bagherzadeh, "A fine-grain multithreading superscalar architecture," *PACT*, Boston, 1996.
- c47 K. Suginuma and Nader Bagherzadeh, "Z-cache, a new cache mechanism for computer graphics," *Proceedings of MTAC '97*.
- c48 S. Wallace and Nader Bagherzadeh, "Multiple branch and block prediction," *Proceedings of HPCA-3*, pp. 94-103 (1997).

Consultant Curriculum Vitae

- c49 S. Shoari, N. Bagherzadeh, D. Goodman, T. E. Milner, D. J. Smithies, and J. S. Nelson, "Parallel algorithm for pulsed laser infrared tomography," *Proceedings of the 7th Annual Meeting of the American Society for Laser Medicine and Surgery*, (1997)
- c50 S. Latifi and N. Bagherzadeh, "Incomplete star graph: an alternative interconnection network for massively parallel systems," *Proceedings of the 26th Hawaii International Conference on System Science*, Vol. II, pp. 660-667, Jan 1993.
- c51 S. Latifi and N. Bagherzadeh, "The clustered-star graph: a new topology for large interconnection networks," *7th International Parallel Processing Symposium*, pp. 514-518, April 1993.
- c52 J. Lenell and N. Bagherzadeh, "A performance comparison of several superscalar processor models with a VLIW processor," *7th International Parallel Processing Symposium*, pp. 44-48, April 1993.
- c53 N. Bagherzadeh, N. Nassif, and S. Latifi, "A routing scheme on faulty star graphs," *Pacific Rim International Symposium on Fault Tolerant Computing*, pp. 214-218, Australia, December 1993.
- c54 S. Wallace, N. Dagli, and N. Bagherzadeh, "Design and implementation of a 100 MHz centralized instruction window for a superscalar microprocessor," *Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers and Processors*, pp. 96-101, Austin, October 1994.
- c55 S. Latifi and N. Bagherzadeh, "Hamiltonicity of the clustered-star graph with embedding Applications," *PDPTA '96 International*, pp. 734-744.
- c56 H. Liu, F. Etemadi, G. Lu, N. Bagherzadeh, and F. Kurdahi "Implementation of an all-digital CAP transceiver," *Proceedings of the International Conference on Computer Simulation*, Sept 1-4, Lebanon, 1997.
- c57 H. Singh, M. Lee, G. Lu, F. Kurdahi, N. Bagherzadeh, T. Lang, R. Heaton, and F. Filho "Morphosys: An Integrated Re-Configurable Architecture," *Proceedings of the NATO Symposium on System Concepts and Integration*, Monterey, CA, April 1998.
- c58 R. Maestre, F. J. Kurdahi, N. Bagherzadeh, H. Singh, R. Hermida, and M. Fernandez, "Kernel Scheduling in Reconfigurable Computing," *Proceedings of DATE*, pp. 90-96, Munich, March 1999.
- c59 G. Lu, M. Lee, M. Singh, N. Bagherzadeh, F. Kurdahi and E. Filho, "MorphoSys: A Reconfigurable Processor Targeted for High-Performance Image Applications," *Proceedings of the Reconfigurable Architecture Workshop at the 11th IPP.S/SPDP'99*, pp. 661-9, Puerto Rico, April 1999.

Consultant Curriculum Vitae

- c60 G. Lu, H. Singh, M. Lee, N. Bagherzadeh, F. Kurdahi, E. Filho, and V. Castro Alves, "MorphoSys Dynamically Reconfigurable System-on-chip," *Proceedings of the First NASA/DOD Workshop on Evolvable Hardware, EH 99*, pp. 152-160, Pasadena, California, July 1999.
- c61 G. Lu, H. Singh, M. Lee, N. Bagherzadeh, F. Kurdahi and E. Filho, "The Morphosys Parallel Reconfigurable System," *Proceedings of the EURO-PAR 99*, pp. 727-734, 1999, Toulouse, France.
- c62 R. Maestre, M. Fernandez, R. Hermida, and N. Bagherzadeh, "A Framework for Scheduling and Context Allocation in Reconfigurable Computing," *Proceedings of the International Symposium on System Synthesis*, pp. 134-140, San Jose, California, November 1999.
- c63 R. Maestre, F. J. Kurdahi, M. Fernandez, R. Hermida, N. Bagherzadeh, and H. Singh, "Optimal vs. Heuristic Approaches to Context Scheduling for Multi-Context Reconfigurable Architectures," *Proceedings of the International Symposium on Field-Programmable Custom Computing Machines*, Napa, California, April 2000.
- c64 R. Maestre, F. J. Kurdahi, M. Fernandez, N. Bagherzadeh, and H. Singh, "Configuration Management in Multi-Context Reconfigurable Systems for Simultaneous Performance and Power Optimizations," *Proceedings of the International Symposium on System Synthesis*, Madrid, Spain, Nov. 2000.
- c65 H. Singh, M. Lee, G. Lu, F. Kurdahi, N. Bagherzadeh, E. Filho, and Rafael Maestre, "MorphoSys: A Case Study of A Reconfigurable System for Multimedia Applications," *Proceedings of Design Automation Conference (DAC 2000)*, Los Angeles, California, June 2000.
- c66 R. Maestre, M. Fernandez, R. Hermida, F. Kurdahi, N. Bagherzadeh, and H. Singh, "Analysis and Evaluation of Context Scheduling Heuristics for Multi-Context Reconfigurable Architectures," *Proceedings of ICCD 2000*.
- c67 H. Singh, M. Lee, G. Lu, F. Kurdahi, N. Bagherzadeh, and E. Filho, "MorphoSys: A Reconfigurable Architecture for Multimedia Applications," *Workshop on Reconfigurable Computing at the International Conference on Parallel Architectures and Compilations, PACT '98*.
- c68 J. Liu, P. Chou, N. Bagherzadeh, and F. Kurdahi, "Power-Aware Scheduling under Timing Constraints for Mission-Critical Embedded Systems," in *Proc. 38th Design Automation Conference*, Las Vegas, NV, June 2001. pp. 840-845.
- c69 J. Liu, P. Chou, N. Bagherzadeh, and F. Kurdahi, "A Constraint-based Application Model and Scheduling Techniques for Power-aware Systems", in, *Proc. International Symposium on Hardware/Software Codesign*, Copenhagen, Denmark, April 2001. pp. 153-158.

