

Petitioner Qualcomm's Trial Hearing Demonstratives

QUALCOMM INCORPORATED,
Petitioner

v.

NETWORK SYSTEM TECHNOLOGIES, LLC
Patent Owner

IPR2024-00338
U.S. Patent No. 7,373,449

March 20, 2025

Prosecution

1. (Currently Amended) Integrated circuit comprising a plurality of processing modules (M; I; S; T), and a network (N; RN), and at least one resource manager (RM) for managing resources of the network (N), wherein the modules, network and resource manager are disposed on a chip and arranged for providing at least one connection between a first and at least one second module, wherein the at least one connection comprises a set of communication channels each having a set of connection properties, the connection properties of the different communication channels of said connection being adjustable independently, wherein said connection supports transactions comprising at least one of outgoing messages from the first module to the second module and return messages from the second module to the first module, wherein the resource manager determines whether the at least one connection with said communication channels is available based on the connection properties being available.

Ex-1002, 21

Allowable Subject Matter

1. Claims 1-18 are allowable over the prior art of records.
2. The following is an Examiner's statement of reasons for the indication of allowable subject matter: Claims 1, 7, are allowable over the prior art of record because the prior arts, cited in its entirety, or in combination, do not teach wherein the **resource manager** determines whether the at least one connection with the communication channels is available based on the connection properties being available.

The remaining claims, not specifically mentioned, are allowed for the same rationale as set for their dependent claims.

Ex-1002, 15

Prior publications by inventors (not disclosed by applicant)

Ground 2: Goossens2003

Chapter 4

GUARANTEEING THE QUALITY OF SERVICES IN NETWORKS ON CHIP

Kees Goossens, John Dielissen, Jef van Meerbergen, Peter Poplavko[†], Andrei Rădulescu, Edwin Rijpkema, Erwin Waterlander, and Paul Wielage

Philips Research Laboratories, Eindhoven, The Netherlands

[†] *Technical University of Eindhoven, Eindhoven, The Netherlands*

Kees.Goossens@philips.com

Abstract

Users expect a *predictable quality of service* (QoS) of embedded systems, even for future, more dynamic, applications. System-on-chip designers use networks on chip (NOC) to solve deep submicron problems, and to divide global problems into local, decoupled problems. NOCs provide services through protocol stacks, and introducing *guaranteed* services enables IP re-use and platform-based design. It also provides globally predictable behaviour, as required by the user, when combining local, decoupled solutions. There are several levels of QoS commitment (correctness, completion, completion bounds), with increasing cost. A combination of guaranteed and best-effort (no commitment) services combines their respective attractive features: predictable behaviour, and good average resource utilisation. The *ÆTHEREAL* NOC is an example of this approach, and forms the basis of a QoS-based design style, as advocated in this chapter.

1. Future applications and systems on chip

In this section we raise the following question: *why and how must systems implementing future applications guarantee the quality of service* (QoS)? We look at each of the components in turn, in the context of embedded systems and systems on a chip (SOC). We then observe that new SOC architectures and design methods divide global problems into local ones, and rely on networks on chip (NOC) to compose local solutions. We answer the question, in Section 2, by presenting a synthesis of a QoS-based design style and networks on chip. Section 3 explains that QoS can be decomposed into several levels of

61

(Ex-1007)

Ground 1: Goossens2002

Networks on Silicon: Combining Best-Effort and Guaranteed Services

K. Goossens, J. van Meerbergen, A. Peeters, and P. Wielage

Philips Research Laboratories, Eindhoven, The Netherlands

{kees.goossens,jef.van.meerbergen,ad.peeters,paul.wielage}@philips.com

Abstract

We advocate a network on silicon (NOS) as a hardware architecture to implement communication between IP cores in future technologies, and as a software model in the form of a protocol stack to structure the programming of NOSs. We claim guaranteed services are essential. In the *ÆTHEREAL* NOS they pervade the NOS as a requirement for hardware design, and as foundation for software programming.

1. Introduction

It will soon be possible to manufacture systems on silicon (SOSs) containing 10M gates and megabytes of embedded software. The complexity of these systems will be overwhelming, and two fundamental questions arise. First, from a hardware view point, how can these SOSs be designed? What are the required hardware architecture concepts? Second, from a software (application) perspective, how can the SOS hardware be programmed? What software concepts are needed? Our tenet is that only a consistent approach to both questions will enable us to harness the opportunities offered by semiconductor technology.

We believe in the intellectual-property block (IP) re-use model. This means that a SOS contains hundreds of IPs that are bought or premade. Assuming their correctness, their communication is then the problem. We propose a *network on silicon* (NOS) to implement the communication between these IPs. A NOS is a hardware architecture together with a programming model.

Guaranteed services are the focus of this paper: they serve as the basis for the programming model, and as a requirement for the NOS hardware architecture. A *protocol stack* is a layered approach to offering different services to different clients, based on a common network. It structures communication complexity from the physical implementation up to the application in a number of layers. The OSI reference model [4], used in this paper, comprises seven layers. In Section 2 we motivate why a packet-switched router network is a suitable hardware architecture, using the lower three layers as a backdrop. The top four layers are independent of the hardware architecture and are discussed in Section 3. In that context we show why guaranteed services offer many advantages over *best-effort* services when programming SOSs.

2. Packet-switched Router Networks

In this section we consider how a SOS containing hundreds of IPs can be designed, especially regarding layout and timing closure. Consider that the communication patterns of the IPs are dynamic at two levels. First, multiple configurations exist, each corresponding to a required user functionality. Second, communication data rates are variable within a configuration (e.g. variable-bit-rate MPEG streams). Structuring the wires used for communication between IPs is the main problem. If dedicated point-to-point wires are used, IPs are connected to all possible communication partners. This leads to many global wires, with three problems. First, they must be dimensioned for worst-case conditions in terms of a) electrical parameters such as cross talk and wire spacing, and b) volume of traffic. Second, it causes wire congestion around each IP. Thin closely-spaced high-power wires solve congestion, but this conflicts with the requirements for global wires, which are fat and widely spaced. Third, layout and timing dependencies between all IPs make timing closure global, making it non-scalable. Worse, it is layout specific: it cannot be re-used over design iterations or different designs [2].

A router network based on packet switching helps on these accounts. Fundamentally, wires are shared, or made programmable, by introducing switches (routers). This is possible because dedicated wires are normally underutilised (as little as 10% [3]). The first benefit is that there are fewer programmable wires, which reduces wire congestion around IP blocks. Second, traffic of multiple IPs is combined over a single wire, which can be dimensioned for the average traffic over all blocks using that wire instead of worst-case traffic per IP. Third, inter-IP wires are now structured: either they are short to get on the network, or they are router to router. The network topology and router layout can both be fixed (e.g. a torus and mesh [3] or a fat tree and Maltese cross [5]), or only the network topology. This allows dedicated optimisations that are re-usable over SOSs. Finally, and most importantly, decoupling of computation (IPs) and communication (NOS) enables compositional layout and timing closure. IPs communicate only using the network so that their layout and timing closure are independent of other IPs.

Programmable wires also have drawbacks. First, different IPs will contend for the use of a wire, requiring arbitration or scheduling. The properties of a router network depend much on the kind of scheduling it uses. A schedule can be fixed for a configuration (static) or dynamic (within a configuration). Static scheduling must be worst case, while dynamic scheduling leads to dimensioning for average traffic volume. Scheduling can also be independent for each router

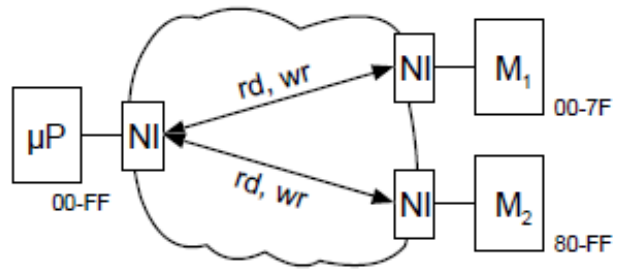
Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 \$17.00 © 2002 IEEE
Authorized licensed use limited to: Kirkland & Ellis LLP. Downloaded on April 30, 2023 at 19:08:01 UTC from IEEE Xplore. Restrictions apply.



