

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

QUALCOMM INCORPORATED and
SAMSUNG ELECTRONICS CO., LTD.
Petitioners

v.

NETWORK SYSTEM TECHNOLOGIES, LLC
Patent Owner

Patent No. 8,072,893

Inter Partes Review No. IPR2024-00355

DECLARATION OF DR. ERIK CHMELAR

TABLE OF CONTENTS

I. INTRODUCTION.....	4
II. EXPERT QUALIFICATIONS AND BACKGROUND	5
III. UNDERSTANDING OF LEGAL PRINCIPLES.....	9
A. Validity Generally	9
B. Prior Art.....	9
C. Anticipation	11
D. Obviousness.....	12
IV. PERSON OF ORDINARY SKILL IN THE ART.....	16
V. OVERVIEW OF THE TECHNICAL BACKGROUND AND THE '2893 PATENT	16
A. Technical Background.....	16
B. Challenges of On-Chip Networks	25
C. Overview of the '2893 Patent.....	33
1. Challenged Claims of the '2893 Patent.....	34
2. The '2893 Patent's File History	36
3. Priority Date	38
D. Claim Constructions	38
The Disputed District Court Constructions	38
1. "each data package comprising N data elements" (claims 1, 7, and 10).....	39
2. "data storage elements" (claims 1, 3, 7, and 10).....	39
Agreed District Court Constructions	39
3. The preamble of claim 10	39
4. "network" (claims 1, 2, 7, 9, and 10)	40
5. "integrated circuit" (claims 1-3, 6, 7, 9, and 10).....	40
VI. ANALYSIS OF THE '2893 PATENT	40
A. Ground 1: Purported Anticipation by Bertozzi (Claims 1-3, 6-7, and 10)	40
1. Independent Claim 10	41
2. Independent Claim 1	50
3. Independent Claim 7	50

4. Dependent Claims	51
B. Ground 2: Purported Obviousness over Bertozzi Alone or in View of Jantsch or Duato (Claims 1–3, 6–7, and 10)	51
C. Ground 3: Purported Obviousness over Dielissen in View of Hassoun (Claims 1–3, 6, 7, 9, and 10)	53
1. Independent Claim 10	53
2. There is No Motivation to Combine Dielissen and Hassoun.....	60
3. Independent Claim 1	62
4. Independent Claim 7	62
5. Dependent Claims	62
D. Ground 4: Purported Obviousness over Grounds 1–3 Further in View of Sullam (Claim 8)	63
VII. CONCLUSION	63

I. INTRODUCTION

1. I have been retained on behalf of Patent Owner Network System Technologies, LLC (“NST”), as an expert consultant.

2. As part of this work, I have been requested by counsel for NST to provide this declaration concerning the technical subject matter relevant to the *inter partes* review (“IPR”) petition of U.S. Patent No. 8,072,893 (the “’2893 patent”) filed by Qualcomm Incorporated (“Qualcomm”) and Samsung Electronics Co., Ltd. (“Samsung”) (collectively, “Petitioners”).

3. I have provided an exemplary, but not exhaustive, summary of materials that I considered or relied upon in preparing this declaration, besides my personal knowledge and experience, in Appendix 2. The paragraphs below may identify additional materials considered or relied upon not listed in Appendix 2. I reserve the right to rely on additional documents as I become aware of them and to respond to documents relied on by Petitioners’ expert.

4. I am being compensated for my time in the amount of \$600 per hour. I have no financial interest in NST or the ’2893 patent, and my compensation is not contingent upon the substance of my declaration, any statements or opinions made, or the outcome of this matter.

II. EXPERT QUALIFICATIONS AND BACKGROUND

5. In 2000, I received a B.S. degree in Electrical Engineering from Michigan Technological University with the highest distinction (summa cum laude). In 2001, I received an M.S. degree in Electrical Engineering from Stanford University, with a focus on computer networks (EE284 Introduction to Computer Networks, EE384A Internet Protocols, EE384S Network Architectures, and EE384S Multimedia Networks). In 2004, I received a Ph.D. degree in Electrical Engineering from Stanford University, with a primary focus on very large-scale integrated (VLSI) systems that culminated in a dissertation entitled “The Test and Diagnosis of Field-programmable Gate Arrays, advised by Dr. Edward J. McCluskey, and with conclusion of a secondary focus on computer networks (EE384X Packet Switch Architectures). While studying at Stanford University from 2000–2004, I taught undergraduate and graduate level courses as a Teaching Assistant (EE284 Introduction to Computer Networks, EE384A Internet Protocols and Standards, Digital Design Laboratory EE121, and E40 Introduction to Electronics), and I was employed as a Research Assistant in the Center for Reliable Computing. In 2011, I received a second B.S. degree in Chemical Physics from Saginaw Valley State University in 2011 with the highest distinction (summa cum laude).

6. During my undergraduate and graduate studies, I was additionally employed at several product-technology companies. From 1999–2002, I was employed as a Manufacturing Engineer at Cisco Systems, Inc, where I oversaw functional testing of layer-2 and layer-3 network switches and determined root-cause failures thereof. During the summers of 2002 and 2003, I was employed as a Product Test Engineer intern at Xilinx, Inc., where I researched and developed software tools to detect and diagnose physical hardware defects in FPGAs.

7. After earning my Ph.D., I was employed at LSI Corp. (now Broadcom, Inc.) from 2004–2012 as a Staff Software Engineer in the Advanced Technology Development department, where I conducted R&D for very large-scale integrated (VLSI) systems and serializer-deserializer (SerDes) circuits for high-speed data communications, for which I was granted approximately 20 U.S. patents. During this period of employment at LSI Corp., I also taught at several Universities. From 2006–2008, I was a consulting Assistant Professor in the Electrical Engineering and Computer Science department at Stanford University, where I advised PhD students in the Center for Reliable Computing on the test and reliability of VLSI systems, and created and taught a PhD-level course on clock and data recovery (CDR) for digital communications (EE391 Clock and Data Recovery Architectures). From 2009–2012, I was an Adjunct Professor in the Electrical and Computer Engineering department at Saginaw Valley State University, where I taught several

undergraduate courses in data communications and computer hardware (ECE455 Data Communications, ECE368 Computer Engineering, ECE368L Computer Engineering Lab, ECE216 Digital Circuits, and ECE216L Digital Circuits Lab).

8. From 2012–2015, I was employed at Stryker Instruments as a Senior Principal Engineer in the Advanced Development department, where I conducted and supervised R&D of computationally intelligent medical instruments and surgical technologies, and I delivered working prototypes for bio-photonic devices and boundary-constraint surgical robotics.

9. From 2015–2018, I was employed at Interdigital, Inc. as a Senior Manager of Innovation Partners, where I conducted and supervised R&D of 5th generation wireless (5G) network-enabled technologies including the Internet of Things, connected/autonomous vehicles, and augmented/virtual reality infrastructure, and I managed scientists and engineers to create over 100 new-invention disclosures for patent protection.

10. From 2019–present, I have provided consulting services to a major patent-licensing corporation as a Technical Expert in the areas of semiconductor circuits, microchips, and data communications. My consulting services have included determining whether and how specific patent claims within the aforementioned areas read on various products or product features. I analyze patents, technical specifications, product white papers, and product marketing materials to

produce detailed claim charts used by attorneys to formulate infringement opinions and licensing decisions. I have also performed technology market analysis and developed detailed technology landscapes and roadmaps in the aforementioned areas.

11. I have written technical publications dating back to 2003 on topics in the areas of semiconductor circuits, microchips, and data communications. I have written technical articles published in peer reviewed professional journals, trade journals, and proceedings of peer-reviewed conferences. I have presented at several technical symposia and technical conferences on topics in the aforementioned areas.

12. I am or have been a member or senior member of several professional organizations, including the Institute of Electrical & Electronic Engineers (IEEE), the Association for Computing Machinery (ACM), the American Society of Mechanical Engineers (ASME), the International Society of Optics & Photonics (SPIE), and the American Chemical Society (ACS).

13. I served as a Program Evaluator for the Accreditation Board of Engineering and Technology (ABET) from 2013–2018. As a Program Evaluator, I was responsible for determining ABET accreditation for university engineering programs categorized as Electrical Engineering programs and Computer Engineering programs.

14. I served as an Industry Mentor for National Science Foundation (NSF) I-Corps program in 2013, and served as an Instructor for the same in 2015. In these roles, I provided industry and technical guidance to engineers, inventors, and entrepreneurs launching technology startups.

15. I hold approximately 30 U.S. Patents, many of which concern technologies in the areas of semiconductor circuits, microchips, and data communications.

16. Additional details on my background and experience can be found in my *curriculum vitae*, attached hereto as Appendix 1.

III. UNDERSTANDING OF LEGAL PRINCIPLES

A. Validity Generally

17. I have a general understanding of validity based on my experience with patents, my career experience, and my discussions with counsel. I have also reviewed the legal principles used to determine invalidity, such as those set forth in the AIPLA Model Patent Jury Instructions. *See, e.g.,* <https://www.aipla.org/home/news-publications/model-patent-jury-instructions>).

B. Prior Art

18. I have a general understanding of prior art and priority dates based on my experience with patents, my career experience, and my discussions with counsel.

19. I understand that inventors may be entitled to a priority date earlier than an actual date of filing of a patent application that provides written description

support for a particular claim to the extent that they can show a complete possession of such a particular claimed invention at such an earlier priority date, and reasonable diligence to reduce such a particular claimed invention to practice between such an earlier priority date and such an actual date of filing. I understand that if Patent Owner contends that particular claims are entitled to an earlier priority date than the actual date of filing, then Patent Owner has the burden to prove this contention with specificity.

20. I understand that an invention by another must be made before the priority date of a particular patent claim in order to qualify as “prior art” under 35 U.S.C. § 102 or 35 U.S.C. § 103. I also understand that a printed publication or a product usage must be publicly available before the priority date of a particular patent claim in order to qualify as “prior art” under 35 U.S.C. § 102(a), that a printed publication or a product usage or offer for sale must be publicly available in the United States more than one year prior to the actual date of filing of a patent application that provides written description support for a particular claim in order to qualify as “prior art” under 35 U.S.C. § 102(b), and that the invention by another must be described in an application for patent filed in the United States before the priority date of a particular patent claim in order to qualify as “prior art” under 35 U.S.C. § 102(e). I understand that Petitioners have the burden of proving that any particular reference or product usage or offer for sale is prior art.

C. Anticipation

21. I have a general understanding of anticipation based on my experience with patents, my career experience, and my discussions with counsel.

22. I understand that the anticipation analysis is a two-step process. The first step is to determine the meaning and scope of the challenged claims. Each claim must be viewed as a whole, and it is improper to ignore any element of the claim. For a claim to be anticipated under U.S. patent law: (1) each and every claim must be identically disclosed, either explicitly or inherently, in a single prior art reference; (2) the claim elements disclosed in the single prior art reference must be arranged in the same way as in the claims; and (3) the identical invention must be disclosed in the single prior art reference, in as complete detail as forth in the claim. Where even one element is not disclosed in a reference, the anticipation contention fails. Moreover, to serve as an anticipatory reference, the reference itself must be enabled. In other words, the reference must provide enough information so that a person of ordinary skill in the art (“POSITA”) can practice the subject matter of the reference without undue experimentation.

23. I further understand that where a prior art reference fails to explicitly disclose a claim element, the prior art reference inherently discloses the claim element only if the prior art reference must necessarily include the undisclosed claim element. Inherency may not be established by probabilities or possibilities. The fact

that an element may result from a given set of circumstances is not sufficient to prove inherency. I have applied these principles in forming my opinions in this matter.

24. I further understand that a dependent claim incorporates each and every limitation of the claim from which it depends. Thus, my understanding is that if a prior art reference fails to anticipate an independent claim, then that prior art reference also necessarily fails to anticipate all dependent claims that depend from that independent claim.

D. Obviousness

25. I have a general understanding of obviousness based on my experience with patents, my career experience, and my discussions with counsel.