Consultant Curriculum Vitae

- c70 M. Ahamed, H. Eldeeb, S. Nassar, and N. Bagherzadeh, "Design and Implementation of Automatic Parallel Detection Layer," *IEEE Conference on Systems, Man, and Cybernetics*, October 2001, pp. 2371-2376.
- c71 G. Venkataramani, W. Najjar, F. Kurdahi, N. Bagherzadeh, W. Böhm, "A compiler framework for Mapping Applications to a coarse-grained reconfigurable computer architecture," *CASES 2001*, pp. 116-125.
- c72 M. Sanchez-Elez, M. Fernandez, R. Hermida, R. Maestre, F. Kurdahi, and N. Bagherzadeh, "A Data Scheduler for Multi-Context Reconfigurable Architectures," *14th International Symposium on System Synthesis*, October 2001, pp. 177-182.
- c73 D. Li, P. Chou, and N. Bagherzadeh, "Mode Selection and Mode-Dependency Modeling for Power-Aware Embedded Systems," in *Proc. 7th Asia South Pacific Design Automation Conference (ASPDAC'02)*, January 2002. pp. 697-704. Best Student Paper Award.
- c74 J. Liu, P. Chou and N. Bagherzadeh, "Communication Speed Selection for Embedded Systems with Networked Voltage-Scalable Processors," *Proceedings of the 10th International Symposium on Hardware/Software Codesign (CODES'02)*, Estes Park, CO, May 2002. pp. 169-174.
- c75 J. Liu, P. Chou and N. Bagherzadeh, "Combined Functional Partitioning and Communication Speed Selection for Networked Voltage-Scalable Processors," in *Proceedings of the International Symposium on System Synthesis*, October 2002. pp. 14-19.
- c76 M. Sainz, N. Bagherzadeh and A. Susin, "Carving 3D Models from Uncalibrated Views," *5th IASTED International Conference Computer Graphics and Imaging (CGIM 2002)*, August 12-14, 2002 Kauai, Hawaii, USA. pp. 144-149.
- c77 S. Latifi S, S.Q. Zheng, and N. Bagherzadeh, "Hamiltonian Path and Cycle in Hypercubes with Faulty Links," *Proceedings Fifth International Conference on Algorithms and Architectures for Parallel Processing. IEEE Comput. Soc.* 2002, pp. 471-8. Los Alamitos, CA, USA.
- c78 M. Sainz, N. Bagherzadeh and A. Susin, "Hardware Accelerated Voxel Carving," *1st Ibero-American Symposium in Computer Graphics (SIACG 2002)*, July 2-5, 2002 Guimarães, Portugal. pp. 289-297.
- c79 H. Parizi, A. Niktash, N. Bagherzadeh, and F. Kurdahi, "MorphoSys: A Coarse Grain Reconfigurable Architecture for Multimedia Applications," *Euro-Par 2002 Parallel Processing. 8th International Euro-Par Conference. Proceedings (Lecture Notes in Computer Science Vol.2400)*. Springer-Verlag. 2002, pp. 844-8, Berlin, Germany.

Consultant Curriculum Vitae

- c80 A. Paar, M. Anido, and N. Bagherzadeh, "A Novel Predication Scheme for a SIMD System-on-chip," *Euro-Par 2002 Parallel Processing. 8th International Euro-Par Conference. Proceedings (Lecture Notes in Computer Science Vol.2400)*. Springer-Verlag. 2002, pp. 834-43. Berlin, Germany.
- c81 M. Anido, N. Tabrizi, H. Du, M. Sanchez-Elez, and N. Bagherzadeh, "Interactive Ray Tracing Using a SIMD Reconfigurable Architecture," *Proceedings 14th Symposium on Computer Architecture and High Performance Computing. IEEE Comput. Soc.* 2002, pp.20-28. Brazil.
- c82 M. Anido, A. Paar, and N. Bagherzadeh, "A Novel Method for Improving the Operation Autonomy of SIMD Processing Elements," *Proceedings 15th Symposium on Integrated Circuits and Systems Design. IEEE Comput. Soc.* 2002, pp.49-54, Los Alamitos, CA, USA.
- c83 M. Sainz, N. Bagherzadeh and A. Susin, "Recovering 3D Metric Structure and Motion from Multiple Uncalibrated Cameras," *Proceedings of International Conference on Information Technology: Coding and Computing*, pp. 268-273, April 2002.
- c84 M. Sanchez-Elez, M. Fernandez, R. Maestre, R. Hermida, N. Bagherzadeh, and F. Kurdahi, "A Complete Data Scheduler for Multi-Context Reconfigurable Architectures," *Design, Automation and Test in Europe Conference and Exhibition, (DATE)* 2002, 4-8 March 2002, pp. 547-552.
- c85 M. Anido, A. Paar, and N. Bagherzadeh, "Improving the Operation Autonomy of SIMD Processing Elements by Using Guarded Instructions and Pseudo Branches," *Digital System Design, 2002. Proceedings. Euromicro Symposium on*, 4-6 Sept. 2002, pp. 148-155.
- c86 C. Pan, N. Bagherzadeh, A. H. Kamalizad, and A. Koohi, "Design and Analysis of a Programmable Single-chip Architecture for DVB-T Base-band Receiver," *Design, Automation and Test in Europe Conference and Exhibition, (DATE), 2003*, March 3-7, 2003, pp. 468 -473.
- c87 H. Du, M. Sanchez-Elez, N. Tabrizi, N. Bagherzadeh, M. Anido, and M. Fernandez, "Interactive Ray Tracing on Reconfigurable SIMD MorphoSys," *Design Automation and Test in Europe Conference and Exhibition, 2003*, March 3-7, 2003, pp. 144-149.
- c88 M. Sanchez-Elez, M. Fernandez, M. Anido, H. Du, N. Bagherzadeh, R. Hermida, "Low Energy Data Management for Different On-chip Memory Levels in Multi-Context Reconfigurable Architectures," *Design, Automation and Test in Europe Conference and Exhibition, (DATE) 2003*, March 3-7, 2003, pp. 36-41.
- c89 H. Du, M. Sanchez-Elez, N. Tabrizi, N. Bagherzadeh, M. Anido, and M. Fernandez, "Interactive Ray Tracing on Reconfigurable SIMD MorphoSys," *Proceedings of the*