(Ex-1005)

Prior publications by inventors (not disclosed by applicant)

Goossens2003 (Ex-1007)



4.1.3 **Connection Properties.** We distinguish the following services, or connection properties: 1) data integrity, 2) transaction order-

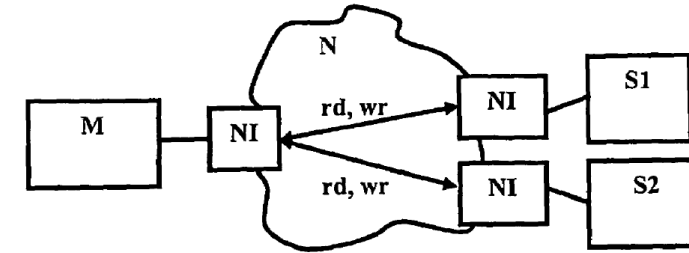
maximum latency. Connections must be *created* stating the requested service level; this is the negotiation phase. The NOC either accepts or rejects the request for the connection. Connection acceptance may lead to resource reservations in the NOC, e.g. buffers or a link bandwidth percentage. After usage,

or cars. Systems, especially SOCs, therefore have **limited resources**, and they must be shared and managed as the application unfolds its dynamic behaviour. Resource management depends on two phases: negotiation to obtain resources, followed by a steady state in which allocated resources are used. As applica-

more general connection property model. Not only are more properties considered, but the request and response communications can be independently configured (e.g. for flow control, and throughput guarantees), allowing more fine-grained resource management. Connection-based service provision there-

(Ex-1007, Fig. 4.5, pp. 72, 71, 65, 74)

'449 Patent (Ex-1001)



The invention is based on the idea to allow connection channels of a connection with different **connection properties**.

The connections according to the embodiments of the invention must be first created or established with the desired properties before being used. This may result in resource reservations inside the network (e.g., buffer space, or percentage of the link usage per time unit). If the

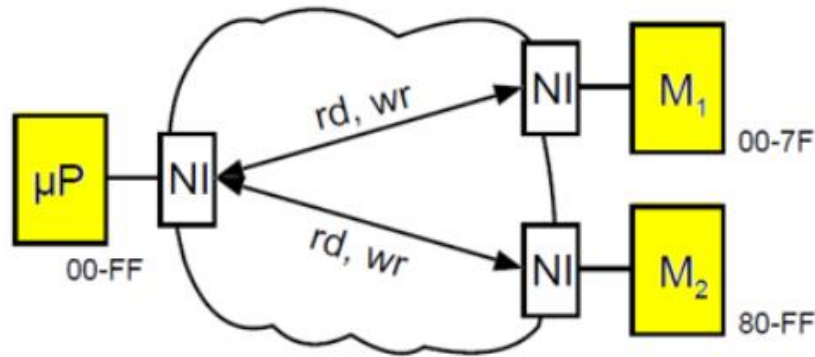
requested when asking for a connection setup. The **resource manager RM** allocates the necessary resources and enquires if such a connection based on these resources is possible.

To solve this problem, the connection properties of the request and response parts of a connection can be configured independently for all of throughput, latency and jitter. Con-

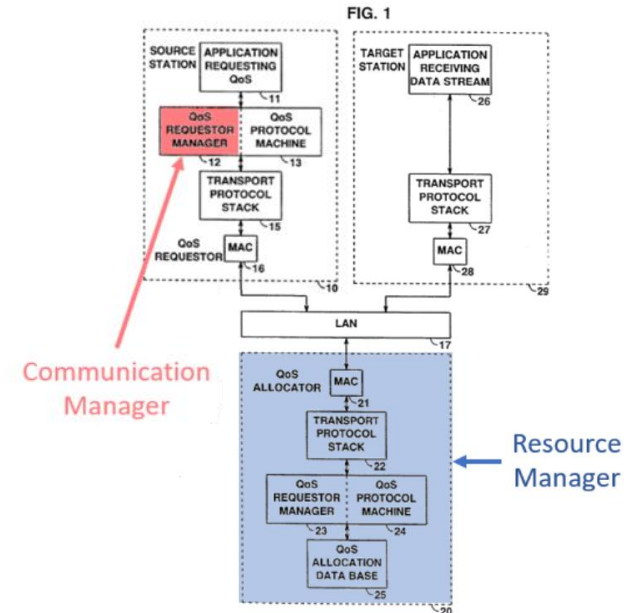
(Ex-1001, Fig. 1, 7:31-33, 10:62-66, 10:24-26, 11:20-22)

Goossens2003 + Drake

Goossens2003



Drake teachings (composite annotations)

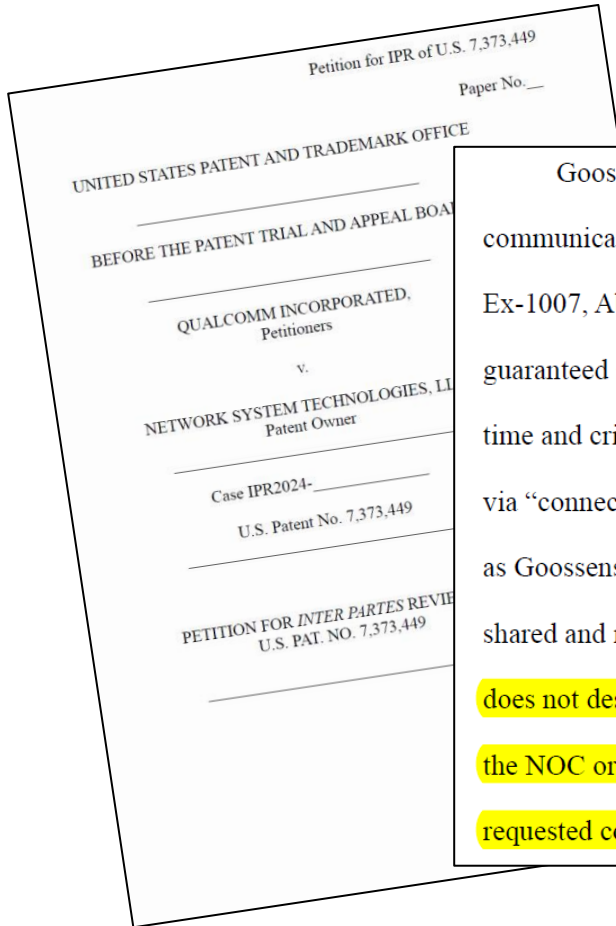


“A POSITA would have understood that using Drake’s QoS management process would improve the performance and efficiency of Goossens2003’s NOC by ensuring that connections established by IPs are able to provide the required services A POSITA would have understood that mature networking technologies like those described by Drake could be usefully applied in the context of newer networks on chip.”

Petition (-01081 IPR, Paper 1), 52-53

Goossens2003 + Drake

Petition (-01081 IPR, Paper 1), 51-52:

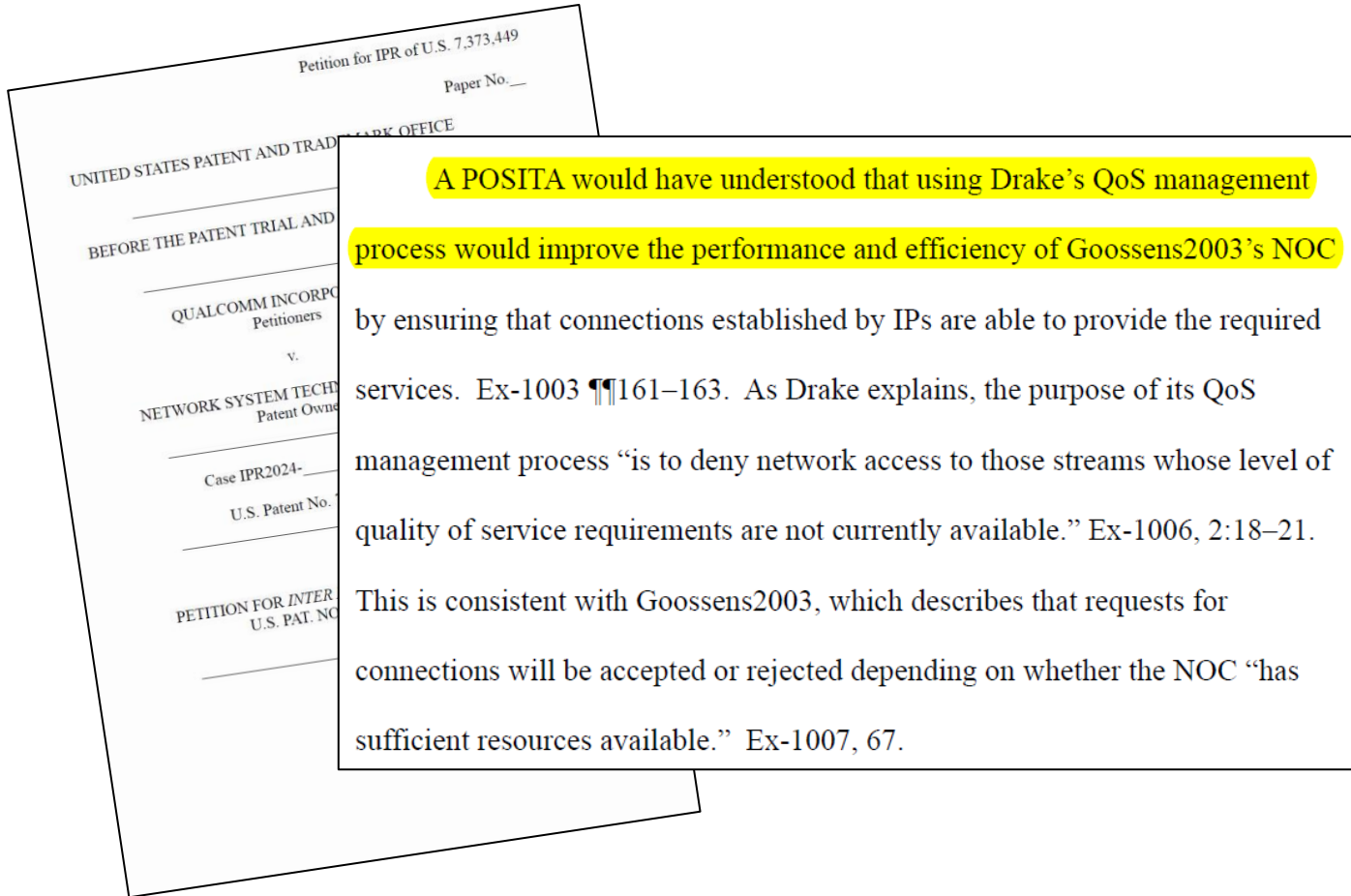


Goossens2003 describes a network on chip (NOC) for use in providing communications between “intellectual property blocks (IP)” in a system-on-chip. Ex-1007, Abstract, 63, 74. Goossens2003 further describes that this NOC “offers guaranteed services to obtain the advantages ... of QOS-based design ... for real-time and critical functions.” Ex-1007, 71. These guaranteed services are provided via “connections” which may require the reservation of NOC resources. *Id.* But, as Goossens2003 explains, “SOCs ... have limited resources, and they must be shared and managed.” Ex-1007, 65; *see also id.*, 67. Goossens2003, however, does not describe a component or mechanism for managing resource usage within the NOC or for determining the availability of the resources required for a requested connection. Ex-1003 ¶¶157–159.

Drake is directed to the same goal of managing network resources in order to “ensure[] selected levels of quality of services,” and a POSITA would have been motivated to combine Goossens2003’s disclosures with Drake’s disclosures regarding a system for QoS management to obtain the benefits of Drake’s known solution to ensuring the availability of network resources for guaranteed services. Ex-1003 ¶160. A POSITA would have looked to Drake’s disclosure of a “QoS requestor manager” to formulate requests for guaranteed services with particular properties and a “QoS allocator” to determine whether the network resources required for the requested services are available. Ex-1006, 4:17–56; *see also id.*, 2:14–40.

Goossens2003 + Drake

Petition(-01081 IPR, Paper 1), 52-53:



A POSITA would also have had a reasonable expectation of success in combining the disclosures of Goossens2003 and Drake. Ex-1003 ¶164. They would have understood that doing so merely combines prior art elements according to known methods to yield predictable results, as both discuss managing resources to provide guaranteed services. Drake provides additional details about such a system that is consistent with and would have worked with the system of Goossens2003. Indeed, Goossens2003 and Drake occupy the same field of data communication in packet-based networks, and both are directed to ensuring that sufficient resources are available for the communication of multimedia content, such as audio and video. See Ex-1006, 1:22-24 (describing "audio and full motion

Goossens2003 + Drake

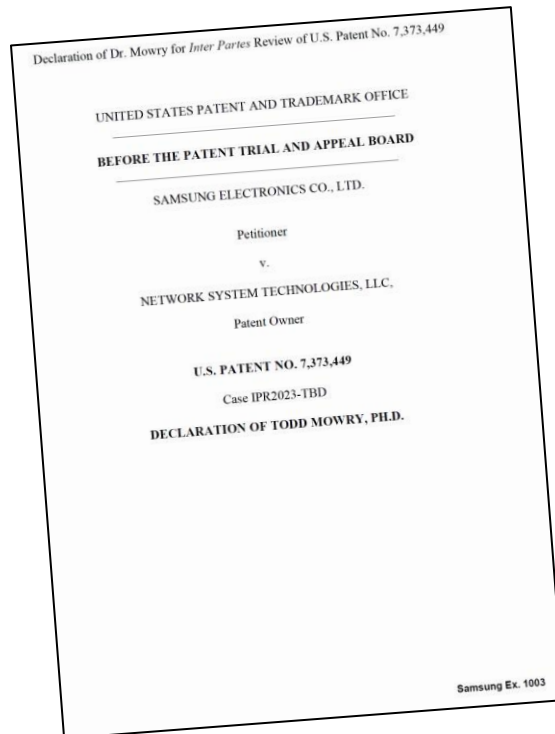
Petition (-01081 IPR, Paper 1), 53-54:



audio and video”). As such, a POSITA would have reasonably expected that use of Drake’s known solution to ensuring the availability of network resources for guaranteed services would improve Goossens2003’s NOC in the same way. Ex-1003 ¶164. A POSITA would also have understood that Drake’s disclosure is applicable to networks on chip, including because certain resources are more limited in the on-chip context and need to be managed carefully. Ex-1007, 65 (“SOCs ... have limited resources, and they must be shared and managed.”). Because Drake’s QoS management process would more efficiently utilize the resources of the network and minimize their waste, it is well-suited for such a resource-constrained environment. Ex-1003 ¶¶165–166. A POSITA would have understood that mature networking technologies like those described by Drake could be usefully applied in the context of newer networks on chip. *Id.* Indeed,

Technology and Motivations

Dr. Mowry testimony (Ex-1003):

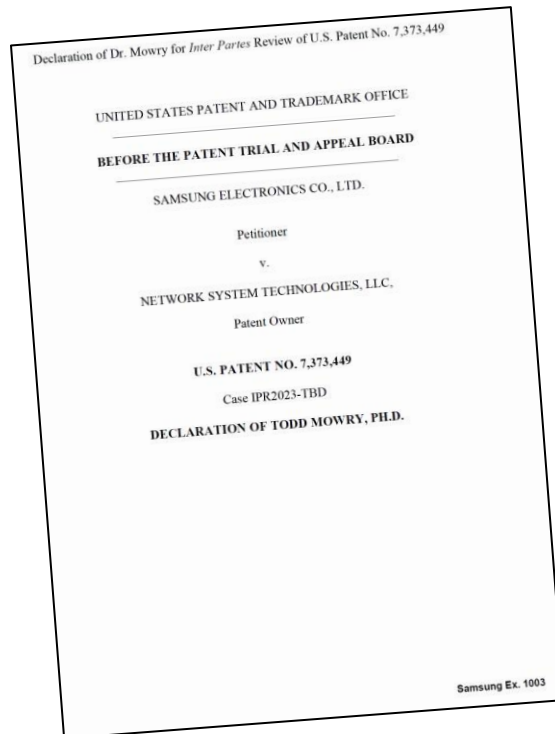


156. As a threshold matter, a POSITA would have been motivated to combine the disclosures of these two references because both relate to the use of “connections” to provide guaranteed services, and Drake provides additional implementation details that would aid a POSITA in creating an SoC with improved performance and efficiency.

159. However, Goossens2003 does not include certain details regarding the implementation of its guaranteed services and connections. For example, the paper explains that requests for connections will be granted if there are sufficient resources available, and that they will otherwise be rejected, but does not describe a component that makes this determination or the process by which it does so. Ex-1007 (Goossens2003) at 67. Goossens2003 also describes that the establishment of connections to provide guaranteed services may require the reservation of network resources, which must be carefully managed given the limited resources in SoCs. Ex-1007 (Goossens2003) at 65, 67. Goossens2003, however, does not describe a component or mechanism for managing resource usage within the NOC or for determining the availability of the resources required for a requested connection.

Technology and Motivations

Dr. Mowry testimony (Ex-1003):

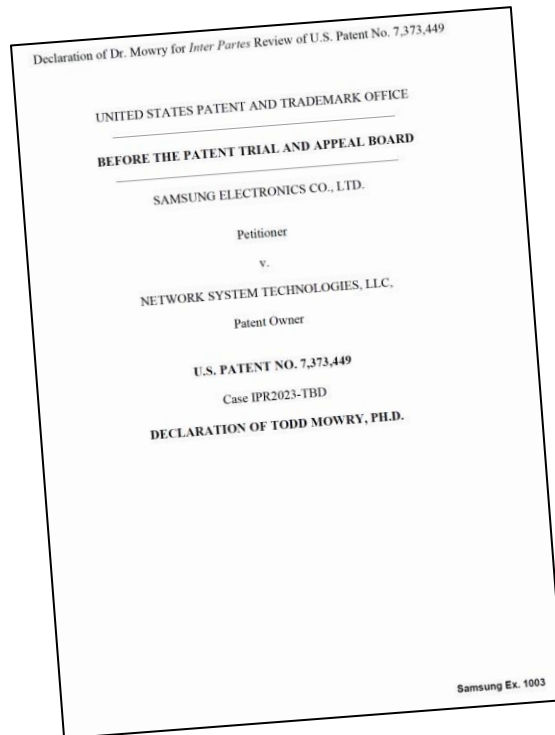


160. As I explained in Ground I, Drake provides details regarding the implementation of these features and a POSITA would have found Drake's known solution for "ensur[ing] selected levels of quality of services" helpful in implementing an SoC like the one described by Goossens2003. See Ex-1006 (Drake) at Abstract. A POSITA would have been particularly interested in Drake's disclosure of a "QoS requestor manager" to formulate requests for guaranteed services with particular properties and a "QoS allocator" to determine whether the network resources required for the requested services are available. See Ex-1006 (Drake) at 4:17–56; see also Ex-1006 (Drake) at 2:14–40.