26. I understand that a patent claim is invalid under 35 U.S.C. § 103 as being obvious only if the differences between the claimed invention and the prior art such that the subject matter as a whole would have been obvious at the time of invention to a POSITA. An obviousness analysis requires consideration of four factors: (1) scope and content of the prior art relied upon to challenge patentability; (2) differences between the prior art and the claimed invention; (3) level of ordinary skill in the art at the time of the invention; and (4) the objective evidence of non-obviousness, such as commercial success, unexpected results, the failure of others to achieve the results of the invention, a long-felt need which the invention fills, copying

of the invention by competitors, praise for the invention, skepticism for the invention, or independent development.

27. I understand that a prior art reference is proper to use in an obviousness determination if the prior art reference is analogous art to the claimed invention. I understand that a prior art reference is analogous art if at least one of the following two circumstances is met. First, a prior art reference is analogous art if it is from the same field of endeavor as the claimed invention, even if the prior art reference addresses a different problem and/or arrives at a different solution. Second, a prior art reference is analogous art if the prior art reference is reasonably pertinent to problem faced by the inventor, even if it is not in the same field of endeavor as the claimed invention.

28. I understand that it must be shown that a POSITA at the time of the invention would have had a reasonable expectation that a modification or combination of one or more prior art references would have succeeded. Furthermore, I understand that a claim may be obvious in view of a single prior art reference, without the need to combine references, if the elements of the claim that are not found in the reference can be supplied by the knowledge or common sense of one of ordinary skill in the relevant art. However, I understand that it is inappropriate to resolve obviousness issues by a retrospective analysis or hindsight reconstruction of

the prior art and that the use of such hindsight reconstruction is improper in analyzing the obviousness of a patent claim.

29. I further understand that the law recognizes several specific guidelines that inform the obviousness analysis. First, I understand that a hindsight reconstruction approach is not permitted. In other words, it is improper to use post-invention information to help perform the selection and combination of prior art references, and it is also improper to use the listing of elements in a claim as a blueprint to identify selected portions of different prior art references in an attempt to show that the claim is obvious.

30. Second, I understand that any prior art that specifically teaches away from the claim subject matter points to non-obviousness. In other words, prior art that would lead a POSITA to a specifically different solution than the claimed invention points to non-obviousness. Conversely, I understand that any prior art that contains any teaching, suggestion, or motivation to modify or combine such prior art references points to obviousness of such a modification or combination.

31. Third, while many combinations of the prior art may be “obvious to try,” I understand that any obvious-to-try analysis will not render a patent invalid unless it is shown that the possible combinations are: (1) sufficiently small in number so as to be reasonable to include that the combination would have been selected; and (2) such

that the combination would have been believed to be one that would produce predictable and well understood results.

32. Fourth, I understand that if a claimed invention that arises from the modification or combination of one or more prior art references uses known methods or techniques that yield predictable results, then that factor also points to obviousness.

33. Fifth, I understand that if a claimed invention that arises from the modification or combination of one or more prior art references is the result of known work in one field prompting variations of it for use in the same field or a different one based on design incentives or other market forces that yield predictable variations, then that factor also points to obviousness.

34. Sixth, I understand that if a claimed invention that arises from the modification or combination of one or more prior art references is the result of routine optimization, then that factor also points to obviousness.

35. Seventh, I understand that if a claimed invention that arises from the modification or combination of one or more prior art references is the result of a substitution of one known prior art element for another known prior art element to yield predictable results, then that factor also points to obviousness.

36. I further understand that a dependent claim incorporates each and every limitation of the claim from which it depends. Thus, my understanding is that if a prior art reference fails or a combination of prior art references fail to render obvious

an independent claim, then that prior art reference fails or that combination of prior art references fail to render obvious all dependent claims that depend from the independent claim.

IV. PERSON OF ORDINARY SKILL IN THE ART

37. I understand that, in assessing the level of ordinary skill, one may consider several factors, including: (1) the education level of the inventor; (2) the education level of active workers in the field; (3) the types of problems encountered in the art; (4) prior art solutions to these problems; (5) the rate of innovation in the field; and (6) the sophistication of the technology.

38. I understand that the District Court in the TI Litigation adopted the following identification of a person of ordinary skill in the art (“POSITA”): “a POSITA would have at least a Bachelor’s degree in electrical engineering, computer engineering, or a related field, and as at least two years of work experience in the field of SoC and NoC technologies.” Claim Construction Order (Dkt. 150) (EX2022). I met and exceeded this level of skill as of 1999. I have applied the Court’s definition of a POSITA throughout this Declaration.

V. OVERVIEW OF THE TECHNICAL BACKGROUND AND THE ’2893 PATENT

A. Technical Background

39. A network-on-chip (“NoC”) is a data communication network on a microchip. In some ways, NoCs are like traditional computer networks, such as local

area networks (“LANs”) and/or wide area networks (“WANs”). For example, both traditional computer networks and NoCs interconnect devices or components: traditional computer networks commonly interconnect discrete devices, such as computers, printers, servers, and so on, whereas NoCs commonly interconnect integrated components, such as processors, cores, caches, memories, input/output (I/O) interfaces, peripherals, and so on. However, implementation of a NoC on a SoC presents many unique challenges, as discussed in more detail below. *Infra*, § VI.B. The components interconnected by a NoC are typically integrated on a very large-scale integrated (“VLSI”) circuit, such as a system-on-chip (“SoC”).

40. An SoC is an integrated circuit in which most or all components of a system are integrated, including memories, processors, high-speed I/O interfaces, and dedicated application-specific logic. *See, e.g.*, EX2003, 1; *see also* EX2004, 4–5. By the early 2000s, certain limitations of statically wired interconnections between components of an SoC were known, and as a result, NoCs were already emerging. *See, e.g.*, EX2004, 7–8, *see also* EX2005. Rather than considering on-chip interconnections as point-to-point wires, they can be abstracted as communication channels over which to transmit data within a “quality-of-service” setting. *See id.*

41. Before describing details of a NoC, it is important to describe how components can be interconnected in an SoC that does not implement a NoC. Interconnections, sometimes called interconnects, traces, wires, and signal lines, are

electrical connections between a first component and a second component. Interconnects that carry similar signals are grouped together into buses, for example interconnects that carry address signals are grouped into an address bus and interconnects that carry data signals are grouped into a data bus. *See* EX2006, 4. A bus can be considered a direct interconnection between components (e.g., the path between a first component and a second component is always the same), where the interconnection may be dedicated or shared. A dedicated bus comprises point-to-point interconnection between components, typically between two components. A shared bus comprises interconnections that can be shared between many components, and requires some type of bus arbitration to resolve conflicts when multiple components attempt to access the shared bus simultaneously.

42. There are several types of bus arbitration methods for shared buses, including centralized and distributed (sometimes called decentralized). In centralized arbitration, a single device, known as a bus controller or bus arbiter, is responsible for managing access to the bus. In distributed arbitration, components compete for access to the bus by sending a request signal to the bus arbiter and waiting for a grant signal. In some distributed arbitration methods, e.g., decentralized, each component has its own priority level, and the component with the highest priority is given access to the bus. A component capable of originating a bus transaction (e.g., requesting control of the bus) is a master component, and a component that cannot originate a

bus transaction (e.g., only listens on the bus for communications intended for it) is a slave component. In large and/or complex SoCs, such shared buses interconnecting master devices and slave devices via bus arbitration may not be sufficiently scalable nor provide enough bandwidth. *See* EX2007, 7, 35.

43. In particular, with increasing quantities of components integrated in SoCs, increasing transistor densities, and increasing clock frequencies, it becomes very difficult to operate all the components synchronously within a same clock domain. This creates a need for a modular approach, where the processing system of the SoC comprises a plurality of relatively independent complex modules. In conventional processing systems, the system modules usually communicate with each other via a shared bus as described above. But as the number of modules increases, shared-bus communication is no longer practical at least because: (1) the large quantity of modules creates too high of an electrical load on the shared bus, and (2) the shared bus forms a communication bottleneck because only one component can control the bus at a time. Consequently, some SoCs implement shared buses that utilize crossbar switches, also called crossbars, cross-point switch, matrix switch, or switch matrix, to achieve higher throughput, with STBus, IBM CoreConnect, and ARM AMBA being several examples of bus standards that may utilize crossbars. *See* EX2007, 320. A crossbar has a plurality of input lines and output lines that form a crossed pattern of intersecting lines between which a connection may be established

by closing a switch located at a given intersection. An n -input, m -output crossbar, i.e., an $n \times m$ crossbar, is the equivalent to a set of m multiplexers (one per output port), each having n inputs (one per input port). *See* EX2007, 323–333. However, a shared bus that utilizes crossbars still has disadvantages. For example, the physical implementation of complex SoCs require large and complex crossbars that are difficult to place-and-route during physical design on the SoC in the areas between the components that implement functional logic of the SoC. *See* EX2008, 3. An on-chip communication network, e.g., a NoC, is an effective way to overcome some of the aforementioned disadvantages.

44. A NoC interconnect paradigm radically changes the on-chip communication compared to direct interconnects, such as buses or switches. This is because of the multi-hop nature of a network, where components are not directly connected, but separated by one or more network nodes. This is in contrast with buses, where components are directly connected as described above. Several implications of this change include arbitration methods (which can no longer be centralized), and communication properties (e.g., ordering, flow control, and so on). In an SoC that implements a NoC, certain design-specific wiring structures of the SoC, typically the longer “global” wires that interconnect components, are replaced with a general-purpose on-chip communication network. EX2005, 2. Instead of connecting these top-level modules by routing dedicated wires, they are

connected to a network that routes packets between them. *See id.* In doing so, these network-connected modules, or components, can be thought of as “network clients” that communicate with each other through communication channels of a network that provide one or more levels of service, e.g., as may be achieved by various communication properties. *See id.*

45. The 2001 paper by Dally and Towles provides a basic example of a NoC implementation. *See* EX2005, 2–3. It describes dividing a chip into 16 “tiles,” where each tile comprises client logic, such as processors, DSPs, peripheral controllers, memory subsystems, and so on. The client logic within a tile (i.e., nearby logic) may communicate via direct interconnections, but client logic in different tiles (i.e., far-apart logic) must communicate only over the network. In other words, there are no top-level (global) bus-based interconnections, only interconnections provided by the network. Figure 1 of the Dally and Towles paper is reproduced below. *See* EX2005, 2.

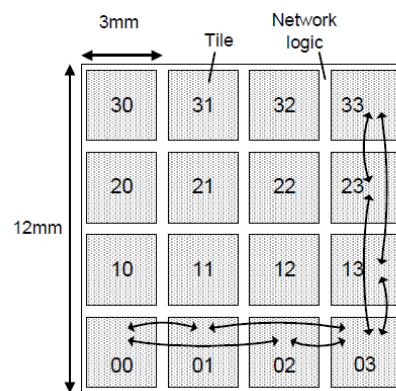


Figure 1 Partitioning the die into module tiles and network logic

46. Each tile interfaces with the network over an input port that is used to insert packets into the network and an output port over which the network delivers packets to the tile (where input and output here refer to packets going in and out of the network, not the tile). Each packet includes a virtual channel field that specifies which virtual channel it may be routed on, to provide different levels of service, and route field to specify a route through the network, i.e., the hops taken from source tile to destination tile. Such a port interface can present a simple and reliable level of service to the network clients. *See* EX2005, 3.

47. Logic local to the network clients can layer higher-level services on top of this interface. For example, this local logic could present a memory read/write service, a flow-controlled data stream, or a logical wire to the client. Local logic can also provide a translation from a destination node to a route. *See* EX2005, 3.

48. The router needed at each tile consists of five input controllers (one for each direction and one for input from the tile) and five output controllers. The router for each tile is distributed across the four edges of a tile with the input and output controller for a given direction (e.g., west) located on that edge of the tile. Figure 2 of the Dally and Towles paper, reproduced below, shows the west input controller and its connection to four output controllers (one for each of the other directions and one for the tile). *See* EX2005, 3.