Consultant Curriculum Vitae

- ASP-DAC 2003. Asia and South Pacific Design Automation Conference 2003* (Cat. No.03EX627), IEEE. 2003, pp. 471-6. Piscataway, NJ, USA.
- c90 M. Sainz, A. Susin, A. Cervantes, and N. Bagherzadeh, "Persepolis: Recovering History with a Handheld Camera," *Eurographics 2003* (EG'03).
- c91 M. Sainz, A. Susin, and N. Bagherzadeh, "MTMesh: Image Based Mesh Reconstruction and Rendering," *In Proceedings of the IAESTED Conference in Visualization, Imaging and Image Processing (VIIP'03)*, 2003.
- c92 Sainz, A. Susin, and N. Bagherzadeh, "Camera Calibration of Long Image Sequences with the Presence of Occlusions," *Proceeding of the International Conference in Image Processing 2003 (ICIP'03)*, Barcelona.
- c93 Li, P. Chou and N. Bagherzadeh, "Topology Selection for Energy Minimization in Embedded Networks," in *Proc. Asia South-Pacific Design Automation Conference (ASPDAC'03)*, Kitakyushu, Japan, January 2003. pp. 693-696.
- c94 Villa-Uriol, M. Sainz, F. Kuester, and N. Bagherzadeh, "Automatic creation of three-dimensional avatars," *Proceedings of SPIE-IS&T Electronic Imaging*, SPIE Vol. 5013 (2003), January 21-22, 2003 Santa Clara, California, USA. pp. 14-25.
- c95 H. Du, M. Sanchez-Elez, N. Tabrizi, and N. Bagherzadeh, M. Fernandez, "MorphoSys: a Reconfigurable Architecture Optimized for Ray Tracing," *Cool-Chips VI*, 2003, April 16-18, Yokohama, Japan.
- c96 A. Paar, H. Du, and N. Bagherzadeh, "A Component Oriented Simulator for HW/SW Co-Designs," *First Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia) in conjunction with [IEEE/ACM CODES-ISSS 2003 Conference](#)*, Newport Beach, CA.
- c97 A. Koochi, N. Bagherzadeh, and Chengzhi Pan, "A Fast Parallel-Solomon Decoder on a Reconfigurable Architecture," *IEEE/ACM CODES-ISSS 2003 Conference*, Newport Beach, CA.
- c98 M. Villa-Uriol, A. Perez, F. Kuester, and N. Bagherzadeh, "Image-Based Avatar Reconstruction," *Proceedings of the NSF Lake Tahoe Workshop on Collaborative Virtual Reality and Visualization*, (2003).
- c99 A. H. Kamalizad, C. Pan, and N. Bagherzadeh, "Fast Parallel FFT on a Reconfigurable Computation Platform," *Proceedings of the 15th Symposium on Computer Architecture and High Performance Computing*, 10-12 November 2003, pp. 254-259.
- c100 M. Villa-Uriol, G. Chaudhary, F. Kuester, T.C. Hutchinson, and N. Bagherzadeh, "Extracting 3D from 2D: Selection Basis for Camera Calibration," *Proceedings of the*

Consultant Curriculum Vitae

7th IASTED International Conference on Computer Graphics and Imaging (CGIM 2004), pp. 315-321.

- c101 N. Tabrizi, N. Bagherzadeh, A. Kamalizad, and H. Du, "MaRS: A Macro-Pipelined Reconfigurable System," *Computing Frontiers (CF'04)*, April 14-16, Ischia, Italy, pp. 343-349.
- c102 B. Gorji-Ara, P. Chou, N. Bagherzadeh, M. Reshadi, and D. Jensen, "Fast and Efficient Voltage Scheduling by Evolutionary Slack Distribution," *Proceedings of the ASP-DAC 2004*, Jan. 2004, pp. 659-662.
- c103 B. Gorji-Ara, N. Bagherzadeh, and P. Chou, "An Efficient Voltage Scaling Algorithm for Complex SoCs with few Number of Voltage Modes," *Proceedings of the 2004 International Symposium on Low Power Electronics and Design (ISLPED 2004)*, August 9-11, 2004, pp. 381-386.
- c104 F. Rivera, M. Sanchez-Elez, M. Fernandez, R. Hermida, and N. Bagherzadeh, "Efficient Mapping of Hierarchical Trees on Coarse-Grain Reconfigurable Architectures," *Proceedings of CODES+ISSS 2004*, 8-10 September 2004, pp. 30-35.
- c105 A. Kamalizad, R. Plettner, C. Pan, and N. Bagherzadeh, "Fast Parallel Soft Viterbi Decoder Mapping on a Reconfigurable DSP Platform," *IEEE International SOC Conference*, 12-15 September, 2004, pp. 3-6.
- c106 J. Hahn, D. Li, Q. Xie, P. Chou, N. Bagherzadeh, D. Jensen, and A. Tribble, "Power Reduction in JTRS Radios with ImpacPro," *IEEE Military Communications Conference (MILCOM 2004)*, 31 Oct. - 03 Nov. 2004, pp. 204-210.
- c107 F. Rivera, M. Sanchez-Elez, M. Fernandez, R. Hermida, and N. Bagherzadeh, "Low Power Data Prefetch for 3D Image Applications on Coarse-Grain Reconfigurable Architectures," *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (2005)*, 04-08 April 2005, pp. 171b-171b.
- c108 A. Kamalizad, N. Tabrizi, N. Bagherzadeh, and A. Hatanaka, "A Programmable DSP Architecture for Wireless Communication Systems," *16th IEEE International Conference on Application-Specific Systems, Architecture Processors (ASAP 2005)*, 23-25 July 2005, pp. 231-238.
- c109 F. Rivera, M. Sanchez-Elez, M. Fernandez, and N. Bagherzadeh, "An Approach to execute Conditional Branches onto SIMD Multi-Context Reconfigurable Architectures," *Proceedings of the 8th Euromicro Conference on Digital System Design*, 30 Aug - 03 September, 2005, pp. 396-402.
- c110 N. Tabrizi and N. Bagherzadeh, "An ASIC Design of a Novel Pipelined and Parallel Sorting Accelerator for a Multiprocessor-on-a-chip," *Proceedings of the 6th International Conference on ASIC (ASICON 2005)*, 24-27 Oct-20 Oct 2005, pp. 46-49.