163. In my opinion, a POSITA would have understood that using Drake's quality of service management process would improve the efficiency and performance of Goossens2003's network on chip by ensuring that the network resources required to provide guaranteed services are actually available, and that the network will be able to fulfill its commitments to the IPs that have been granted such services. Indeed, Goossens2003 describes the same goal, explaining that "The NOC that offers communication services will honour the request if has sufficient resources available, and will reject it otherwise." Ex-1007 (Goossens2003) at 67.

Technology and Motivations

Dr. Mowry testimony (Ex-1003):



166. Furthermore, the parallels between mature off-chip networks and newer on-chip networks would have been well-known to POSITA. As I explained above, NoCs have borrowed many design features from scalable interconnection networks that were designed for large-scale parallel machines in the 1990s. For example, as described in one text devoted to networks-on-chip “[d]esigning a network for on-chip communication is similar to the problem of designing a local area computer network with some computing and communication requirements.” Ex-1014 (Jantsch) at 102. Another chapter in the same text explained that addressing issues related to network resource allocation—as Goossens2002 and Drake do—is substantially similar in on-chip and off-chip networks: “The resource allocation, optimization of network usage and verification of performance and correctness are the main problems that are basically similar to what distributed and parallel system designers have to face.” Ex-1014 (Jantsch) at 33. Indeed, the ’449 patent itself acknowledges that many of the issues faced by networks on chip had been studied in the context of off-chip networks and that results of those studies are applicable to on-chip networks as well. Ex-1001 (’449 patent) at 1:50–66.

Technology and Motivations

Dr. Mowry testimony
(Ex-1003):

Additional evidence corroborates:

Declaration of Dr. Mowry for *Inter Parties* Review of U.S. Patent No. 7,373,449

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS CO., LTD.

Petitioner

v.

NETWORK SYSTEM TECHNOLOGIES, LLC,

Patent Owner

U.S. PATENT NO. 7,373,449

Case IPR2023-TBD

DECLARATION OF TODD MOWRY, PH.D.

chip

eded

signs

some

ators

es or

are

chip

essor

ing

oC

Wolf

as of

n the

rious

ntire

con.

on");

ngle

and

at

ch

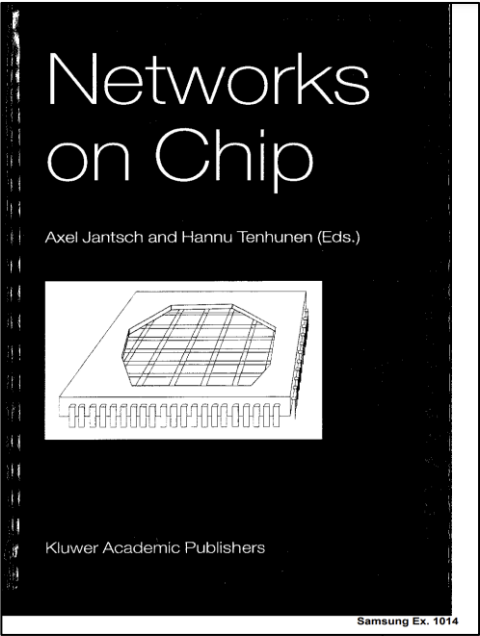
ks)

17

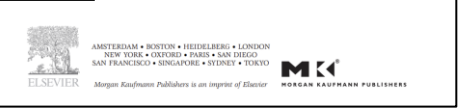
Samsung Ex. 1003

18

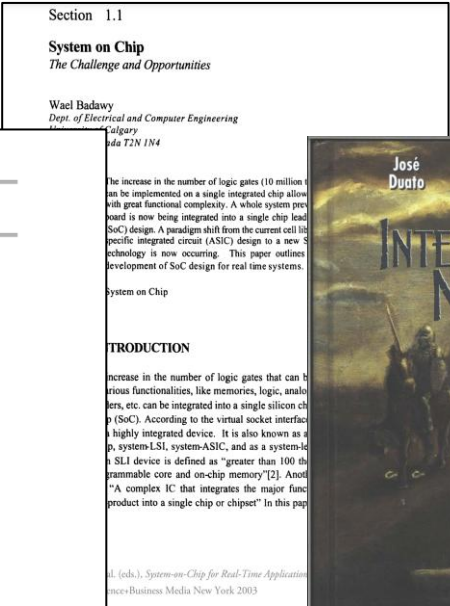
Samsung Ex. 1003



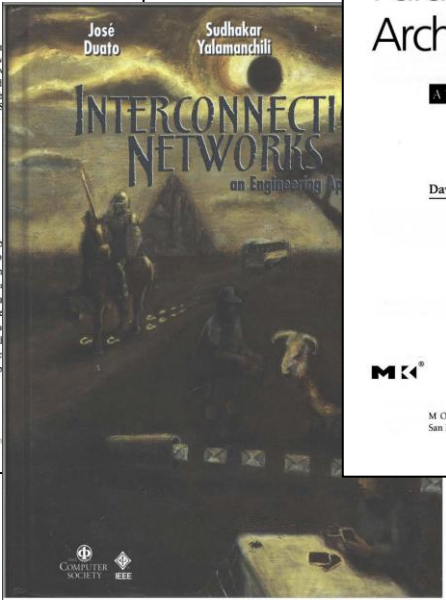
(Ex-1014)



(Ex-1008)



(Ex-1009)



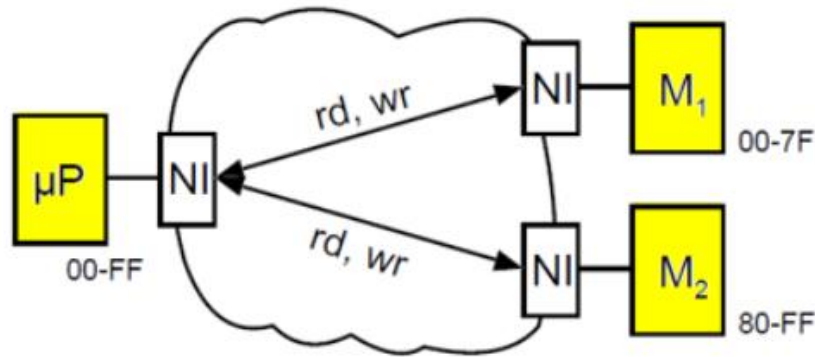
(Ex-1013)



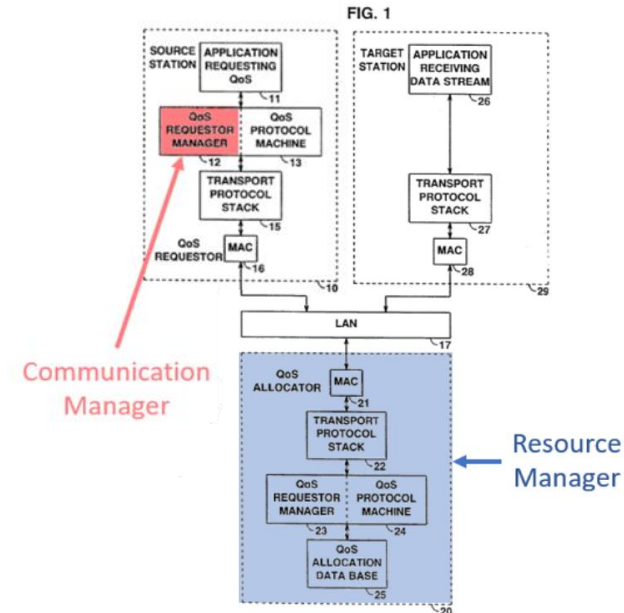
(Ex-1015)

Goossens2003 + Drake

Goossens2003



Drake teachings (composite annotations)



“A POSITA would have understood that using Drake’s QoS management process would improve the performance and efficiency of Goossens2003’s NOC by ensuring that connections established by IPs are able to provide the required services A POSITA would have understood that mature networking technologies like those described by Drake could be usefully applied in the context of newer networks on chip.”