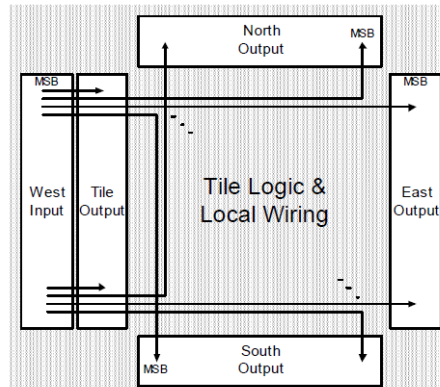


Figure 2 West input connections to output controllers

49. Details of an input controller and output controller are shown in Figure 3 of the Dally and Towles paper, reproduced below. Each input controller has an input buffer and input state logic for each virtual channel. When a head flit (flow control digit, e.g., part of a packet) arrives at each tile, the input controller strips the next entry off the route field and uses these two bits to select one of four output ports. The flit then arbitrates with the other virtual channels on that input port and, if it wins the arbitration, is forwarded to the output port. This arbitration and forwarding takes place in parallel with allocating a virtual channel and checking available buffer space to reduce latency. The output port provides a single stage of buffering for each input port connection. The flits in these buffers arbitrate for the link to the input controller on the next tile. Credits for buffer allocation are piggybacked on flits travelling in the reverse direction. *See* EX2005, 4.

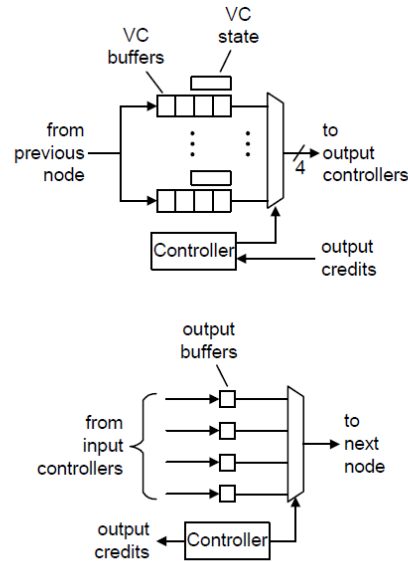


Figure 3 Input (top) and output controllers

50. As indicated earlier, the communication between modules (e.g., components, network clients, etc.) are performed over connections.

51. With connection properties, on-chip networks are typically able to provide varying levels of communication service, which may be referred to as a quality of service (QoS) levels. “The QoS is a very important feature in the interconnect infrastructures because it provides a regulation mechanism allowing specification of guarantees on some of the parameters related to the traffic.” EX2007, 325. Traffic here refers to a flow of data between a first module and a second module, for example, a series of data packets transmitted from the first module to the second module. With QoS, different traffic classes can be supported, such as:

- Real time and low latency (RTLL)—Traffic flows that require the lowest possible latency.
- Guaranteed throughput (GT)—Traffic flows that must maintain their throughput over a relatively long time interval.
- Guaranteed bandwidth (GBW)—Traffic flows that require a guaranteed amount of bandwidth over a relatively long time interval.
- Best effort (BE)—Traffic flows that do not require guaranteed latency or throughput but have an expectation of fairness.

EX2007, 326.

B. Challenges of On-Chip Networks

52. The issues and problems of implementing a multi-hop on-chip network interconnect are complex. Prior to the invention of a network-on-chip interconnect, functional blocks of an integrated circuit were typically interconnected using buses or crossbars. However, both buses and crossbars have respective drawbacks, as discussed in more detail below.

53. Buses, such as ARM’s Advanced Microcontroller Bus Architecture (AMBA), were an early solution for interconnecting on-chip functional blocks, e.g., initiators and targets. However, “the integration of multiple cores within chips” caused “too many initiators trying to access different targets simultaneously” which “created bottlenecks on the bus.” EX2009, 3. Buses also

had “latency issues.” *Id.* As a result, “[t]he industry wanted to enable concurrent access and more overall system data throughput.” *Id.*

54. Following the use of buses, crossbars were introduced as a next interconnect solution. While crossbars cure some of the deficiencies caused by buses, “crossbars require complex control logic to keep track of outstanding transactions” and require “wide data paths for the initiators and targets that need the most throughput as well as signaling for address and control information.” EX2009, 3. “To support continued system scaling, crossbars were cascaded with bridges between fabrics” which “carried a significant cost in silicon area” and “limited clock frequencies and added significant latency to transactions.” *Id.* at 4. Moreover, “during the 2000s, the initiators and targets became so numerous and widely distributed in the physical floorplan of a chip that the crossbar became a physical wiring nightmare.” *Id.*

55. Due to the drawbacks of buses and crossbars, “Network-on-Chip (NoC) technology emerged as a solution to the wiring problem.” EX2009, 5. NoC technology packetizes address and data information on the same wires, thereby allowing transactions to be sent over a smaller number of wires compared to prior interconnect solutions “while maintaining high quality of service for each transmission.” *Id.* NoC technology also contributed to decreased die density by

“distributing the interconnect logic throughout the chip rather than having bridges as chokepoints.” *Id.*

56. Engineers at Koninklijke Philips Electronics (the original assignor of the '2893 Patent) identified the same issues with existing interconnects and described the distinction between prior interconnects (e.g., buses and crossbars) and a multi-hop interconnect in patents related to network-on-chip technology.

NoCs help resolve the electrical problems in new deep-submicron technologies, as they structure and manage global wires. At the same time they share wires, lowering their number and increasing their utilization. ... NoCs also decouple computation from communication, which is essential in managing the design of billion-transistor chips. NoCs achieve this decoupling because they are traditionally designed using protocol stacks, which provide well-defined interfaces separating communication service usage from service implementation.

EX2017 (U.S. Patent No. 7,373,449 (the “’449 Patent”)). Moreover, using multi-hop interconnects “radically changes the communication when compared to direct interconnects, such as buses or switches.” *Id.* at 3:46–57 (“This is because of the multi-hop nature of a network, where communication modules are not directly connected, but separated by one or more network nodes. This is in contrast with the prevalent existing interconnects (i.e., buses) where modules are directly connected. The implications of this change reside in the arbitration (which must change from centralized to distributed), and in the communication properties (e.g., ordering, or flow control.)”).

57. The Specific examples of some differences between a multi-hop interconnect and a bus include: (1) media arbitration; (2) data routing; (3) data formatting; (4) latency; and (5) flow control for input buffers. Regarding media arbitration, “[i]n a bus, master modules request access to the interconnect, and the arbiter grants the access for the whole interconnect at once,” whereas in a NoC, “the arbitration is distributed, because it is performed in every router, and is based only on local information.” EX2017 at 5:25–42.

58. Regarding data routing, “[f]or a bus, the command, address, and data are broadcasted on the interconnect. They arrive at every destination, of which one activates based on the broadcasted address, and executes the requested command,” whereas in a NoC, “it is not feasible to broadcast information to all destinations, because it must be copied to all routers and network interfaces. This floods the network with data. The address is better decoded at the source to find a route to the destination module.” EX2017 at 5:43–54.

59. Regarding data formatting, “[i]n modern bus interfaces the data format is defined by separate wire groups for the transaction type, address, write data, read data, and return acknowledgements/errors (e.g., VCI, OCP, AMBA, or CoreConnect),” whereas in a NoC, “there is typically no distinction between different kinds of data. Data is treated uniformly, and passed from one router to another.” EX2017 at 6:8–25.

60. Regarding latency:

[L]atency is caused by two factors: (a) the access time to the bus, which is the time until the bus is granted, and (b) the latency introduced by the interconnect to transfer the data. For a bus, where the arbitration is centralized the access time is proportional to the number of masters connected to the bus. The transfer latency itself typically is constant and relatively fast, because the modules are linked directly. However, the speed of transfer is limited by the bus speed, which is relatively low. In a NoC, arbitration is performed at each router for the following link. The access time per router is small. Both end-to-end access time and transport time increase proportionally to the number of hops between master and slave. However, network links are unidirectional and point to point, and hence can run at higher frequencies than buses, thus lowering the latency. From a latency prospective, using a bus or a network is a trade off between the number of modules connected to the interconnect (which affects access time), the speed of the interconnect, and the network topology.

EX2017 at 5:55–6:7.

61. Multi-hop interconnects also differ from buses with respect to flow control for input buffers.

For buses, the source and destination are directly linked, and, destination can therefore signal directly to a source that it cannot accept data. This information can even be available to the arbiter, such that the bus is not granted to a transaction trying to write to a full buffer. In a NoC, however, the destination of a transaction cannot signal directly to a source that its input buffer is full. Consequently, transactions to a destination can be started, possibly from multiple sources, after the destinations input buffer has filled up. If an input buffer is full, additional incoming transitions are not accepted, and stored in the network. However, this approach can easily lead [sic] to network congestion, as the data could be eventually stored all the way to the sources, blocking the links in between.

EX2017 at 6:48–62.

62. The '449 Patent also emphasizes that “NoC’s premises are different from off-chip networks, and, therefore, most of the network design choices must be reevaluated” because “[o]n-chip networks have different properties (e.g., tighter link synchronization) and constraints (e.g., higher memory cost) leading to different design choices, which ultimately affect the network services.” EX2017 at 1:66–2:5. Another distinction between an on-chip network compared to off-chip networks is that “[a]n off-chip network interface usually contains a dedicated processor to implement the protocol stack up to network layer or even higher, to relieve the host processor from the communication processing;” however, “[i]ncluding a dedicated processor in a network interface is not feasible on chip, as the size of the network interface will become comparable to or larger than the IP to be connected to the network.” EX2017 at 2:16–25.

63. Further, data sent via a NoC may be subject to out-of-order arrival, diverse routing paths, and retransmission after dropping.

In a network, data sent from a source to a destination may arrive out of order due to reordering in networks nodes, following different routes, or retransmission after dropping. For off-chip networks out-of-order data deliver is typical. However, for NoCs where no data is dropped, data can be forced to follow the same path between a source and a destination (deterministic routing) with no reordering. This in-order data transportation requires less buffer-space, and reordering modules are no longer necessary.

EX2017 at 3:8–17.

64. On-chip networks also differ from off-chip networks with respect to time-related guarantees.

Off-chip networks typically use packet switching and offer best-effort services. Contention can occur at each network node, making latency guarantees very hard to offer. Throughput guarantees can still be offered using schemes such as rate-based switching or deadline-based packet switching, but with high buffering costs. An alternative to provide such time-related guarantees is to use time-division multiple access (TDMA) circuits, where every circuit is dedicated to a network connection. Circuits provide guarantees at a relatively low memory and computation cost. Network resource utilization is increased when the network architecture allows any left-over guaranteed band width to be used by best-effort communication.

EX2017 at 3:34–46.

65. On-chip networks can also provide more reliable communication compared to off-chip networks.

A consequence of the tight on-chip resource constraints is that the network components (i.e., routers and network interfaces) must be fairly simple to minimize computation and memory requirements. Luckily, on-chip wires provide a reliable communication medium, which can help to avoid the considerable overhead incurred by off-chip networks for providing reliable communication. Data integrity can be provided at low cost at the data link layer. However, data loss also depends on the network architecture, as in most computer networks data is simply dropped if congestion occurs in the network.

EX2017 at 2:43–53

66. On-chip networks also differ from off-chip networks because of their constraints and synchronization.

Typically, resource constraints are tighter on chip than off chip. Storage (i.e., memory) and computation resources are relatively more

expensive, whereas the number of point-to-point links is larger on chip than off chip. Storage is expensive, because general-purpose on-chip memory, such as RAMS, occupy a large area. Having the memory distributed in the network components in relatively small sizes is even worse, as the overhead area in the memory then becomes dominant.

EX2017 at 2:6–15

67. Given the technical challenges, I do not agree with Dr. Hassoun's simplistic assertion that the prior art technologies (e.g., bus-based interconnects and traditional computer networks) could be easily combined without undue experimentation by a person of ordinary skill in the art. A POSITA would have been dissuaded from attempting to modify or combine different prior art to arrive at the respective claimed inventions. My observations and opinions in this respect apply to all alleged modifications and combinations proposed by Dr. Hassoun.