Consultant Curriculum Vitae

- c111 A. Niktash, R. Masetre, and N. Bagherzadeh, "A Case Study of Performing OFDM Kernels on a Novel Reconfigurable DSP Architecture," *IEEE Military Communications Conference (MILCOM 2005)*, 17-20 October 2005, pp. 1813-1818.
- c112 H. Parizi, A. Niktash, A. Kamalizad, and N. Bagherzadeh, "A Reconfigurable Architecture for Wireless Communication Systems," *Third International Conference on Information Technology: New Generations, (ITNG 2006)*, 10-12 April 2006, pp. 250-255.
- c113 J. Davila, A. de Torres, J. M. Sanchez, M. Sanchez-Elez, N. Bagherzadeh, and F. Rivera, "Design and Implementation of a Rendering Algorithm in a SIMD Reconfigurable Architecture (MorphoSys)," *Proceedings of Design, Automation, and Test in Europe (DATE 2006)*, 6-10 March 2006, pp. 6.
- c114 S. Lee, J. Bahn, and N. Bagherzadeh, "Design of a Feasible On-Chip Interconnection Network for a Chip Multiprocessor (CMP)," *19th SBAC*, October 2007.
- c115 S. Lee and N. Bagherzadeh, "Increasing the Throughput of an Adaptive Router in Network-on-Chip (NoC)," *Proceedings of the 4th International Conference on Hardware/software codesign and system synthesis, 2006. CODES+ISSS '06*, 22-25 Oct. 2006 Page(s):82 - 87
- c116 F. Rivera, M. Sanchez-Elez, M. Fernandez, R. Hermida, and N. Bagherzadeh, "Configuration Scheduling for Conditional Branch Execution onto Multi-Context Reconfigurable Architectures," *International Conference on Field Programmable Logic and Applications, 2006. FPL '06*, 28-30 Aug. 2006, Page(s):1 – 8.
- c117 A. Niktash, H. Parizi, and N. Bagherzadeh, "A Multi-Standard Viterbi Decoder for Mobile Applications Using a Reconfigurable Architecture," *Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th*, Page(s):1 – 5.
- c118 N. Tabrizi and N. Bagherzadeh, "A Distributed and Shared Register File for a Multiprocessor-on-Chip to Support Real-Time Applications," *The 9th International Conference on Advanced Communication Technology*, Dec. 2006 Page(s):215 – 220.
- c119 N. Tabrizi and N. Bagherzadeh, "A Parallel Sort Engine with Dynamic Memory for a Multiprocessor-on-a-Chip," *Proceedings of the Fourth IASTED International Conference, Circuits, Signals, and Systems*, November, 2006 Page(s):169 – 174.
- c120 A. Niktash, H. Parizi, and N. Bagherzadeh, "A Study of Implementation of IEEE 802.11 a Physical Layer on a Heterogeneous Reconfigurable Platform," *The 9th International Conference on Advanced Communication Technology*, Volume 3, 12-14 Feb. 2007 Page(s):2101 – 2105.

Consultant Curriculum Vitae

- c121 B. Gorjiara, N. Bagherzadeh, and P. Chou, "Integrating Power Management into Distributed Real-time Systems at Very Low Implementation Cost," *Asia and South Pacific Design Automation Conference, 2007. ASP-DAC '07.*, 23-26 Jan. 2007 Page(s):872 – 877.
- c122 A. Hatanaka and N. Bagherzadeh, "A Modulo Scheduling Algorithm for a Coarse-Grain Reconfigurable Array Template," *IEEE International Parallel and Distributed Processing Symposium, 2007. IPDPS 2007*, 26-30 March 2007 Page(s):1 – 8.
- c123 A. Niktash, H. Parizi, and N. Bagherzadeh, "Application of a Heterogeneous Reconfigurable Architecture to OFDM Wireless Systems," *IEEE International Symposium on Circuits and Systems, 2007. ISCAS 2007*, 27-30 May 2007 Page(s):2586 – 2589.
- c124 S. Lee, J. Bahn, and N. Bagherzadeh, "Design of a Feasible On-Chip Interconnection Network for a Chip Multiprocessor (CMP)," *19th International Symposium on Computer Architecture and High Performance Computing, 2007. SBAC-PAD 2007*, 24-27 Oct. 2007 Page(s):211 – 218.
- c125 J. Bahn, S. Lee, and N. Bagherzadeh, "On Design and Analysis of a Feasible Network-on-Chip (NoC) Architecture," *International Conference on Information Technology, ITNG 2007*.
- c126 H. Eldeeb, H. Elsadek, H. Abdallah, and N. Bagherzadeh, "FDTD Accelerator for Calculating SAR Distributed Inside Human Head due to Radiation from Wireless Devices," *EMTS 2007 International URSI Commission B - Electromagnetic Theory Symposium.*, July 26-28, 2007 Ottawa, ON, Canada.
- c127 A. Niktash, H. Parizi, A. H. Kamalizad, and N. Bagherzadeh, "PHY 23-3 - RECFEC: A Reconfigurable FEC Processor for Viterbi, Turbo, Reed-Solomon and LDPC Coding," *IEEE Wireless Communications and Networking Conference, 2008. WCNC 2008*, March 31 2008-April 3 2008 Page(s):605 – 610.
- c128 J. Bahn, J. Yang, and N. Bagherzadeh, "Parallel FFT Algorithms on Network-on-Chips," *Fifth International Conference on Information Technology: New Generations, 2008. ITNG 2008*, 7-9 April 2008 Page(s):1087 – 1093.
- c129 N. Bagherzadeh and M. Matsuura, "Performance Impact of Task-to-Task Communication Protocol in Network-on-Chip," *Fifth International Conference on Information Technology: New Generations, 2008. ITNG 2008*, 7-9 April 2008 Page(s):1101-1106.
- c130 J. Bahn and N. Bagherzadeh, "Efficient Parallel Buffer Structure and Its Management Scheme for a Robust Network-on-Chip (NoC) Architecture," *13th International CSI Computer Conference (CSICC 2008)*, Kish Island, Iran, 2008.