Petition (-01081 IPR, Paper 1), 52-53

Goossens2003

Claim 1[a]: “a plurality of processing modules (M; I; S; T)”

b. Claim 1[a]

Goossens2003 discloses this limitation. Ex-1003 ¶170. Goossens2003 describes that its SoC includes multiple “intellectual property blocks (IP),” which it also refers to as “masters” and “slaves.” Ex-1007, 63; *see also id.*, 61, 67–69 (describing “combining IPs to obtain a SOC”), 71–74 (“communication[s] between one master and one or more slaves”), 77–78. A POSITA would have understood that these IPs constitute processing modules. Ex-1003 ¶170. Figure 4.5 of Goossens2003, annotated below, depicts three such IPs, including one labeled “μP”—microprocessor—and two labeled “M,” for “memory module.”

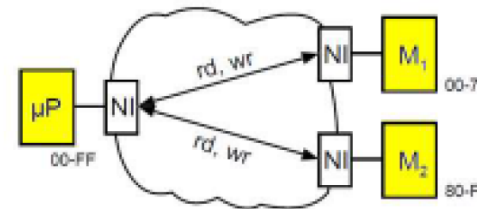


Figure 4.5. A narrowcast connection.

Petition (-01081 IPR, Paper 1), 55

Claim 1[c]: “at least one resource manager (RM) for managing resources of the network (N)”

Claim 1[g]: “wherein the resource manager determines whether the at least one connection with said communication channels is available based on the connection properties being available”

d. Claim 1[c]

Goossens2003 in view of Drake renders this limitation obvious. Ex-1003

¶173. As discussed above, Goossens2003 explains the implementation of connections in a NOC requires “resource management,” but does not describe a specific component or mechanism for tracking resource usage within the NOC or for determining the availability of the resources required for a requested connection. See VII.B.1; *see also* Ex-1007, 65, 68–69, 74. As discussed in Ground I, however, Drake discloses a “resource manager” in the form of “a quality of service (QoS) allocator.” *See* VII.A.2.d.

h. Claim 1[g]

Goossens2003 in view of Drake renders this limitation obvious. Ex-1003

¶182. Goossens2003 describes that when requests for connections are issued by IPs, its NOC “will honor the request if has sufficient resources available, and will reject it otherwise.” Ex-1007, 67. But Goossens2003 does not describe the component of its NOC that determines whether sufficient resources are available, nor the process by which it does so.

Claim 2: “...at least one communication manager (CM) for managing the communication between different modules”

3. Claim 2

Claim 1 is obvious. *See* VII.B.2. Goossens2003 in view of Drake renders this limitation obvious. Ex-1003 ¶184. Goossens2003 describes that its system includes “network interfaces,” and explains that “end-to-end” connection properties – those properties which relate to the interactions between the master and slave modules, including “reordering, transaction completion, and flow control”—“are implemented by the network interfaces.” Ex-1007, 76–77; Ex-1003 ¶185. Goossens2003 further describes that “IPs negotiate with network interfaces to obtain connections with certain properties.” Ex-1007, 77. This is just like the interaction between the modules and communication manager described in the ’449 Patent. Ex-1001, 10:3–54. Thus, a POSITA would have understood that Goossens2003’s network interfaces are circuits that manage the communication between different modules and disclose a communication manager under PO’s proposed construction. Ex-1019, 12; Ex-1003 ¶186.

Additionally, as discussed in Ground I, Drake discloses a “communication manager” in the form of a “QoS requestor manager.” *See* VII.A.3. For the reasons

Petition (-01081 IPR, Paper 1), 62-63

Claim 6[a]: “wherein said first module (M; I) is adapted to issue a request (REQ) for a connection with at least one of said second modules to said communication manager (CM)”

b. Claim 6[a]

Goossens2003 in view of Drake renders this limitation obvious. Ex-1003

¶193. Goossens2003 describes that “IPs negotiate with network interfaces to obtain connections with certain properties.” Ex-1007, 77. Thus, a POSITA would have understood that Goossens2003’s master IPs (*i.e.*, first module) issue requests for connections to the network interfaces (*i.e.*, communication managers). Ex-1003 ¶193. As explained above, the requested connections are between a “master” (*i.e.*, first module) and “one or more slaves” (*i.e.*, second modules). Ex-1007, 71.

Petition (-01081 IPR, Paper 1), 65

Goossens2002 (similar to Goossens2003)

Claim 1[c]: “at least one resource manager (RM) for managing resources of the network (N)”

Claim 1[g]: “wherein the resource manager determines whether the at least one connection with said communication channels is available based on the connection properties being available”

d. Claim 1[c]: “at least one resource manager (RM) for managing resources of the network (N)”

This limitation is disclosed by Goossens2002 and/or rendered obvious by Goossens2002 in view of Drake. Ex-1003 ¶90. Goossens2002 describes that its system on silicon includes a “quality-of-service manager” (*i.e.*, resource manager) that can “observe and steer” the NOS (Ex-1005, 3), and a POSITA would have recognized that it is a circuit and that its functionality involves managing resources of the network. Ex-1003 ¶90. Thus, Goossens2002 discloses a resource manager,

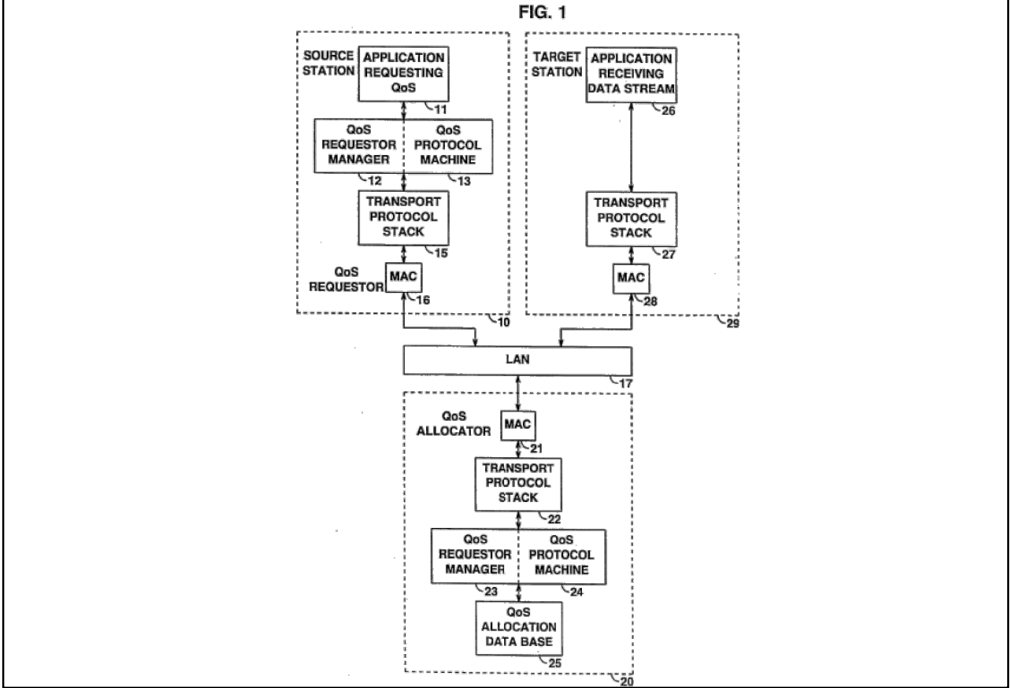
h. Claim 1[g]: “wherein the resource manager determines whether the at least one connection with said communication channels is available based on the connection properties being available”

Goossens2002 in view of Drake renders this limitation obvious.⁵ Ex-1003 ¶113. As discussed regarding element 1[c], Goossens2002 in view of Drake discloses a “resource manager” in the form of quality-of-service manager/QoS allocator that receives requests for connections with certain QoS properties. *See* VII.A.2.d. As shown above, Drake’s QoS allocator examines the network