68. I note that all of the elected references suffer from the issues stated immediately above. There is no teaching, and Dr. Hassoun never cites any teachings, for how the prior art systems could be modified or combined to arrive at the claimed multi-hop interconnect that provides the characteristics described above, for example, quality of service guarantees. In my opinion, arriving at this limitation would require extensive re-design, which is neither suggested nor alluded to, and would raise the difficulties and challenges discussed herein.

C. Overview of the '2893 Patent

69. The '2893 Patent relates to “an integrated circuit having a plurality of functional blocks interconnected via a data communication network.” EX1001 at 1:5–7. The data communication network comprises “a plurality of network stations being interconnected via a plurality of communication channels for communicating data packages between the functional blocks, each data package comprising N data elements including a data element comprising routing information for the network stations.” *Id.* At 1:65–2:6. The '2893 Patent identifies that “[a] problem associated with communication channels . . . is that the clock speed at which the [integrated circuit] can operate may be determined by the slowest communication channel, i.e. the channel exhibit the largest data transfer delay.” *Id.* 1:34–38.

70. The '2893 Patent teaches that “[t]o avoid having to limit the communication speed, e.g. the clock speed, of the data communication network 100 in accordance with the delay characteristics of the communication channel 150 between two network stations ... [the] communication channel ... is equipped with M*N data storage elements to introduce a delay of M*N cycles into the communication channel, with N corresponding to the size of a data package.” EX1001 at 5:49–58. This allows for “an increase of the maximum communication frequency of the data network 100 at the expense of the introduction of a delay of a

single data package, i.e. the number of clock cycles it takes to communicate such a data package, on the slow communication channel.” *Id.* 5:58–62.

71. Thus, the ’2893 Patent discloses an integrated circuit including an on-chip network where data storage elements are introduced on certain slower communication channels to introduce a delay to resynchronize the slower channels with other channels.

1. Challenged Claims of the ’2893 Patent

72. I understand that Petitioners challenge the following claims of the ’2893 Patent in this IPR: Claims 1–3, 6–7, and 9–10.

73. The exact language of independent claim 1 is reproduced below:

1. [preamble] An integrated circuit comprising:

[a] a plurality of functional blocks; and

[b] a data communication network comprising a plurality of network stations being interconnected via a plurality of communication channels for communicating data packages between the functional blocks, each data package comprising N data elements including a data element comprising routing information for the network stations, N being an integer of at least two, the plurality of network stations comprising a plurality of data routers and a plurality of network interfaces, each of the data routers being coupled to a functional block via a network interface, the data communication network comprising a first network station and a second network station interconnected through a first communication channel, the data communication network further comprising $M*N$ data storage elements, M being a positive integer, the data communication introducing a delay of $M*N$ cycles on the first communication channel when the data communication network identifies the first communication

channel as having a data transfer delay exceeding a predefined delay threshold.

74. The exact language of independent claim 7 is reproduced below:

7. [preamble] An integrated circuit comprising:

[a] a plurality of functional blocks; and

[b] a data communication network comprising a plurality of network stations being interconnected via a plurality of communication channels for communicating data packages between the functional blocks, each data package comprising N data elements including a data element comprising routing information for the network stations, N being an integer of at least two, the plurality of network stations comprising a plurality of data routers and a plurality of network interfaces, each of the data routers being coupled to a functional block via a network interface, the data communication network comprising a first network station and a second network station interconnected through a first communication channel, the data communication network further comprising $M*N$ data storage elements, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel,

[c] wherein each communication channel of the plurality of communication channels comprises:

[d] a first subchannel between an input port of a first associated network station and an output port of a second associated network station; and

[e] a second subchannel between an input port of the second associated network station and an output port of the first associated network station, and

[f] wherein the first subchannel and the second subchannel of the first communication channel each comprise a delay introduced by $\frac{1}{2}*M*N$ data storage elements.

75. The exact language of independent claim 10 is reproduced below:

10. [preamble] A method of designing an integrated circuit comprising a plurality of functional blocks, and a data communication network comprising a plurality of network stations being interconnected via a plurality of communication channels for communicating data packages between the functional blocks, each data package comprising N data elements including a data element comprising routing information for the network stations, N being an integer of at least two, the plurality of network stations comprising a plurality of data routers and a plurality of network interfaces, each of the data routers being coupled to a functional block via a network interface; the method comprising the acts of:

[a] identifying a first communication channel between a first network station and a second network station that has a data transfer delay exceeding a predefined delay threshold; and

[b] in response to the identifying act, inserting $M*N$ data storage elements into the data communication network, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel.

2. The '2893 Patent's File History

76. The '2893 Patent issued from U.S. Application No. 11/912,175 ("the '175 application"). During prosecution of the '175 application, the Examiner rejected then pending claims 1–3, 6, and 10 under 35 U.S.C. §103(a) as being unpatentable over U.S. 7,362,763 ("Wybenga") in view of U.S. 7,197,009 ("Dispensa"). *See* EX1002, Office Action dated July 22, 2010. In response, the applicant made stylistic amendments to the claims, but did not amend the subject matter of the claims. *See* EX1002, Applicant's Amendments and Response dated October 15, 2010. Instead of amending the claims, the applicant submitted that the office action conceded that Wybenga does not disclose the claimed " $M*N$ data storage elements, M being a positive integer, for introducing a delay of $M*N$

cycles on the first communication channel.” *Id.* Applicant further submitted that Dispensa’s teaching that the “delay variation is the difference between the delay of link 411 and the delay of link 412” does not “disclose or suggest any storage elements, let alone disclosing or suggesting any ‘network comprising $M*N$ data storage elements, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel.” *Id.*

77. Thereafter, the Examiner issued another office action that rejected claims 1–3, 6, and 9 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent Publication No. 2002/0087716 (“Mustafa”) and rejected claim 10 under 35 U.S.C. § 103(a) as being unpatentable over Mustafa in view of U.S. Patent Publication No. 2004/019709 (“Selvaggi”). EX1002, *See* Office Action dated January 12, 2011. In response, the applicant amended then pending independent claim 10 to recite that “in response to the identifying act, inserting $M*N$ data storage elements into the data communication network, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel.” *See* EX1002, Applicant’s Amendments and Response dated April 12, 2011. Applicant submitted that neither Mustafa or Selvaggi, alone or in combination, “disclose or suggest introducing any delay, let alone ... **in response to identifying** that the first communication channel has a data transfer delay exceeding a predefined threshold,

inserting M*N data storage elements into the data communication network for introducing a delay of M*N cycles on the first communication channel.” *Id.*

78. Following the applicant’s amendment and response of April 12, 2011, the Examiner granted the ’175 application. EX1002, Notice of Allowance Dated August 8, 2011.

79. To the extent the file history is specifically relevant to my opinions, I will note it in the paragraphs below.

3. Priority Date

80. I understand that the earliest possible priority date for the ’2893 Patent is October 13, 2004.

D. Claim Constructions

81. I understand that the following terms of ’2893 Patent have proposed constructions by either the Petitioners or the Patent Owner in the related District Court litigations, or have been construed, either by agreement or by the Court.

The Disputed District Court Constructions

82. In the Qualcomm Litigation and Samsung Litigation, the parties disagreed with respect to the following terms. I understand that Petitioners state that they “do not believe the Board needs to resolve those disputes.” Petition, 8-9.

1. “each data package comprising N data elements” (claims 1, 7, and 10)

Term	Samsung/Qualcomm District Court Construction	NST District Court Construction
“each data package comprising N data elements”	“every data package communicated by the network stations has a fixed size of N data elements”	“data package” means “fixed-size arrangement of data,” “data elements” means “groups of data,” and the remaining language should be afforded its plain and ordinary meaning.

EX2033, Appendix 2; EX2040, 25.

2. “data storage elements” (claims 1, 3, 7, and 10)

Term	Qualcomm District Court Construction	NST District Court Construction
“data storage elements”	“each being a structure that stores one data element”	“Array of one or more elements that store data”

83. Samsung did not propose a construction for “data storage elements” in the district court case. *See* EX2033; EX2040, 30.

Agreed District Court Constructions

3. The preamble of claim 10

84. In the Qualcomm Litigation and Samsung Litigation, the parties agreed that the preamble of claim 10 should be construed as limiting. EX2033, Appendix 1; EX2040, 5. The same construction was adopted by the district court in the TI case. *See* EX2022, 58.

4. “network” (claims 1, 2, 7, 9, and 10)

85. In both the Qualcomm Litigation and Samsung Litigation, the parties agreed that “network” should be construed as “multi-hop interconnection between modules separated by one or more network nodes.” EX2040, 6; EX2033, Appendix 1. The same construction was adopted by the district court in the TI case. See EX2022, 58.

5. “integrated circuit” (claims 1-3, 6, 7, 9, and 10)

86. In both the Qualcomm Litigation and Samsung Litigation, the parties agreed that the term “integrated circuit” should be construed as “interconnected circuitry on a chip.” EX2040, 6; EX2033, Appendix 1. The district court in the TI case construed “integrated circuit” as “a combination of interconnected circuit elements inseparably associated on or within a continuous substrate.” EX2022, 59.

87. For all other claim terms in the challenged claims of the ’2893 Patent that are not identified above, I have applied the plain and ordinary that a POSITA would understand for such terms and/or constructions at the time of the invention.

VI. ANALYSIS OF THE ’2893 PATENT

A. Ground 1: Purported Anticipation by Bertozzi (Claims 1-3, 6-7, and 10)

88. In Ground 1, Dr. Hassoun opines that claims 1–3, 6–7, and 10 are anticipated by EX1005 (“Bertozzi”). See EX1003, ¶¶ 93–165.

1. Independent Claim 10

89. Claim 10 of the '2893 Patent describes a method of designing an integrated circuit that includes inserting a quantity of data storage elements ($M*N$) that is based on a specific quantity of data elements (N) in a data package. The quantity of data storage elements ($M*N$) is a function of the quantity of data elements (N). Claim 10 is reproduced below in relevant part:

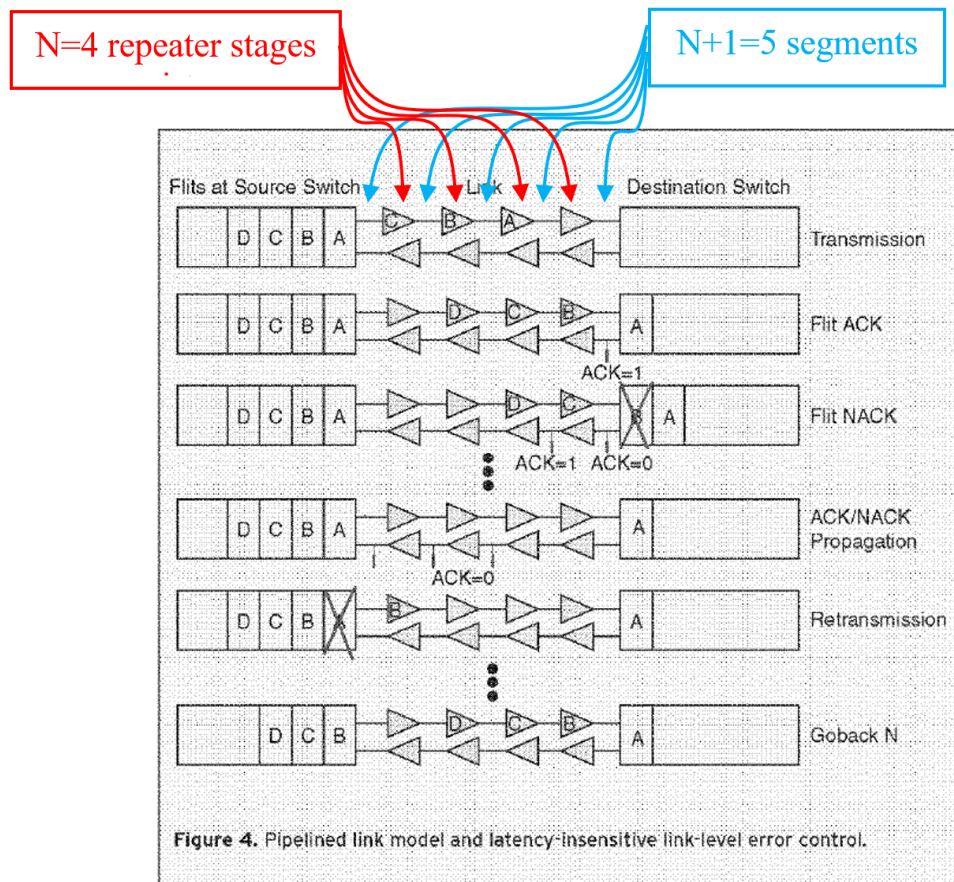
A method of designing an integrated circuit comprising . . . a plurality of network stations being interconnected via a plurality of communication channels for communicating data packages between the functional blocks, **each data package comprising N data elements** including a data element comprising routing information for the network stations, N being an integer of at least two, . . . the method comprising the acts of: . . . **inserting $M*N$ data storage elements into the data communication network**, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel.