Consultant Curriculum Vitae

- c131 W. Trumler, S. Schlingmann, T. Ungerer, J. Bahn, and N. Bagherzadeh, "Self-optimized Routing in a Network-on-a-Chip," *20th IFIP World Computer Congress*, Milano, Italy 7-10 September 2008.
- c132 H. Elsadek, H. Eldeeb, H. Abdallah, M. Desouky and N. Bagherzadeh, "Specific Absorption Rate Calculation using Parallel Finite Difference Time Domain," *WORLDCOMP'08*, Las Vegas, July 2008.
- c133 F. Bagci, T. Ungerer, and N. Bagherzadeh, "ESTR – Energy Saving Token Ring Protocol for Wireless Sensor Networks," *WORLDCOMP'08 Conference, Proceedings of the International Conference on Wireless Networks (ICWN 08)*, Las Vegas, July 2008.
- c134 F. Bagci, F. Kluge, T. Ungerer, and N. Bagherzadeh, "LocSens- An Indoor Location Tracking System using Wireless Sensors," *Proceedings of the First IEEE International Workshop on Sensor Networks (SN 2008)*, St. Thomas, VI, USA, August 2008.
- c135 W. Hu, J. Bahn, and N. Bagherzadeh, "Parallel LDPC Decoding on a Network-on-Chip Based Multiprocessor Platform," *Computer Architecture and High Performance Computing, 2009, SBAC-PAD '09*, Sao Paulo, October 2009, Pages 35-40.
- c136 A. Hatanaka and N. Bagherzadeh, "Scheduling Techniques for Multi-Core Architectures," *Proceedings of 6th International Conference on Information Technology: New Generations, 2009, ITNG '09*, Las Vegas 2009, Pages 865-870.
- c137 W. Hu, C Chen, and N. Bagherzadeh, "A Spectral-Based Partitioning Algorithm for Parallel LDPC Decoding on a Multiprocessor Platform," *SoC Design Conference*, New Port Beach, CA 2009, Pages 245-248.
- c138 Y. Yang, J. Bahn, S. Lee and N. Bagherzadeh, "Parallel and Pipeline Processing for Block Cipher Algorithms on a Network-on-Chip," *Proceedings of 6th International Conference on Information Technology: New Generations, 2009, ITNG '09*, Las Vegas 2009, Pages 849-854.
- c139 S. Lee, C. Wilkerson, M. Zhang, R. Yavatkar, S. Lu, and N. Bagherzadeh, "Low Power Adaptive Pipeline Based on Instruction Isolation," *10th Int'l Symposium on Quality Electronic Design ISQED 2009*, Pages 788-793.
- c140 F. Bagci, J. Wolf, T. Ungerer, and N. Bagherzadeh, "Mobile Agents for Wireless Sensor Networks," *ICWN'09 - The 2009 International Conference on Wireless Networks*, Las Vegas, USA, July 13-16, 2009
- c141 F. Bagci, T. Ungerer, and N. Bagherzadeh, "SecSens - Security Architecture for Wireless Sensor Networks," *Third International Conference on Sensor Technologies and Applications, SENSORCOMM 2009*, Pages 449-454.

Consultant Curriculum Vitae

- c142 J. Huseynov, S. Baliga, M. Dillencourt, L. Bic, and N. Bagherzadeh, "Gas-Leak Localization Using Distributed Ultrasonic Sensors," *Smart Sensor Phenomena, Technology, Networks, and System, Proc. of SPIE Vol. 7293 72930Z-18*, 2009.
- c143 H. Abdullah, J. Yang, N. Bagherzadeh, and K. Hussein, "Parallel Electric Field Integral Equation Solver for Arbitrary Shaped Conducting Bodies," *Progress In Electromagnetics Research Symposium*, Beijing, China, March 23-27, 2009, Pages 1661-1666
- c144 C. Wang, W Hu, N. Bagherzadeh, and S Lee, "Area and Power-Efficient Innovative Network-on-Chip Architecture," *18th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)2010*, Pages 533-539.
- c145 F. Penczek, S. Herhut, S. Scholz, A Shafarenko, J, Yang, C. Chen, N. Bagherzadeh, and C. Grelck, "Message Driven Programming with S-Net Methodology and Performance," *39th International Conference on Parallel Processing Workshops (ICPPW), 2010*, Pages 405-412.
- c146 A. Alhussien, C. Wang, and N. Bagherzadeh, "A Scalable Delay Insensitive Asynchronous NoC with Adaptive Routing," *IEEE 17th International Conference on Telecommunications (ICT) 2010*, Pages 995-1002.
- c147 C. Wang, W Hu, and N. Bagherzadeh, "Congestion-Aware Network-on-Chip Router Architecture," *15th CSI International Symposium on Computer Architecture and Digital Systems (CADS) 2010*, Pages 137-144.
- c148 H. Eldeeb, H. Elsadek, M. Dessokey, H. Abdullah, and N. Bagherzadeh, "High Performance Parallel Computing for FDTD Numerical Technique in Electromagnetic Calculations for SAR Distribution Inside Human Head," *14th WSEAS International Conference on Computer*, July 23-25 2010, Corfu, Greece.
- c149 H. Abdullah, H. Elsadek, H. Eldeeb, and N. Bagherzadeh, "High Performance Grid Computation of the Scattered Field Formulation for Nth Order Debye Modeling of the General Dispersive Media," *International Conference on Bioinformatics and Computational Biology, BIOCOMP'11*, July 18-21, 2011, Monte Carlo Resort, Las Vegas, Nevada, USA.
- c150 C. Wang, W Hu, and N. Bagherzadeh, "A Wireless Network-on-Chip Design for Multicore Platforms," *2011 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing*, Pages 409-416
- c151 J. Yang, C. Chun, N. Bagherzadeh, and S. Lee "Load Balancing for Data-Parallel Applications on Network-on-Chip Enabled Multi-Processor Platform," *2011 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing*, Pages 439-446