Petition (-01081 IPR, Paper 1), 23-24, 32-33

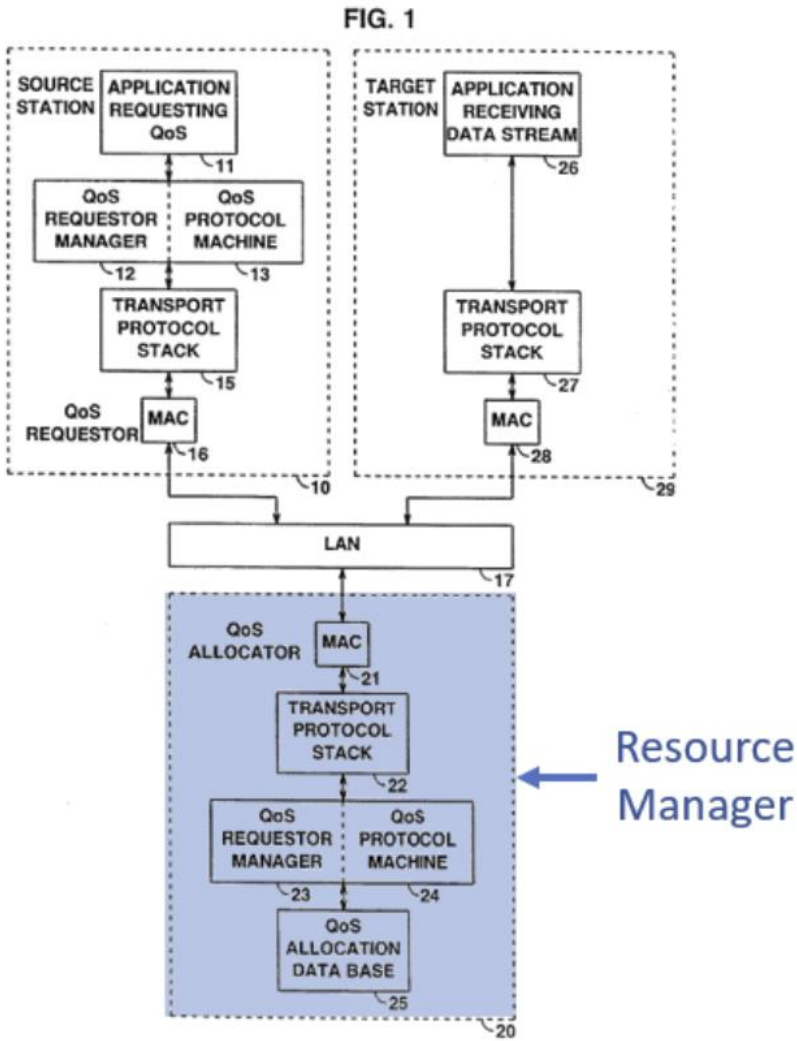
Drake's QoS teachings

67. To address this need, Drake describes a quality of service management system that receives requests for “QoS connections” with specified properties, determines whether the required network resources are available, and reserves those resources for the requesting station. Ex-1006 (Drake) at Abstract. An exemplary embodiment of Drake’s system is depicted in Figure 1 below.



Ex-1003 (Dr. Mowry Decl.), ¶167

Drake’s QoS teachings – Resource Manager



91. Additionally, Drake similarly discloses a “resource manager” in the form of a “quality of service (QoS) allocator.” Ex-1006 (Drake) at 4:22–25; *see also* Ex-1006 (Drake) at 4:66–5:14, Fig. 1. Drake further describes the QoS allocator as managing resources of the network. Specifically, the QoS allocator receives

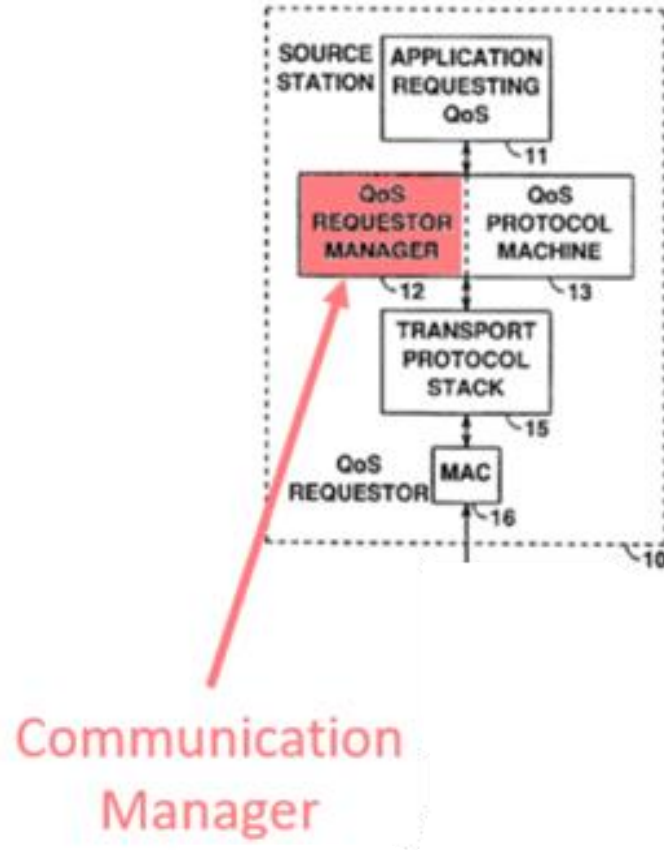
[. . .]

92. Indeed, Drake’s QoS allocator provides the same functionality as the “resource manager” that is described in the specification of the ’449 patent. Specifically, like Drake, the ’449 patent describes that its resource manager receives requests for connections with certain properties, “enquires if such a connection based on these resources is possible,” and “reserves the available connection i.e. the required resources.” Ex-1001 (’449 patent) at 10:19–57. Additionally, the “quality-

[. . .]

Ex-1003 (Dr. Mowry Decl.), ¶¶91-94, 173

Drake's QoS teachings – Communication Manager

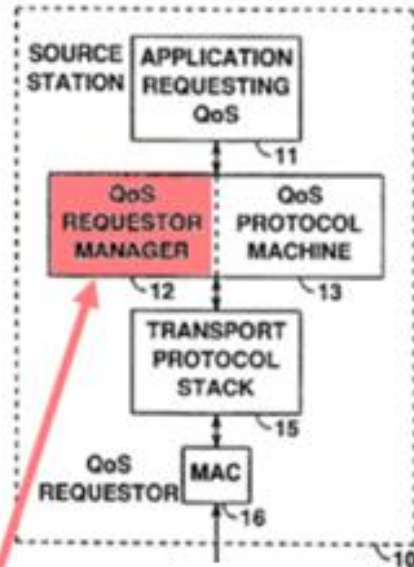


118. For example, Drake discloses a “communication manager” in the form of a “QoS requestor manager.” Ex-1006 (Drake) at 4:32–37; *see also* Ex-1006 (Drake) at 4:62–65, 5:5–8, Fig. 1. Drake explains there are QoS requestor managers that are associated with source stations, and that the QoS requestor managers manage communications between modules by “assembl[ing] a request to reserve a QoS connection for a multi-media data stream ... between source station 10 and target station 29,” which request is sent to the QoS allocator (*i.e.*, resource manager). *Id.*

119. Indeed, this is consistent with the disclosure of the communication manager described in the specification of the '449 patent, which explains that “[t]he communication manager CM request[s] a connection with a set of properties between two modules from the resource manager RM.” Ex-1001 ('449 patent) at 10:19–24.

Ex-1003 (Dr. Mowry Decl.), ¶¶118-119;
see also id. ¶¶ 120-122, 130, 187, 195

Drake's QoS teachings – Communication Manager

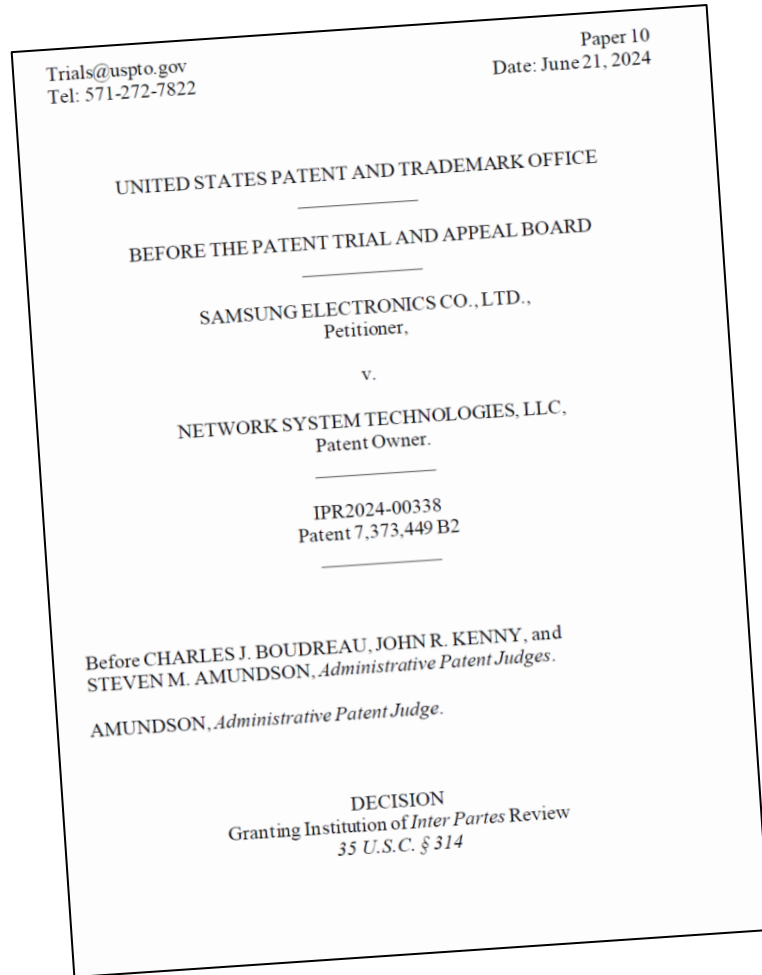


Communication
Manager

120. Additionally, in my opinion, to the extent Patent Owner argues that the claimed communication manager must have the ability to request that the resource manager release reserved connections and resources (*see* Ex-1006 (Drake) at 10:51–54), that too is disclosed by Drake. *See, e.g.*, Ex-1006 (Drake) at 8:65–9:1 (describing “QoS de-allocation announcement” message), 11:49–12:34 (describing a process for “de-allocating a previously allocated QoS data stream”). Moreover, in my opinion, it would have been obvious to a POSITA that the QoS requestor managers could either accept or reject connections established by the QoS allocator based on the specified QoS parameters.