EX1001, 12:22–42 (emphasis added). Bertozzi does not disclose “in response to the identifying act, **inserting $M*N$ data storage elements** into the data communication network, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel” (emphasis added), where the preamble recites “**each data package compris[es] N data elements**” (emphasis added), as recited in claim 10 of the '2893 Patent.

90. Bertozzi describes “a solution to overcome the interconnect-delay problem consists of pipelining interconnects [34, 35]. Wires can be partitioned into segments (or relay stations, which have a function similar to the one of latches in a

pipelined datapath) whose length satisfies pre-defined timing requirements.”

EX1005, 7 (brackets in original). FIG. 4 (reproduced below and annotated in color) shows an example of inserting four pipeline registers, i.e., repeater stages, onto a link that partitions the link into five segments. In general, inserting N repeater stages onto a link partitions the link into N+1 segments.



EX1005, FIG. 4

91. Bertozzi explains that “[b]ased on the clock speed target and link routing, the number of pipeline stages for each link is also determined.” EX1005, 12. Bertozzi further explains that “[s]witch-to-switch links are subdivided into basic segments whose length can be tailored to the desired clock frequency that

needs to be achieved in the design.” EX1005, 7. In other words, Bertozzi discloses that the quantity of repeater stages inserted onto a link is a function of a desired length of each segment. Bertozzi does not disclose or suggest that the quantity of repeater stages (e.g., data storage elements) inserted onto a link (e.g., channel) is a function of the quantity of flits, or phits (e.g., data elements) in a packet (e.g., data package). (The parentheticals in the previous sentence indicate the mapping between terminology of Bertozzi and terminology of claim 10 of the ’2893 Patent as provided in paragraphs 99–111 of EX1003 (Dr. Hassoun’s Expert Declaration).)

92. As an initial matter, Dr. Hassoun alleges that “A POSA would understand that Bertozzi means that each data packet is the same size or length,” (EX1003, ¶ 116), because Bertozzi states “[m]essages that have to be transmitted across the network are usually partitioned into fixed-length packets,” (EX1005, 5). However, “usually partitioned into fixed-length packets” says nothing about the length of each data packet that would result from partitioning a message. Partitioning means to “divide into parts,” not to divide into *equal* parts. *See, e.g.,* <https://www.dictionary.com/browse/partition>. Dr. Hassoun Declaration oddly cites to several references to define “fixed-length,” but fails to provide any evidence as to why a POSITA would understand partitioning of a message into packets to mean partitioning into equal-length packets. *See* EX1003, ¶116. Further, even if a first message in Bertozzi were partitioned into equal-length first packets, Dr. Hassoun’s

declaration provides no evidence that a second message, if partitioned, would result in the second packets having the same length as the first packets.

Consequently, Dr. Hassoun has not shown that Bertozzi discloses that each data packet is the same size or length. Indeed, Dr. Hassoun's Declaration provides inapposite justification that "[a] POSA would understand that there are various advantages associated with using fixed-length packets, including . . . timing across the NoC." EX1003, ¶117. Bertozzi explicitly states that correct functionality of its latency-insensitive NoC does not depend on timing:

Xpipes interconnect makes use of pipelined links and of latency-insensitive operation of its building blocks. . . . **If network switches are designed in such a way that their functional correctness depends on the flit arriving order and not on their timing**, the input links of the switches can be different and of any length. **These design choices are at the basis of latency insensitive operation of the NoC.**

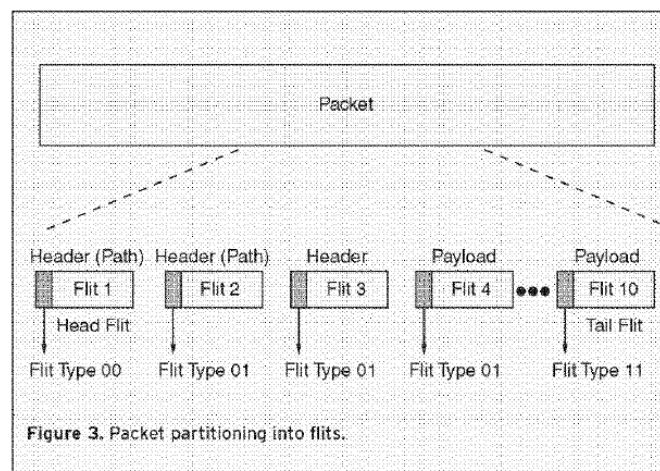
EX1005, 7–8 (emphasis added).

93. The invalidity opinion provided in Dr. Hassoun's Expert Declaration relies on FIG. 4 of Bertozzi. *See, e.g.*, EX1003, ¶ 77 ("Figure 4 of Bertozzi represents a specific embodiment of a data packet having four flits A, B, C, and D transmitted between two successive switches."); *see also id.*, ¶ 118 ("In Figure 4, Bertozzi shows a packet embodiment ('data package') that has a fixed length of N=4 data elements (in Bertozzi's implementation also called flits) lettered A through D"). However, Dr. Hassoun's Expert Declaration mischaracterizes FIG. 4

of Bertozzi. FIG. 4 of Bertozzi illustrates “[a] solution to the interconnect-delay problem consist[ing] of pipelining interconnects” where “**multiple outstanding flits** propagate across the link.” EX1003, 7 (emphasis added). FIG. 4 of Bertozzi (reproduced above) illustrates $N=4$ repeater stages on the link from the source switch to the destination switch and $N=4$ repeater stages on the link from the destination switch to the source switch. *See* EX1003, 8 (“an ACK is propagated back to the source, and after N clock cycles (where N is the link length expressed in terms of number of repeater stages)”); *see also id.*, 26 (“the flit has to propagate across the link (N cycles), . . . the ACK/NACK signal has to be propagated back (N cycles)”).

94. Bertozzi does not disclose or suggest that the multiple outstanding flits shown in FIG. 4 compose a packet of four flits (i.e., constitute a single packet). In fact, in my opinion, a POSITA would understand the four “outstanding flits” to constitute a portion of a single packet, specifically, as four outstanding flits of the 10-flit packet that is illustrated in FIG. 3 of Bertozzi (reproduced below). FIG. 3 of Bertozzi is the only place in Bertozzi that depicts a complete packet, and it consists of three header flits, six payload flits, and one tail flit (which is also considered a payload flit). Dr. Hassoun’s Declaration asserts “[a] POSA would understand that this [FIG. 3] is a general illustration of the partitioning procedure of a packet.” EX1003, ¶121, n. 3. However, based on this “general illustration of

the partitioning procedure,” an alleged four-flit packet would consist of three header flits and one tail flit—a rather useless packet with practically no payload, and therefore highly unlikely. Alternatively, a POSITA could understand the outstanding four flits as being comprised in multiple packets, such as the last two flits of an earlier packet and the first two flits of a later packet. In summary, Bertozzi does not disclose or suggest that the four flits in FIG. 4 of Bertozzi compose a single packet, and moreover, the most likely understanding that a POSITA would have is that the four flits in FIG. 4 of Bertozzi are, exactly as Bertozzi refers to them, “multiple outstanding flits” of a packet, such as the 10-flit packet shown in FIG. 3. The assertion that “Figure 4 of Bertozzi represents a specific embodiment of a data packet having four flits A, B, C, and D transmitted between two successive switches,” (EX1003, ¶ 77) is unsupported by evidence.



EX1005, FIG. 3

95. Bertozzi does not disclose or suggest that the quantity of repeater stages shown in FIG. 4 has any relationship to the quantity of flits shown in FIG. 4,

and moreover, that the quantity of repeater stages inserted onto a link is a function of the quantity of flits in a packet, as would be required by claim 10 of the '2893 Patent. To be clear, the illustration in FIG. 4 of Bertozzi of four “outstanding flits” to be transmitted over a link having four repeater stages is divorced from the requirement of claim 10 of the '2893 Patent that the quantity of data storage elements ($M*N$) inserted onto the data communication network is a function of the quantity of data elements (N) in a data package. Recall that Bertozzi refers to the flits in FIG. 4 simply as “[m]ultiple outstanding flits.” EX1005, 8.

96. For at least the reasons described above, Bertozzi does not disclose or suggest the limitation of claim 10 of the '2893 Patent that recites “in response to the identifying act, **inserting $M*N$ data storage elements** into the data communication network, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel” (emphasis added), where the preamble recites “**each data package compris[es] N data elements**” (emphasis added).

97. One reason why the quantity ($M*N$) of data storage elements is a function of the quantity of data elements (N) in a data package in claim 10 of the '2893 Patent is to ensure that the network stations receive routing information only during predefined evaluation cycles, i.e., routing-information periodicity. *See* EX1001, 2:26–33 (“Since a network station expects routing information to arrive

periodically (i.e. after every N cycles), the introduction of a delay on the first communication channel (which typically is the slowest channel of the network) matching such a period, i.e. by delaying the whole data package rather than a single data element, the delayed data package is received by the receiving network station during the appropriate period, e.g. clock cycle."); *see also id.*, 6:51–53 ("it is essential that the headers of the data packages do not arrive at any data router 110, 120, 130, 140 outside such an evaluation clock cycle."); *see also id.*, 6:60–67 ("due to the presence of $N=3$ data storage elements, . . . the header . . . is in phase with the header evaluation cycle of the data routers."); *see also id.*, 7:31–34 ("This ensures that the correct phase relation between clock cycle and routing information evaluation is maintained throughout the whole data communication network 100").

98. However, Bertozzi states that the switches utilized in Xpipes must operate correctly *regardless* of the quantity of repeater stages on a link. EX1005, 7 ("A solution to overcome the interconnect-delay problem consists of pipelining interconnects [34, 35]. . . . This requires the system to be composed of modules whose behavior does not depend on the latency of input communication channels [34].") (brackets in original) (citing L.P. Carloni, N.E. McMillan. and Sangiovanni-Vincentelli, "Theory of latency-insensitive design," *IEEE Trans. On CAD of ICs and Systems*, vol. 20, no. 9, pp. 1059-1076. Sept. 2001) (hereinafter, EX2011 or "Carloni"); EX1005, 8 ("Switch operation is latency insensitive, in that

correct operation is guaranteed for arbitrary link pipeline depth.”); *see also*

EX2011, 1065 (“[T]he latency-insensitive design methodology is to be able to ‘pipeline’ a communication channel by inserting an arbitrary amount of memory elements. In the framework of our theory, this operation corresponds to adding some particular processes, called relay stations, to the given system.”). Note that Carloni is reference 34 of Bertozzi. Accordingly, the solution disclosed by Bertozzi is not merely silent about ensuring that the switches receive routing information only during predefined evaluation cycles, but it explicitly teaches away from this routing-periodicity requirement described in the ’2893 Patent by stating that the switches *must* operate correctly for *any arbitrary quantity of repeater stages on a link*, i.e., for routing information that arrives at a destination switch in any cycle rather than in only predefined evaluation cycles. There is nothing explicit or inherent in Bertozzi that discloses or suggests the relationship that the quantity of data storage elements ($M \times N$) is a non-arbitrary quantity of data elements (N) in a data package as required by claim 10 of the ’2893 Patent.