Consultant Curriculum Vitae

- c152 C. Wang, W Hu, and N. Bagherzadeh, "Design and Evaluation of a High Throughput QoS-Aware and Congestion-Aware Router Architecture for Network-on-Chip," *2012 20th International Euromicro Conference on Parallel, Distributed and Network-Based Processing*, Pages 457-464
- c153 W. Hu, C. Wang, and N. Bagherzadeh, "Design and Analysis of a Mesh-based Wireless Network-on-Chip," *2012 20th International Euromicro Conference on Parallel, Distributed and Network-Based Processing*, Pages 483-490
- c154 S. Azampanah, A. Khademzadeh, N. Bagherzadeh, M. Janidarmian, and R. Shojaei "LATEX: New Selection Policy for Adaptive Routing in Application-Specific NoC," *PDP 2012*, Pages 515-519
- c155 A. Alhussien, N. Bagherzadeh, F. Verbeek, B. van Gastel, and J. Schmaltz "A Formally Verified Deadlock-Free Routing Function in a Fault-Tolerant NoC Architecture," *SBCCI 2012*, Pages 1-6
- c156 A. Eskandari, A. Khademzadeh, N. Bagherzadeh, and M. Janidarmian "Quality of Service Optimization for Network-on-Chip Using Bandwidth-Constraint Mapping Algorithm," *PDP 2013*, Pages 504-508
- c157 A. Demiriz, N. Bagherzadeh, and A. Alhussien "CPNoC: On Using Constraint Programming in Design of Network-on-Chip Architecture," *PDP 2013*, Pages 486-493
- c158 A. Demiriz and N. Bagherzadeh "On Heterogeneous Network-on-Chip Design Based on Constraint Programming," *NoCArc, MICRO 2013*, Pages 29-34
- c159 J. Bahn and N. Bagherzadeh "A Generic Traffic Model for On-Chip Interconnection Networks," *NoCArc*, 2008, Pages 22-29
- c160 W. Hu and N. Bagherzadeh "DMesh: a Diagonally-Linked Mesh Network-on-Chip Architecture," *NoCArc*, 2008, Pages 14-20
- c161 P. Yaghini, A. Eghbal, and N. Bagherzadeh "A GALS Router for Asynchronous Network-on-Chip," *MES '14 Proceedings of International Workshop on Manycore Embedded Systems*, 2014
- c162 A. Eghbal, P. Yaghini, S. Yazdi, and N. Bagherzadeh, "TSV-to-TSV Inductive Coupling Aware Coding Scheme for 3D Network-on-Chip," *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2014 IEEE International Symposium on*, pp. 92–97, Oct 2014.
- c163 F. Hosseinzadeh, N. Bagherzadeh, A. Khademzadeh, M. Janidarmian, and F. Karimi, "Improving Reliability in Application-Specific 3D Network-on-Chip," *Proceedings of the World Congress on Engineering and Computer Science 2012 Vol I WCECS*, October 24-26, 2012, San Francisco, USA.

Consultant Curriculum Vitae

- c164 R. Salamat, M. Ebrahimi, and N. Bagherzadeh, "An Adaptive, Low Restrictive and Fault Resilient Routing Algorithm for 3D Network-on-Chip," *2015 23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, DOI 10.1109/PDP.2015.91, pp. 392-395, 2015.
- c165 A. Eghbal, P. Yaghini, and N. Bagherzadeh, "Capacitive Coupling Mitigation for TSV-Based 3D ICs," *2015 IEEE 33rd VLSI Test Symposium (VTS)*, (2015)
- c166 M. S. Rusli, O. C. Yee, M. N. Marsono, P. Yaghini, N. Bagherzadeh, D. Patti, V. Dieei, and M. Palesi, "An Energy Aware Transmission Control in Wireless Network-on-Chip," *Recent Advances in Computer Science*, (ISBN: 978-1-61804-297-2), pp. 24-33, 2015,
- c167 H. Elsadek, H. Eldeeb, H. Abdallah, and N. Bagherzadeh, "SAR Distribution over whole Human Body Existed in the Field of Mobile Base Station Antenna," *International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, 2015 *IEEE MTT-S*, <https://doi.org/10.1109/NEMO.2015.7415033>, Ottawa, Canada
- c168 P. Yaghini, A. Eghbal, S. Yazdi, and N. Bagherzadeh "Accurate System-level TSV-to-TSV Capacitive Coupling Fault Model for 3D-NoC," *NOCS 2015*, Vancouver, BC, Canada, DOI: <http://dx.doi.org/10.1145/2786572.2786598>
- c169 Q. Zhao, M. Rajaei, I. Krivorotov, M. Nilsson, N. Bagherzadeh, and O. Boyraz "Optical Investigation of Radiation Induced Conductivity Changes in STT-RAM Cells," *Conference on Lasers and Electro-Optics, OSA Technical Digest (online) (Optical Society of America 2016)*, DOI https://doi.org/10.1364/CLEO_QELS.2016.FM3B.2
- c170 F. Verbeek, P. Yaghini, A. Eghbal, and N. Bagherzadeh "ADVOCAT: Automated Deadlock Verification for On-chip Cache coherence and Interconnects," *DATE 2016*, Dresden, Germany
- c171 R. Salamat, M. Ebrahimi, N. Bagherzadeh, and F. Verbeek, "CoBRA: Low Cost Compensation of TSV Failures in 3D-NoC," *2016 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, DOI [10.1109/DFT.2016.7684081](https://doi.org/10.1109/DFT.2016.7684081),
- c172 Z. Ghaderi, A. Alqahtani, and N. Bagherzadeh, "Online Monitoring and Adaptive Routing for Aging Mitigation in NoCs," *2017 DATE*, DOI <https://doi.org/10.23919/DATE.2017.7926960>, 27-31 March 2017, Lausanne, Switzerland