Ex-1003 (Dr. Mowry Decl.), ¶120;
see also id. ¶¶ 117-122, 130, 187, 195

Institution Decision



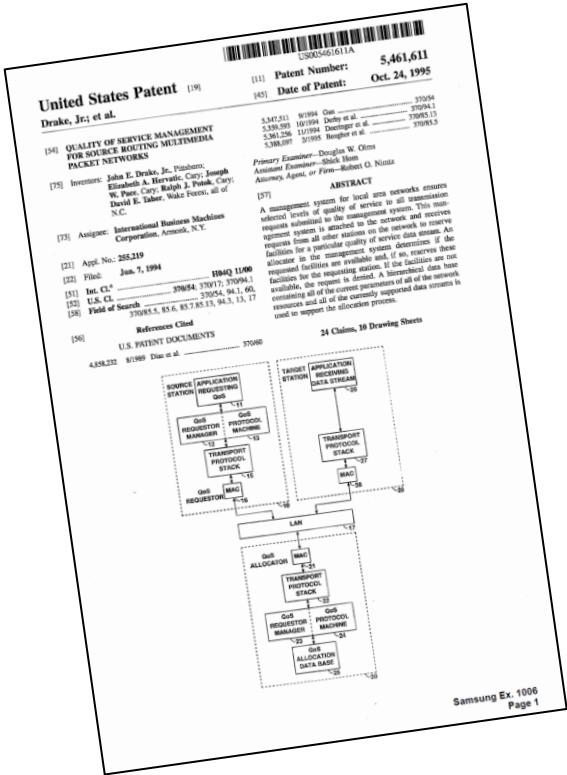
Moreover, Patent Owner mischaracterizes the proposed modification when contending that modifying Goossens2003's IP block to include Drake's requestor manager 12 conflicts with Goossens2003's teaching to decouple "computation from communication." *See* Pet. 48–51; Prelim. Resp. 62–63. Goossens2003 describes the *Æ*thereal NoC for SoCs that integrates "heterogeneous" intellectual property blocks (IPs). Ex. 1007, 61, 66, 71–74; *see* Ex. 1003 ¶¶ 69–70. Petitioner proposes implementing Drake's QoS management process in the NoC, not modifying an IP block. *See* Pet. 49–50; Ex. 1003 ¶¶ 162–163, 173. Consistent with this, Drake explains that the QoS management of "a local area network is almost entirely concentrated in a single QoS management station on the LAN." Ex. 1006, 2:66–3:1.

Institution Decision (-00338 IPR, Paper 10), 85 (Ground 2, Goossens2003); *see also id.* 66 (Ground 1, Goossens2002)

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

Drake:



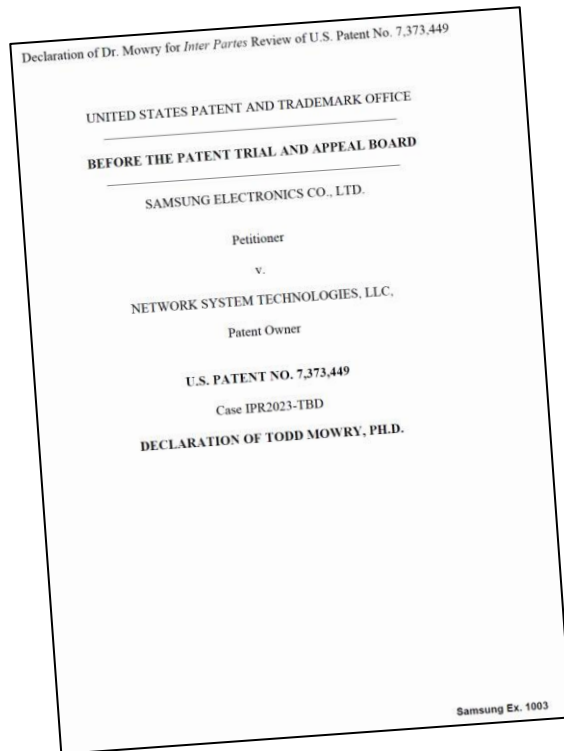
More particularly, station 10 is the source of a desired multimedia transmission and comprises an application 11 which requests a QoS connection to the destination or target station 29. Application 11 also is responsible for causing the

Ex-1006, 4:26-30

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

Dr. Mowry testimony:



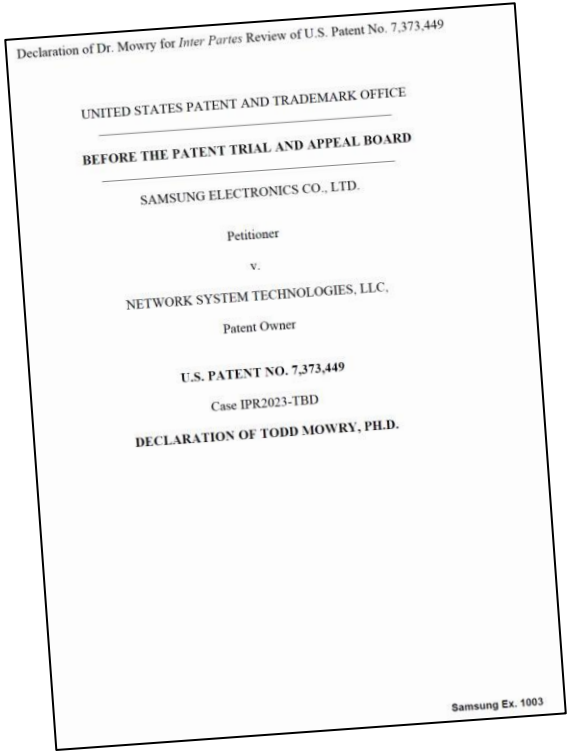
129. Similarly, Drake describes that an application in a source station (*i.e.*, first module) “requests a QoS connection to the destination or target station 29” (*i.e.*, second module). Ex-1006 (Drake) at 4:26–29; *see also* Ex-1006 (Drake) at Abstract, 5:25–50, 2:32–41 (describing “requested QoS values”). In my opinion, a POSITA

Ex-1003, ¶129

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

Dr. Mowry testimony:



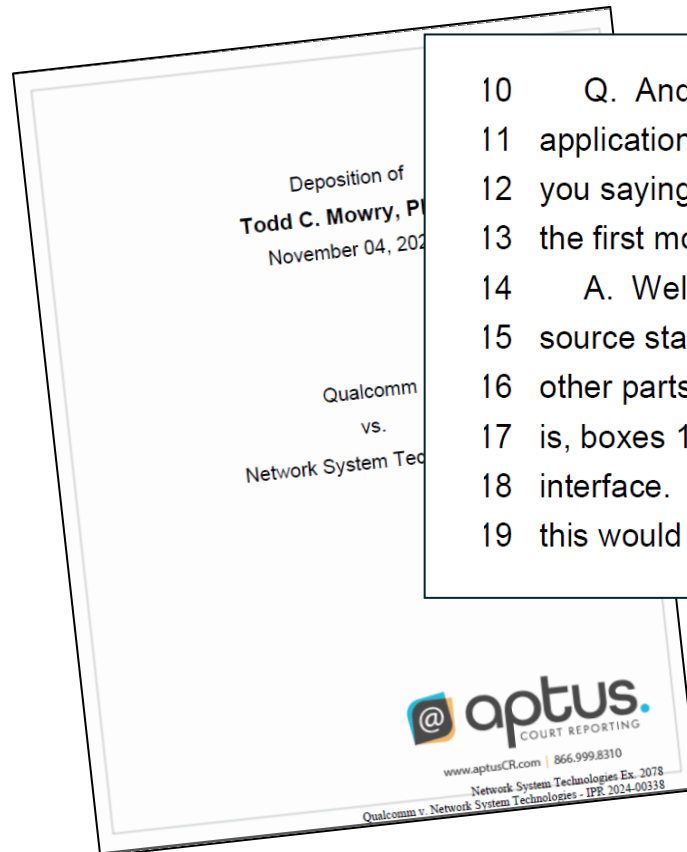
130. Additionally, Drake also describes that the application in the source station (*i.e.*, first module) communicates this request to a QoS requestor manager (*i.e.*, communication manager) for subsequent transmission to a QoS allocator, as discussed further below. *See* Ex-1006 (Drake) at 3:21–28, 4:32–37, 4:62–65 (“The QoS requestor manager 12 interacts with application using internal messages as is well known in the art”), 5:5–8, Fig. 1. This is indicated by the arrows shown in Drake’s Figure 1 annotated below:

Ex-1003, ¶130

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

Dr. Mowry testimony (first deposition):



10 Q. And my question is, are you saying that the
11 application 11 corresponds to the first module or are
12 you saying that the entire source station corresponds to
13 the first module?