99. Indeed, Bertozzi teaches that “[s]witch operation is latency insensitive, in that correct operation is guaranteed for arbitrary link pipeline depth.” EX1005, 8. Moreover, Bertozzi explains that its system supports “heterogenous architectures,” where “the input links of the switches can be different and of any length.” EX1005, 7–8. In other words, Bertozzi’s system is

explicitly designed to support links between switches that have different and arbitrary quantities of repeater stages. For example, one link may have three repeater stages, another link may have five repeater stages, and yet another link may have eight repeater stages—none of which are a quantity $M*N$ of repeater stages (M being a positive integer) given a data packet with N flits.

100. For at least these additional reason, Bertozzi does not disclose or suggest the limitation of claim 10 of the '2893 Patent that recites “in response to the identifying act, inserting $M*N$ data storage elements into the data communication network, M being a positive integer, for introducing a delay of $M*N$ cycles on the first communication channel” (emphasis added), where the preamble recites “each data package compris[es] N data elements” (emphasis added).

2. Independent Claim 1

101. I understand that Dr. Hassoun opines that Bertozzi discloses all limitations of independent claim 1 for the same reasons discussed for independent claim 10. See EX1003, ¶¶ 93–165. Thus, my opinions above regarding Bertozzi for claim 10 apply to claim 1 as well.

3. Independent Claim 7

102. I understand that Dr. Hassoun opines that Bertozzi discloses all limitations of independent claim 7 for the same reasons discussed for independent

claim 10. See EX1003, ¶¶ 93–165. Thus, my opinions above regarding Bertozzi for claim 10 apply to claim 1 as well.

4. Dependent Claims

103. The dependent claims are not anticipated by Bertozzi for at least the same reasons that the independent claims from which they depend are not anticipated by Bertozzi.

B. Ground 2: Purported Obviousness over Bertozzi Alone or in View of Jantsch or Duato (Claims 1–3, 6–7, and 10)

104. In Ground 2, Dr. Hassoun opines that claims 1–3, 6–7, and 10 are rendered obvious over EX1005 (“Bertozzi”) alone or in view of EX1006 (“Jantsch”) or EX1009 (“Duato”). *See* EX1003, ¶¶164–189. I understand that in Ground 2, Dr. Hassoun argues that Bertozzi, Jantsch, and Duato at least render the limitation “in response to the identifying act, inserting $M \times N$ data storage elements into the data communication network, M being a positive integer, for introducing a delay of $M \times N$ cycles on the first communication channel” of each challenged claim obvious for the same reasons discussed in Ground 1. *See* EX1003, ¶¶164–189. Nothing disclosed or suggested in Bertozzi, Jantsch, or Duato cure the deficiencies I discussed above with respect to Ground 1. Thus, my opinions above with respect to the arguments made for Ground 1 of Dr. Hassoun’s Declaration apply to Ground 2 as well.

105. For example, Dr. Hassoun opines that “it was well-known that synchronization of certain networks and types of communications require packets to arrive at network stations ‘at precisely scheduled time slots’,” and provides seven pages of inapposite discussion about such synchronization. EX1003, ¶180. As I discussed for Ground 1, Bertozzi states that the switches utilized in Xpipes must operate correctly for data packets arriving at any time slot, not just for precisely scheduled time slots. *See* EX1005, 7 (“A solution to overcome the interconnect-delay problem consists of pipelining interconnects [34, 35]. . . . This requires the system to be composed of modules whose behavior does not depend on the latency of input communication channels [34].”) (brackets in original) (citing EX2011); EX1005, 8 (“Switch operation is latency insensitive, in that correct operation is guaranteed for arbitrary link pipeline depth.”); *see also* EX2011, 1065 (“[T]he latency-insensitive design methodology is to be able to ‘pipeline’ a communication channel by inserting an arbitrary amount of memory elements. In the framework of our theory, this operation corresponds to adding some particular processes, called relay stations, to the given system.”). Note that Carloni is reference 34 of Bertozzi. Accordingly, the solution disclosed by Bertozzi is not merely silent about ensuring that packets arrive at switches at precisely scheduled time slots, e.g., that the switches receive routing information of only during predefined evaluation cycles, but Bertozzi explicitly teaches away from this

because the switches must operate correctly for any arbitrary quantity of repeater stages on a link, i.e., for routing information that arrives at a destination switch in any cycle rather than in only predefined evaluation cycles.

C. Ground 3: Purported Obviousness over Dielissen in View of Hassoun (Claims 1–3, 6, 7, 9, and 10)

106. In Ground 3, Dr. Hassoun opines that claims 1–3, 6, 7, 9, and 10 are rendered obvious over EX1007 (“Dielissen”) in view of EX1011 (“Hassoun”). *See* EX1003, ¶¶ 190–248.

1. Independent Claim 10

107. Dr. Hassoun concedes that Dielissen does not disclose the limitations of claim 10 that concerns inserting $M*N$ data storage elements in the network for introducing $M*N$ delay cycles on a channel, and turns to Hassoun for these limitations. *See* EX1003, ¶ 211. However, no combination of Hassoun, Dielissen, or any other references cited in the relevant section of Dr. Hassoun’s Declaration disclose or suggest this claim limitation. *See id.*, ¶¶ 211–227. Rather, Dr. Hassoun provides a 16-page narrative that reduces to a meritless obviousness argument:

- (1) Dielissen states that *Æthereal*’s data packets have three flits, but Dielissen never mentions channel delays or pipeline registers;
- (2) Hassoun provides an example minimum-delay registered buffered routing path with three pipeline registers, but Hassoun never mentions anything about a quantity of flits or phits within a data packet; and

(3) therefore, Dielissen in view of Hassoun discloses inserting a specific quantity $M*N = 1*3$ delay storage elements onto a channel where $N=3$ is the quantity of data elements in a data package, because “[d]ealing with delays caused by global wires or other long wires was a known issue in on-chip interconnects.” EX1003, ¶ 212; *see also id.* ¶¶ 213–215.

108. First, Dielissen describes several aspects of the Æthereal NoC. Dielissen at 1 (“[W]e present the Æthereal NoC.”). Dielissen states that “[f]or the current Æthereal network, the flit size is chosen [to be] 3 to optimize the data clock frequency and control frequency.” EX1007, 2. Dielissen’s FIG. 4 (reproduced below) shows this example data packet.

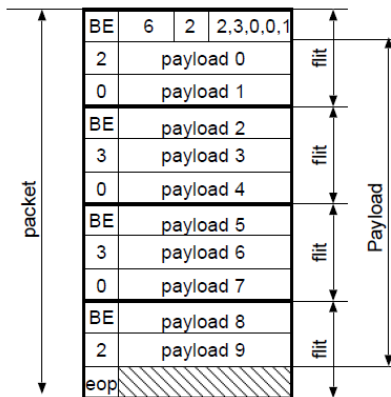


Figure 4. BE-packet example

EX1007, FIG. 4

Dielissen never mentions long wires, global wires, channel delays, channels having transfers delay exceeding predefined delay threshold, or an insertion or even

existence of any data storage elements (e.g., pipeline registers, pipeline stages, repeaters, repeater stages, etc.) in the network to introduce a delay on a channel.

See EX1007. Dielissen never discloses or suggests any need or desire to find a minimum delay path for a path, as taught by Hassoun, at least because the data clock frequency and control frequency for the *Æ*thereal NoC were already optimized by choosing a flit size of three. EX1007, 2 (“For the current *Æ*thereal network, the flit size is chosen [to be] 3 to optimize the data clock frequency and control frequency.”).

109. Second, Hassoun discloses an algorithm for finding a minimum-delay source-to-sink registered buffered routing path. EX1011, 1 (“The actual ‘Fast Path’ algorithm finds a minimum delay path for a net while simultaneously exploring all buffering and routing solutions.”); *see also id.*, 2 (“The Fast Path algorithm finds the minimum delay source-to-sink buffered routing path.”). Hassoun teaches a modification to the Fast Path algorithm, called the “Registered Buffered Path” algorithm, which accounts synchronization elements, such as registers, on the path. EX1011, 1 (“We extend the algorithm to handle additional synchronization elements such as registers, relay stations, and MCFIFOs.”). The Fast Path algorithm is shown in FIG. 1 below and the Registered Buffered Path algorithm is shown in FIG. 4 below, both annotated in red.

Fast Path (G, B, s, t, m')
Input: $G(V, E) \equiv$ Routing grid graph $B \equiv$ Buffer library $s \equiv$ source node $t \equiv$ sink node $m' \equiv$ initial labeling
Vars: $Q \equiv$ priority queue of candidates $\alpha = (c, d, m, v) \equiv$ Candidate at v
Output: $m \equiv$ Complete labeling of s - t path
$Q \leftarrow \{(C(m'(t)), 0, m', t)\}$. while ($Q \neq \emptyset$) do $(c, m, b, u) \leftarrow \text{extract_min}(Q)$ if $c = 0$ then return labeling m. if $u = s$ then $d' \leftarrow d + R(m(s)) \cdot c + K(m(s))$ push $(0, d', m, u)$ onto Q and prune. continue for each $(u, v) \in E$ do $c' \leftarrow c + C(u, v)$ $d' \leftarrow d + R(u, v)(c + C(u, v))/2$ push (c', d', m, v) onto Q and prune if $p(u) = 1$ and $m(u) = 0$ then for each $b \in B$ do $c' \leftarrow C(b)$ $d' \leftarrow d + R(b) \cdot c + K(b)$ $m(u) = b$ push (c', d', m, v) onto Q and prune

Figure 1. The Fast Path algorithm.

RBP Algorithm ($G, B, s, t, m', r, T_\phi$)
Input: $G(V, E) \equiv$ Routing grid graph $B \equiv$ Buffer library $s \equiv$ source node $t \equiv$ sink node $m' \equiv$ initial labeling with $m'(s) = m'(t) = r$ $r \equiv$ register for clocking signal $T_\phi \equiv$ required clock period.
Vars: $Q \equiv$ priority queue of candidates $Q^* \equiv$ queue holding next candidate wave $\alpha = (c, d, m, v) \equiv$ Candidate at v $A \equiv$ Marking of registered nodes
Output: $m \equiv$ Labeling of complete s - t path
$Q \leftarrow \{(C(r), \text{Setup}(r), m', t)\}$. $Q^* = \emptyset$, $A(v) = 0$, $\forall v \in V$ while ($Q \neq \emptyset$) and ($Q^* \neq \emptyset$) do if ($Q = \emptyset$) then $Q = Q^*$, $Q^* = \emptyset$. $(c, d, m, u) \leftarrow \text{extract_min}(Q)$ if $u = s$ then $d' \leftarrow d + R(m(s)) \cdot c + K(m(s))$ if $d' \leq T_\phi$ then return labeling m. for each $(u, v) \in E$ do $c' \leftarrow c + C(u, v)$ $d' \leftarrow d + R(u, v)(c + C(u, v))/2$ if $d' \leq T_\phi - K(r) - \min(R(B \cup r))c'$ then push (c', d', m, v) onto Q and prune if $p(u) = 1$ and $m(u) = 0$ then for each $b \in B$ do $c' \leftarrow C(b)$, $m(u) = b$ $d' \leftarrow d + R(b) \cdot c + K(b)$ if $d' \leq T_\phi - K(r)$ then push (c', d', m, u) onto Q and prune if $A(u) = 0$ and $d + R(r) \cdot c + K(r) \leq T_\phi$ then $m(u) = r$, $A(u) = 1$ push $(C(r), \text{Setup}(r), m, u)$ onto Q^*

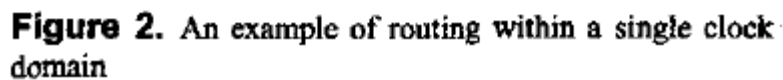
Figure 4. The Registered Buffered Path algorithm.