Consultant Curriculum Vitae

- c173 M. Nilsson, I. Krivorotov, O. Boyraz, N. Bagherzadeh, R. Ngelale, B. Adolf, J. Chen, E. Montoya, H. Lee, C. Sha, S. Mohammadi, P. Sadri, H. Tseng, L. Wan, and E. Yang, “Effect of Ionizing Radiation on STT-RAM Multilayers,” *GOMAC 2017*.
- c174 P. Sadri, M. Khan, Q. Zhao, I. Krivorotov, M. Nilsson, N. Bagherzadeh, and O. Boyraz, “Plasmonic Detection of Possible Defects in Multilayer Nanohole Array Consisting of Essential Materials in Simplified STT-RAM Cell,” 2017 *SPIE Optics and Photonics Conference*, San Diego, CA.
- c175 A. Albaqsami, M. Hosseini, and N. Bagherzadeh, “HTF-MPR: A Heterogeneous TensorFlow Mapper Targeting Performance using Genetic Algorithms and Gradient Boosting Regressors,” *DATE* (2018), 978-3-9819263-0-9/DATE18/c 2018 EDAA
- c176 M. Torabzadehkashi, S. Rezaei, V. Alves, and N. Bagherzadeh, “CompStor: An In-Storage Computation Platform for Scalable Distributed Processing,” 2018 *IEEE International Parallel and Distributed Processing Symposium Workshops*, DOI 10.1109/IPDPSW.2018.00195
- c177 L. Morales, J. Aedo, and N. Bagherzadeh, “Simulation-Based Evaluation Strategy for Task Mapping Approaches in WNoC Platforms,” 26th *Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (2018)*, DOI 10.1109/PDP2018.2018.00104
- c178 M. Kim, A. Del Barrio, R. Hermida, and N. Bagherzadeh, “Low-Power Implementation of Mitchell’s Approximate Logarithmic Multiplication for Convolutional Neural Networks,” [2018 23rd Asia and South Pacific Design Automation Conference \(ASP-DAC\)](#), DOI: [10.1109/ASPDAC.2018.8297391](#)
- c179 H. Kim, M. Kim, A. Del Barrio, and N. Bagherzadeh, “A Cost-Efficient Iterative Truncated Logarithmic Multiplication for Convolutional Neural Networks,” 2019 *IEEE 26th Symposium on Computer Arithmetic (ARITH)*, DOI 10.1109/ARITH.2019.00029
- c180 M. Torabzadehkashani, S. Rezaei, A. Heydarigorji, H. Bobarshad, A. Alves, and N. Bagherzadeh, “Catalina: In-Storage Processing Acceleration for Scalable Big Data Analytics,” 2019 27th *Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, DOI 10.1109/PDP.2019.00069
- c181 S. Mohammadi, M. Jasemi, S. Talebi, N. Bagherzadeh, and M. Green, “A Radiation Hard Sense Circuit for Spin Transfer Torque Random Access Memory,” [2019 IEEE International Symposium on Circuits and Systems \(ISCAS\)](#), DOI: [10.1109/ISCAS.2019.8702416](#)
- c182 S. Shahhosseini, A. Albaqsami, M. Jasemi, and N. Bagherzadeh, “Partition Pruning: Parallelization-Aware Pruning for Dense Neural Networks,” 28th *Euromicro*

Consultant Curriculum Vitae

International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2020, DOI 10.1109/PDP50117.2020.00052

- c183 M. Alnemari and N. Bagherzadeh, "Efficient Deep Neural Networks for Edge Computing," *2019 IEEE International Conference on Edge Computing (EDGE)*, **Best Paper**, DOI 10.1109/EDGE.2019.00014
- c184 P. Sadri, M. Wahiduzzaman, M. Shafiqul Islam, I. Krivorotov, M. Nilsson, N. Bagherzadeh, and O. Boyraz, "Array of Symmetric Nanohole Dimers for STT-RAM Ultrathin Layer Sensing," *CLEO 2019*
- c185 Z. Ren, A. Alqahtani, N. Bagherzadeh, and J. Lee, "Thermal Analysis of 3D ICs with TSVs Placement Optimization," *Proceedings of the ASME 2019 International Technical Conference and Exhibition on Packaging and Integration of Electronic and Photonic Microsystems*, October 7-9, 2019, Anaheim, CA
- c186 S. Sadri, M. Wahiduzzaman, M. Bayer, M. Islam, E. Montoya, I. Krivorotov, M. Nilsson, N. Bagherzadeh, and O. Boyraz, "Effect of Tantalum and MgO Adhesion Layers on Plasmonic Nanostructures," *SPIE Nanoscience + Engineering*, 2019, San Diego, California.
- c187 M. Torabzadehkashi, A. Heydarigorji, S. Rezaei, H. Bobarshad, V. Alves, and N. Bagherzadeh, "Accelerating HPC Applications Using Computational Storage Devices," *HPCC*, 2019
- c188 L. Tavares, M. S. Kim, A. Del Barrio, N. Bagherzadeh, and R. Menotti, "Design of Power-Efficient FPGA Convolutional Cores with Approximate Log Multiplier," *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, April 2019
- c189 L. Alhubail and N. Bagherzadeh, "Power and Performance Optimal NoC Design for CPU-GPU Architecture Using Formal Models," *DATE-19*, 2019
- c190 L. Alhubail, M. Jasemi, and N. Bagherzadeh, "NoC Design Methodologies for Heterogeneous Architecture," *2020 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, DOI 10.1109/PDP50117.2020.00052
- c191 M. Moghadam, E. Masoumi, N. Bagherzadeh, D. Ramsingh, and Z. Kain, "Supervised Machine-Learning Algorithms in Real-time Prediction of Hypotensive Events," [2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology](#),
- c192 M. Hosseini, M. Ebrahimi, P. Yaghini, and N. Bagherzadeh, "Application Characterization for Near Memory Processing," *2021 29th Euromicro International*

Consultant Curriculum Vitae

Conference on Parallel, Distributed and Network-Based Processing (PDP), DOI 10.1109/PDP52278.2021.00031

- c193 P. Sadri-Moshkenani, M. Wahiduzzaman Khan, Md Shafiqul Islam, D. Shi, E. Montoya, I. Krivorotov, N. Bagherzadeh, and O. Boyraz, "Optoelectronic Readout of STT-RAM Memory Cells Using Plasmon Drag Effect," *CLEO 2021*
- c194 Y. Qiao, M. Alnemari, and N. Bagherzadeh, "A Two-Stage Efficient 3-D CNN Framework for EEG Based Emotion Recognition," *23rd IEEE International Conference on Industrial Technology (ICIT 2022)*, August 2022.
- c195 H. Kaeley, Y. Qiao, and N. Bagherzadeh, "Support for Stock Trend Prediction Using Transformers and Sentiment Analysis," *IISES 18th Economics and Finance Conference*, London 2023. <https://doi.org/10.48550/arXiv.2305.14368>

Professional Associations and Achievements

- Fellow Member, Institute of Electrical and Electronics Engineers (IEEE)
- Member, IEEE Computer Society

Extramural Funding (over 8 million dollars)

- (with Dan Gajski, Lubomir Bic, and Kane Kim), 32 node Intel hypercube, *Information and Computer Science and Engineering (CISE) Research Instrumentation - NSF*, \$187,274, 1990. This equipment is shared with other co-principles to study parallelism.
- (with Jack Sklansky), "Multiprocessor Architecture for Image Processing," \$26,700. A MICRO Research Proposal with Rockwell International, 1990-91.
- (with Alex Nicolau), "Design of a RISC Based Multiprocessor for a Fine-Grain Optimizing Compiler," \$33,900. A MICRO Research Proposal with Northrop Corporation, 1990-91.
- (with Dan Gajski), "Design of a RISC Cluster VLSI Chip," \$120,000. A Western Digital Fellowship in VLSI Systems Engineering; support for two students for two years at \$30,000 per fellow per year (this amount includes a stipend of \$18,000, tuition and fees, \$3,000 for their project, and the remainder for the use of the Electrical and Computer Engineering Department)
- (with Kevin Tsai and Jack Sklansky), "A Hypermesh Architecture for Image Multiprocessing," \$20,000. A MICRO Research Proposal with Rockwell International, 1990-1991.

Consultant Curriculum Vitae

- ARMY faculty fellowship, \$10,000, 1990.
- From Sony Corporation for support of advanced computer architecture design, \$30,000, Tokyo, Japan, 1990.
- NASA/ASEE faculty fellowship, \$10,000, 1991.
- From NKK Corporation for support of advanced computer architecture design, \$129,000, Tokyo, Japan, 1991-92.
- "A Quantitative Performance Comparison of a VLIW and Superscalar Architectures," \$30,000, a MICRO Research Proposal with Unitec, 1992-93.
- From Asahi Chemical for support of advanced computer architecture design, \$40,000, 1993-94.
- (with Douglas Blough), Intel MICRO project for iPSC/860 list price (\$265,000) plus cash donation from the state (\$15,000), 1994.
- "Software techniques for improving the efficiency of superscalar microprocessors," (\$25,000 from Unitech Research Inc), (\$11,247 from UC MICRO), 1995.
- (with Fadi Kurdahi), "Digital QAM structures," (\$20,000 from SMC), (\$14,288 from UC MICRO), 1995.
- (with Fadi Kurdahi), "Design and efficient VLSI implementation of a transceiver chip," (\$20,000 from SMC), (\$11,869 from UC MICRO), 1996.
- (with Fadi Kurdahi), "A study of high-speed LAN transceiver chips," (\$20,000 from SMC), (\$11,824 from UC MICRO), 1997.
- (with Fadi Kurdahi), "Dynamically reprogrammable architecture: a new approach to adaptive computing," \$1,182,000 DARPA, 1997-2000.
- (with Fadi Kurdahi), "CISE-NSF-CNPq collaboration research," \$150,000, NSF-CNPq, 1998-2000
- (with Fadi Kurdahi), "Dynamically reprogrammable architecture: a new approach to adaptive computing," \$280,000 DARPA Supplemental Funding, 2000-2001
- (with Pai Chou, Scott Jordan, Fadi Kurdahi, and J.L. Gaudiot-USC), "IMPAC²T: integrated management of power aware computation and communication technologies," \$750, 000, DARPA-ITO, 2000-2002

Consultant Curriculum Vitae

- (with Fadi Kurdahi), “Synthesis of mission-specific processors,” \$500,000, NSF-ITR, 2000-2003
 - (with Fadi Kurdahi), “Using low power reconfigurable computing for high-speed and wireless network appliances,” \$200,000, CoRe (with Broadcom), 2000-2002
 - (with Hutchinson, Kuester, Shinozuka, and Warschauer), “Expanding Engineering Thinking: Interactive Visualization of Numerical Models”, \$410,000, NSF, 2002-2005
 - (with Chou), “Mission Aware Power Management,” \$680, 000, DARPA PACC Phase II, 2002-2004
 - “Design and Implementation of Automatic Parallel Detection Layer,”, \$24,000, NSF-INT (Egypt), 2001-2003
 - (with Faryar, Yang, and Bobrow), “ITR-Dev. of Intelligent Infrastructure For Hazard Mitigation,” \$300,000, NSF, 2002-2005.
 - “Development of FDTD Accelerator for 3D Mutual Interaction Between Human Body and Radiation From Wireless Devices,” \$28,500, NSF, 2005-2008
 - “Autonomic and Organic Real-time SoC Design,” €5,000, Bavaria California Technology Center (BaCaTec), 2004-2006
 - “Task Management on a Reconfigurable Architecture,” €10,200, Uuniversidad Complutense, Madrid, 2006-2007
 - “Multicore Computing,” €4,000, Bavaria California Technology Center (BaCaTec), 2008-2009
 - “US-Egypt Cooperative Research: Health Hazards due to Radiation from Mobile Base Stations,” \$50,000, NSF, Office of International Science & Engineering, 2010-2012
 - (with Nilsson, Miller, Detwiler, and Boyraz), “Nuclear Science and Security Consortium: Nuclear Engineering,” \$2,500,000, NNSA, 2011-2016
 - (with Li and Kurdahi), “Plug Load Energy Reporting and Display,” \$49,335, California Energy Resources Conservation and Development Commission, 2013-2014
 - (with Boyraz, Krivorotov, and Nilsson), “Radiation Characterization of STT-RAM Devices,” \$1,050,000, DTRA, 2016-2018; Additional One Year; \$350,000, 2020-2021
 - (with Jaeho Lee), “3D IC Thermal Management Based on TSV Placement Optimization and Novel Materials,” \$270,000, 2017-2020
-

Consultant Curriculum Vitae