EX1011, FIGS 1, 4

110. The inputs to the Fast Path algorithm (FIG. 1 above left), highlighted in red, consist of a routing grid path $G(V, E)$, a buffer library B , a source node s , a sink node t , and an initial labeling m' . The inputs to the Registered Buffered Path algorithm (FIG. 4 above right), highlighted in red, consist of the same inputs as the Fast Path algorithm plus registers r and a required clock period T_ϕ . What is conspicuously missing from either algorithm, and especially the Registered

Buffered Path algorithm, is any indication or suggestion that a size or length of a data packet it utilized as an input (e.g., any information indicating that a data package comprises N data elements). This is because neither algorithm considers a size or length of a data packet. Indeed, the modification made to the Fast Path algorithm to create the Registered Buffered Path algorithm consists only of the registers and a required clock period.

111. Hassoun's "Figure 2 shows an example of a buffered-register path on a grid graph." EX1011, 3. FIG. 2 of Hassoun is reproduced below and annotated in red. The example shows three registers (annotated in red) on the path from source s to sink t . However, Hassoun provides no explanation as to why this routing example utilizes three registers. The opinion that "Hassoun teaches inserting three repeaters into the communication channel" exceeds Hassoun's actual disclosure. EX1003, ¶ 220. Hassoun's FIG. 2 simply discloses an example routing that is arbitrarily illustrated with some quantity (three) of registers. Further, because neither the Fast Path algorithm nor the Registered Buffered algorithm considers a size or length of a data packet, a POSITA would have no reason to believe that the quantity of three registers shown in Hassoun's FIG. 2 has, or could have, any correlation whatsoever with a quantity of flits in a data packet as suggested by Dr. Hassoun. EX1003, ¶¶ 222, 225.



EX1011, FIG. 2

113. Further, Dr. Hassoun has failed to show that a POSITA would have had a reasonable expectation that the combination of Dielissen and Hassoun would

have succeeded. *See* EX1003, ¶ 221 (“A POSA would have been motivated, with a reasonable expectation of success, to modify Dielissen in view of the teachings of Hassoun to first identify a data transfer delay exceeding a predefined threshold (given the known long wire problem) and to then predictably improve the chip by adding repeaters as taught by Hassoun to allow or maintain a higher clock rate within the chip.”). Recall that Dr. Hassoun concedes that Dielissen fails to disclose (at least) inserting $M*N$ data storage elements into the network (where N is the quantity of data elements in a data package) and that Dr. Hassoun relies on Hassoun for (at least) this limitation. But, as explained above, neither the Fast Path algorithm nor the Registered Buffered Path algorithm disclosed by Hassoun utilizes packet length (or flit length) as an input to determine its output, i.e., to determine a minimum-delay source-to-sink buffered or registered buffered routing path. Thus, modifying Dielissen to incorporate Hassoun’s teachings cannot succeed because Hassoun’s Fast Path algorithm or Registered Buffered Path algorithm do not produce their outputs (the alleged $M*N$ data storage elements) according to a packet length or a flit length (the alleged N data elements of a data package). Further, Dr. Hassoun fails to show how Hassoun’s Fast Path algorithm or Registered Buffered Path algorithms could be modified to produce their outputs according to a packet length or a flit length. Thus, any attempt at combining Dielissen and Hassoun would necessarily fail.

114. For at least the reasons described above, Dielissen in view of Hassoun does not disclose or suggest the limitation of claim 10 of the '2893 Patent that recites “in response to the identifying act, **inserting M*N data storage elements into the data communication network**, M being a positive integer, for introducing a delay of M*N cycles on the first communication channel” (emphasis added), where the preamble recites “**each data package compris[es] N data elements**” (emphasis added).

2. There is No Motivation to Combine Dielissen and Hassoun

115. Dr. Hassoun’s circuitous motivation-to-combine narrative reduces to a series of generalized observations disjointed from either Dielissen or Hassoun. My summaries of Dr. Hassoun’s paragraphs below are not endorsements of the content of those paragraphs:

- (1) Dielissen discusses a packet size as three flits and a flit size as three words. EX1003, ¶ 198; EX1007, 2, FIG. 4.
- (2) Chips have long wires. EX1003, ¶ 212. However, Dielissen never discusses this.
- (3) Long wires are slow. EX1003, ¶ 213. However, Dielissen never discusses this.
- (4) Long wires can be pipelined with registers. EX1003, ¶ 214–215.
However, Dielissen never discusses this.

(5) Pipelining can be tailored to a desired clock frequency. EX1003, ¶

216. However, Dielissen never discusses this.

(6) Adding pipeline registers to links was known. EX1003, ¶ 217.

However, Dielissen never discusses this.

(7) Pipelining long wires improves clock rate. EX1003, ¶ 218. However,

Dielissen never discusses this.

(8) Hassoun's registers store data and add delay. EX1003, ¶¶ 219–220.

However, Hassoun never discusses packet sizes.

(9) Therefore, a POSA would be motivated to combine Dielissen's

packet size with Hassoun pipeline registers. EX1003, ¶¶ 221–222.

116. To be clear, neither Dielissen, Hassoun, or any other reference cited by Dr. Hassoun provides any motivation to incorporate any specific teaching—i.e., the Registered Buffered Path algorithm of Hassoun—into the design or implementation of the NoC described by Dielissen. Further, neither Dielissen, Hassoun, or any other reference cited by Dr. Hassoun identifies a specific problem in Dielissen that could be solved by incorporating the Registered Buffered Path algorithm of Hassoun into the design or implementation of the NoC described by Dielissen. Finally, it would not be obvious to try incorporating the Registered Buffered Path algorithm of Hassoun into the design or implementation of the NoC described by Dielissen, and such combination would not succeed, because

Hassoun's Registered Buffered Path algorithm does not utilize a size of a packet as an input to produce its output.

3. Independent Claim 1

117. I understand that Dr. Hassoun opines that Dielissen in view of Hassoun discloses all limitations of independent claim 1 for the same reasons discussed for independent claim 10. See EX1003, ¶¶ 190–248. Thus, my opinions above regarding Dielissen in view of Hassoun for claim 10 apply to claim 1 as well.

4. Independent Claim 7

118. I understand that Dr. Hassoun opines that Dielissen in view of Hassoun discloses all limitations of independent claim 7 for the same reasons discussed for independent claim 10. See EX1003, ¶¶ 190–248. Thus, my opinions above regarding Dielissen in view of Hassoun for claim 10 apply to claim 1 as well.

5. Dependent Claims

119. The dependent claims are not obvious over Dielissen in view of Hassoun for at least the same reasons that the independent claims from which they depend are not obvious over Dielissen in view of Hassoun.

D. Ground 4: Purported Obviousness over Grounds 1–3 Further in View of Sullam (Claim 8)

120. In Ground 4, Dr. Hassoun opines that claim 8 is disclosed by EX1013 (“Sullam”) and that it would be obvious to combine Sullam with the references in Ground 1. See EX1003, ¶¶ 249–255. Claim 8 depends from claim 1 and is patentable for at least the same reasons as claim 1, as discussed above in relation to Grounds 1–3.

VII. CONCLUSION

121. In addition to the materials identified in Appendix 2, I may also review other documents in this matter. I may also review and comment upon the testimony of others that has been or will be given in this matter.

122. I reserve the right to add or to modify this expert declaration in the event that inaccuracies or omissions are discovered or if new or additional information is provided to me. I also reserve the right to make any such other changes as may be deemed necessary or appropriate.

123. I reserve the right to supplement and/or amend my analyses and opinions expressed herein in response to any new, additional, or modified positions taken by Petitioners, its experts or witnesses in reports, declarations, affidavits, or testimonies, whether in response to my report herein or in their coincident reports (or amendments thereto), to the extent such positions are relevant to the subject

matter I have addressed in my report herein, or to the extent such positions cover topics relevant to this report for which Petitioners have the burden of proof.

I declare under penalty of perjury under the laws of the United States that the foregoing is a true and correct reflection of my opinions. This declaration is executed this 27th day of March, 2024.

A handwritten signature in black ink, appearing to read "Erik Chmelar". The signature is fluid and cursive, with the first name "Erik" and last name "Chmelar" clearly distinguishable.

Dr. Erik Chmelar

APPENDIX 1

ERIK CHMELAR

(650) 533-7952 | echmelar@yahoo.com

Subject matter expert in semiconductor hardware and software with over two decades of experience developing, implementing, and protecting advanced technologies including digital and mixed-signal systems, high-speed serial communications, 5G/wireless telecommunications, and smart medical devices.

CREDENTIALS

EDUCATION —

- **JD, Intellectual Property**, Univ. Michigan Law School, Ann Arbor, MI 2015-2017
- **MBA, Management**, Western Michigan Univ., Kalamazoo, MI 2012-2013
- **BS, Chemical Physics**, Saginaw Valley State Univ., University Center, MI 2008-2011
- **PhD, Electrical Engineering**, Stanford Univ., Stanford, CA 2002-2004
Dissertation: “Test and Diagnosis of Field-programmable Gate Arrays.”
- **MS, Electrical Engineering**, Stanford Univ., Stanford, CA 2000-2001
- **BS, Electrical Engineering**, Michigan Technological Univ., Houghton, MI 1997-2000

LICENSES AND CERTIFICATIONS —

- **Agile Certified Practitioner** (PMI-ACP, PMI) 2015-pres.
- **Certified Project Management Professional** (PMP, PMI) 2014-pres.
- **Licensed Professional Engineer** (PE, State of Michigan LARA) 2010-pres.
- **Licensed Patent Agent/Attorney** (State Bar of Michigan, USPTO) 2007-pres.

EXPERIENCE

Technical Expert, Semiconductor Hardware/Software, Midland, MI 08/2007-pres.

- Serve as a technical expert witness for litigation in court and inter partes review (IPRs)
- Mine and analyze patent portfolios and develop assertion strategies for non-practicing entities (NPEs)
- Recent technology areas include memory, programmable hardware, microcontrollers, and AI devices

Senior Counsel, Young Basile Hanlon & MacFarlane, PC, Troy, MI 06/2023-pres.

- Draft and prosecute patent applications in a variety of semiconductor, software, and related arts
- Draft legal opinions for non-infringement, invalidity, and freedom-to-operate (FTO)
- Interview inventors, harvest inventions, and manage patent portfolios

Senior Associate, Dilworth IP, New Haven, CT 10/2020-06/2023

- Drafted and prosecuted patent applications in a variety of semiconductor, software, and related arts
- Drafted legal opinions for non-infringement, invalidity, freedom-to-operate (FTO), and export controls
- Interview inventors, harvest inventions, manage patent portfolios, and create technology landscapes

Senior Associate, Murphy, Bilak & Homiller, PLLC., Cary, NC 01/2020-06/2020

- Prosecuted patent applications for industry-leading wireless and semiconductor corporations
- Drafted approx. 100 office-action responses for complex 5G technologies
- Routinely referenced ETSI 3GPP specifications for prosecution of standard essential patents (SEPs)

Senior Manager, Innovation Partners, InterDigital, Inc., Wilmington, DE 09/2015-10/2018

- Created patent portfolios for 5G, IoT, connected/autonomous vehicles, and augmented/virtual reality
- Managed scientists and engineers to create over 100 new invention disclosures / patent applications
- Forged research partnerships with commercial and academic organizations to develop patent portfolios

APPENDIX 1

- Co-founder, Managing Member**, Dera Industries LLC., Midland, MI 05/2016-07/2017
- Developed and productized the award-winning One-Tie™ re-usable tie strap consumer product
 - Achieved \$1 million in annual sales prior to licensing the intellectual property
- Senior Principal Engineer, Adv. Dev.**, Stryker Corp., Kalamazoo, MI 02/2012-03/2015
- Supervised R&D of disruptive medical instruments and surgical technologies
 - Formalized new business opportunities, developed budgets and timelines, and staffed project teams
 - Collaborated with several top-tier universities to bring necessary expertise into the organization
 - Delivered working prototypes for biophotonic devices and boundary-constraint surgical robotics
- Staff SW Engineer, Adv. Tech. Dev.**, LSI Corp. (Broadcom), Milpitas, CA 07/2004-01/2012
- Conducted R&D for very large scale integrated (VLSI) and serializer-deserializer (SerDes) circuits
 - Granted approx. 20 patents for physical (PHY) interfaces, SerDes, and VLSI test architectures
 - Made cutting-edge contributions in SerDes and test that were implemented in customer designs
- Adjunct Professor, ECE Dept.**, SVSU, University Center, MI 01/2009-01/2012
- Taught courses in digital circuits, computer architecture and organization, and data communications
- Consulting Assistant Professor, EE/CS Dept.**, Stanford University, Stanford, CA 09/2006-08/2008
- Advised PhD students on the test and reliability of VLSI circuits
 - Created and taught a PhD-level course on clock and data recovery (CDR) for digital communications
- Product Test Engineer**, Xilinx, Inc., Los Gatos, CA (PhD. Intern, two summers) 06/2002-09/2003
- Engineered tools to detect and diagnose defects in FPGAs, which reduced diagnosis time by 75%
- Manufacturing Engineer**, Cisco Systems, Inc., San Jose, CA 11/1999-04/2002
- Oversaw functional testing of layer-2/3 network switches and determined root-cause failures

SERVICE AND MENTORING

- Admissions Representative**, Univ. Michigan Law School, Ann Arbor, MI 2018
- Advised prospective law students at 30 law fairs and forums
- Instructor**, National Science Foundation, Western Michigan Univ. I-Corps, Kalamazoo, MI 2015
- Taught Lean Startup and Business Canvas fundamentals to a cohort of 30 entrepreneurs
- Engineering Program Evaluator**, Accred. Board of Eng. & Tech. (ABET), Baltimore, MD 2013-2014
- Evaluated electrical engineering and computer engineering university educational programs
- Industry Mentor**, National Science Foundation, Univ. Michigan I-Corps, Ann Arbor, MI 2013
- Mentored teams of medical-device entrepreneurs participating in the I-Corps program
- Industry Mentor**, Western Michigan Univ., Starting Gate Accelerator, Kalamazoo, MI 2013
- Mentored entrepreneurs participating in the university startup-accelerator program

SELECT PROFICIENCIES

- | | |
|--|--|
| • Hardware – Verilog, VHDL, ASICs, FPGAs | • Software – C/C++, Perl, Java, MatLab |
| • Engineering – COMSOL, ANSYS | • CAD – SolidWorks, AutoCAD, Blender |

APPENDIX 1

Select US Patents

US11446109B2, “Metal detection system for use with medical waste container,” issued 2022-09-23.

US11389556B2, “Sterilization container capable of providing an indication regarding whether or not surgical instruments sterilized in the container were properly sterilized,” issued 2022-07-19.

US11357877B2, “Sterilization enclosure for surgical instruments,” issued 2022-06-14.

US§ 11225996B1, “Dual hook clasp,” issued 2022-01-18.

US11002308B2, “Hook device with opposing jaws,” issued 2021-05-11.

US11000341B2, “Metal detection system for use with medical waste container,” issued 2021-05-11.

US10994906B2, “Easy-to-remove cable tie,” issued 2021-05-04.

US10731698B2, “Hook device with rotatable opposing jaws,” issued 2020-08-04.

US10604317B2, “Reusable tie strap with multiple apertures,” issued 2020-03-31.

US10258706B2, “Sterilization container capable of providing an indication regarding whether or not surgical instruments sterilized in the container were properly sterilized,” issued 2019-04-16.

US10010636B2, “Sterilization container capable of providing an indication regarding whether or not surgical instruments sterilized in the container were properly sterilized,” issued 2018-07-03.

US9292644B2, “Row based analog standard cell layout design and methodology,” issued 2016-03-22.

US9087157B2, “Low-loss transmission line TDM communication link and system,” issued 2015-07-21.

US9014313B2, “Error signature analysis for data and clock recovery in a communication system,” issued 2015-04-21.

US8982941B2, “Predictive selection in a fully unrolled decision feedback equalizer,” issued 2015-03-17.

APPENDIX 1

US8929497B2, “Dynamic deskew for bang-bang timing recovery in a communication system,” issued 2015-01-06.

US8923382B2, “Tap adaptation with a fully unrolled decision feedback equalizer,” issued 2014-12-30.

US8615062B2, “Adaptation using error signature analysis in a communication system,” issued 2013-12-24.

US8527912B2, “Digitally obtaining contours of fabricated polygons,” issued 2013-09-03.

US8432250B2, “Process variation based microchip identification,” issued 2013-04-30.

US8121186B2, “Systems and methods for speculative signal equalization,” issued 2012-02-21.

US7956790B2, “Systems and methods for synchronous, retimed analog to digital conversion,” issued 2011-07-05.

US7973692B2, “Systems and methods for synchronous, retimed analog to digital conversion,” issued 2011-06-07.

US7779320B2, “Low power scan shifting with random-like test patterns,” issued 2010-08-17.

US7696915B2, “Analog-to-digital converter having reduced number of activated comparators,” issued 2010-04-13.

US7656340B2, “Systems and methods for pipelined analog to digital conversion,” issued 2010-02-02.

US7656339B2, “Systems and methods for analog to digital conversion,” issued 2010-02-02.

US7328386B2, “Methods for using checksums in X-tolerant test response compaction in scan-based testing of integrated circuits,” issued 2008-02-05.

US7293312B2, “Multipurpose skateboard tool,” issued 2007-11-13.

US7210083B2, “System and method for implementing postponed quasi-masking test output compression in integrated circuit,” issued 2007-04-24.

US7206983B2, “Segmented addressable scan architecture and method for implementing scan-based testing of integrated circuits,” issued 2007-04-17.

US7093842B2, “Skateboard truck assembly,” issued 2006-08-22.

APPENDIX 1

US6979142B1, “Retractable tip mechanical pencil assembly,” issued 2005-12-27.

US6920621B1, “Methods of testing for shorts in programmable logic devices using relative quiescent current measurements,” issued 2005-07-19.

APPENDIX 1

Select Publications

- E. Chmelar, “Error signature analysis (ESA) receiver architecture for data communication,” DesignCon, 2012.
- E. Chmelar and C. Ito, “Mostly digital SerDes (MDS): a comprehensive low power receiver architecture,” DesignCon, 2012.
- I. Park, D. Lee, E. Chmelar, and E. McCluskey, “Inconsistent fails due to limited tester timing accuracy,” Proc. 26th VLSI Test Symp., 2008.
- A. Al-Yamani, N. Devta-Prasanna, E. Chmelar, M. Grinchuk, and A. Gunda, “Scan test cost and power reduction through systematic scan reconfiguration,” IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 26, pp. 907–918, 2007.
- A. Al-Yamani, E. Chmelar, and M. Grinchuk, “Segmented addressable scan architecture,” Proc. 23rd VLSI Test Symp., 2005.
- E. Chmelar, “Minimizing the number of test configurations for FPGAs,” Int. Conf. Computer Aided Design, 2004.
- E. Chmelar, “The test and diagnosis of FPGAs,” Ph.D. thesis, Stanford University, 2004.
- E. Chmelar and S. Toutounchi, “FPGA bridging fault detection and location via differential IDDQ,” Proc. 22nd VLSI Test Symp., 2004.
- E. Chmelar, “Subframe multiplexing: FPGA manufacturing test time reduction,” CRC TR-04-01, 2004.
- E. Chmelar, “Subframe multiplexing for FPGA manufacturing test,” Proc. Int. Symp. FPGAs, 2004.
- E. Chmelar, “FPGA interconnect delay fault testing,” Proc. Int. Test Conf., pp. 1239–1247, 2003.

APPENDIX 1

Select Professional Organizations

State Bar of Michigan
Member (2018-pres.)

Project Management Institute
Certified Agile Practitioner (2014-pres.)
Certified Project Management Professional (2015-pres.)

Association for Computing Machinery
Member (2018-2019)

Accreditation Board of Engineering and Technology
Program Evaluator (2013-2018)

Institute of Electrical & Electronic Engineers
Senior Member (2004-2018)

American Chemical Society
Member (2013-2016)

National Science Foundation I-Corps
Instructor (2015)
Industry Mentor (2013)

American Society of Mechanical Engineers
Member (2014-2015)

International Society of Optics & Photonics
Member (2015)

APPENDIX 2--LISTING OF MATERIALS CONSIDERED

Exhibit No.	Description
2003	Neil Weste & David Harris, CMOS VLSI DESIGN: A CIRCUIT AND SYSTEMS PERSPECTIVE (4th ed. 2010)
2004	Jan Rabaey, Anantha Chandrakasan & Borivoje Nikolic, DIGITAL INTEGRATED CIRCUITS: A DESIGN PERSPECTIVE (2nd ed. 2003)
2005	William J. Dally & Brian Towles, <i>Route Packets, Not Wires: On-Chip Interconnection Networks</i> , PROC. 38TH DESIGN AUTOMATION CONF. (2001)
2006	John F. Wakerly, DIGITAL DESIGN: PRINCIPLES AND PRACTICES (4th ed. 2005)
2007	Fayez Gebali, Haytham Elmiligi & Mohamed Watheq El-Kharashi, NETWORKS-ON-CHIPS: THEORY AND PRACTICE (2009)
2008	Kurt Shuler, <i>Busses, Crossbars and NoCs: The 3 Eras of SoC Interconnect History</i> , ARTERISIP (Oct. 22, 2010), https://www.arteris.com/blog/busses-crossbars-and-nocs-the-3-eras-of-soc-interconnect-history/
2009	Kurt Shuler, <i>The SoC Interconnect Fabric: A Brief History</i> , ARTERISIP (Aug. 6, 2013), https://www.arteris.com/blog/the-soc-interconnect-fabric-a-brief-history/
2011	L.P. Carloni, N.E. McMillan, & Sangiovanni-Vincentelli, “Theory of Latency-Insensitive Design,” <i>IEEE Trans. On CAD of ICs and Systems</i> , vol. 20, no. 9, (Sep. 2001)
2017	U.S. Patent No. 7,373,449
2022	Claim Construction Order, <i>Network System Technologies, LLC v. Texas Instruments Incorporated</i> , Civil Action No. 2:22-cv-00482-RWS (E.D. Tex.), Dkt. 150 (Jan. 3, 2024)
2033	P.R. 4-3 Claim Construction and Prehearing Statement, <i>Network System Technologies, LLC v. Samsung Electronics Co., Ltd.</i> , Civil Action No. 2:22-cv-00482-RWS (E.D. Tex.), Dkt. 91 (Jan. 9, 2024)

Exhibit No.	Description
2040	Defendants' Joint Opening Claim Construction Brief, <i>Network System Technologies, LLC v. Qualcomm Incorporated et al.</i> , Civil Action No. 1:22-cv-1331-DAE (W.D. Tex.), Dkt. 77 (Oct. 12, 2023)
	2023-12-20 - Paper 1 - Petition for Inter Partes Review of US 8,072,893
1001	U.S. Patent No. 8,072,893
1002	Prosecution History of U.S. Patent No. 8,072,893
1003	Declaration of Dr. Marwan Hassoun
1005	D. Bertozzi, L. Benini, " <i>Xpipes: A Network-on-Chip Architecture for Gigascale Systems-on-chip</i> ", IEEE Circuits and Systems Magazine (2004) ("Bertozzi")
1006	Networks on Chip, Axel Jantsch, Hannu Tenhunen (Eds.), Kluwer Academic Publishers, Boston (2003) ("Jantsch")
1007	J. Dielissen, et al. " <i>Concepts and Implementation of the Philips Network-on-Chip</i> ," Philips Research Laboratories, Eindhoven, The Netherlands (2003) ("Dielissen")
1009	Duato et al., Interconnection Networks: IEEE Computer Society (1997)
1011	S. Hassoun et al., " <i>Optimal Buffered Routing Path Constructions for Single and Multiple Clock Domain Systems</i> ", ICCAD 2002 (2002) ("Hassoun")
1013	U.S. Patent No. 6,614,320 ("Sullam")