14 A. Well, I'm saying that the application in a
15 source station -- so the application is box 11. The
16 other parts of -- the other things within box 10 -- that
17 is, boxes 12, 13, 15, and 16 -- these are the network
18 interface. So, for example, in figures 1, 2, and 3,
19 this would be the NI or ANIP.

20 Q. All right. Why don't about we go back to Drake
21 figure 1.

22 A. Okay.

23 Q. Can you tell me which of the components in
24 figure 1 of Drake correspond to the claimed first
25 module?

1 A. Well, in this scenario it's box 11, which is
2 inside the source station.

[...]

8 A. When the communication occurs, it's -- from the
9 LAN's perspective the communication is between boxes 10
10 and 29. But it is really between box 11 and box 26.

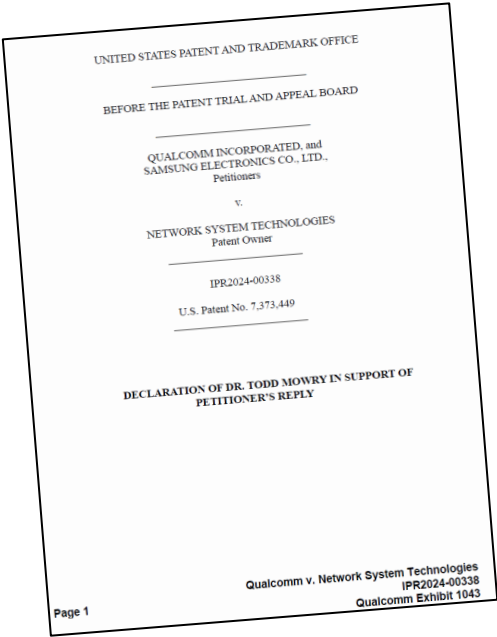
11 But to answer your question, which I -- which
12 is the same answer I said -- gave before, which is it's
13 box 11.

Ex-2078, 174:10-175:13

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

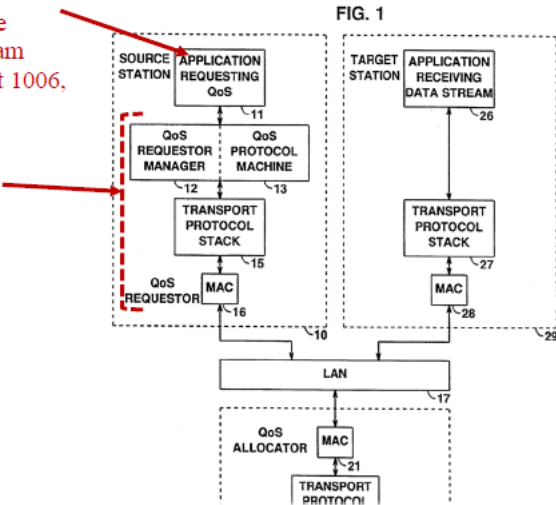
Dr. Mowry testimony (second declaration):



15. Drake does sometimes refer to the service request or response being sent from or to “source station 10.” This makes sense, because Drake shows that application 11 (which is the component making the original request and receiving the response) is in source station 10. For example, a POSITA would understand source station 10 could be something like a server. It is not wrong to say that in that context a service request comes from the server, but it actually originates from an application running on the server. A POSITA would understand the application running on the server to be a component (such as the main processor) that is separate from the interfacing “components” that are also part of the server. This is just like what Drake shows, where application 11 and the network interface “components” (such as QoS requestor manager 12) are shown and described as separate things. *See* Exhibit 1006, Fig. 1 and 4:26-36.

Application 11 “requests a QoS connection,” and also transmits the multimedia data stream (for example, Exhibit 1006, 4:27-32, 4:48-50)

Network interface “components” (for example, Exhibit 1006, 4:34-35)

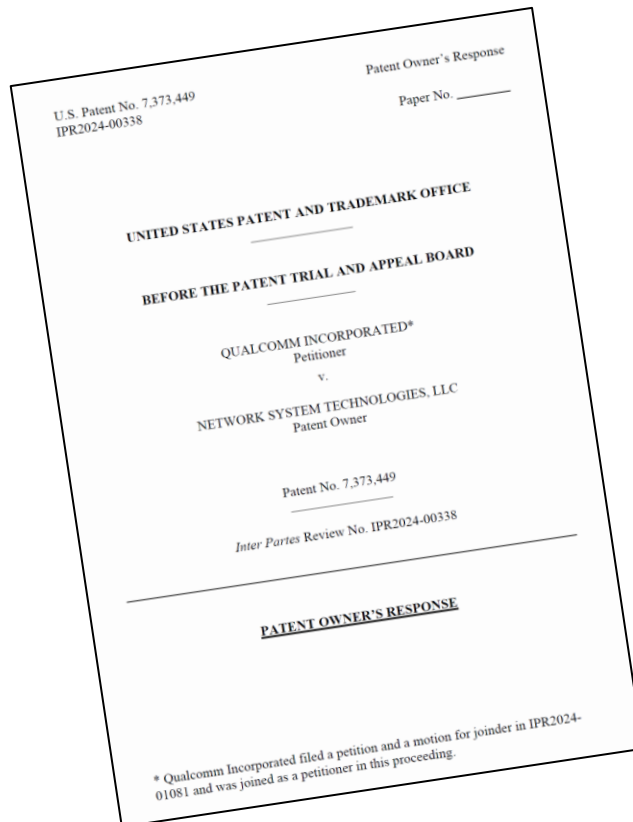


Ex-1043, ¶15

Drake and the claimed “first module”

Application 11 in Drake’s source station 10 requests a QoS connection

PO Response (Paper 35), 18-19:



Dr. Mowry later clarified that *only the application 11* within source station 10 corresponds to the claimed “first module”:

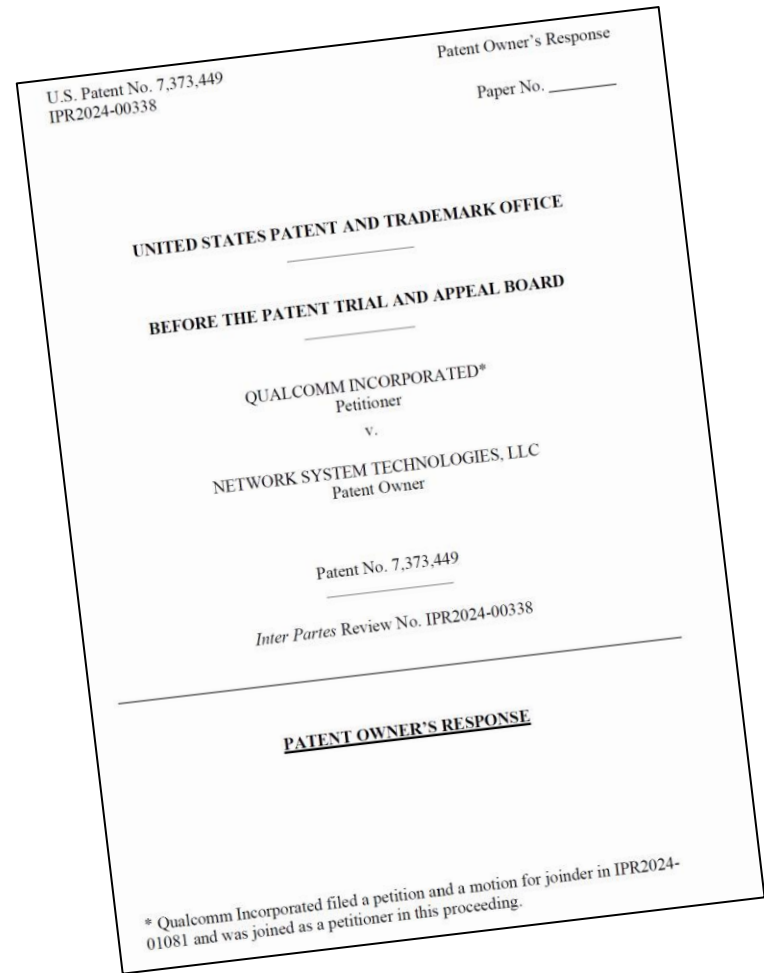
Q. Can you tell me which of the components in figure 1 of Drake correspond to the claimed first module?

A. Well, in this scenario it’s box 11, which is inside the source station.

EX2078, 174:23-175:19; *see also id.*, 174:10-19 (“I’m saying that the application in a source station – so the application is box 11. The ... other things within box 10 ... are the network interface”), 175:21-176:2 (“box 11 is the first module”).

Drake and the claimed “first module”

PO’s new post-institution claim construction:



U.S. Patent No. 7,373,449
IPR2024-00338

Patent Owner’s Response

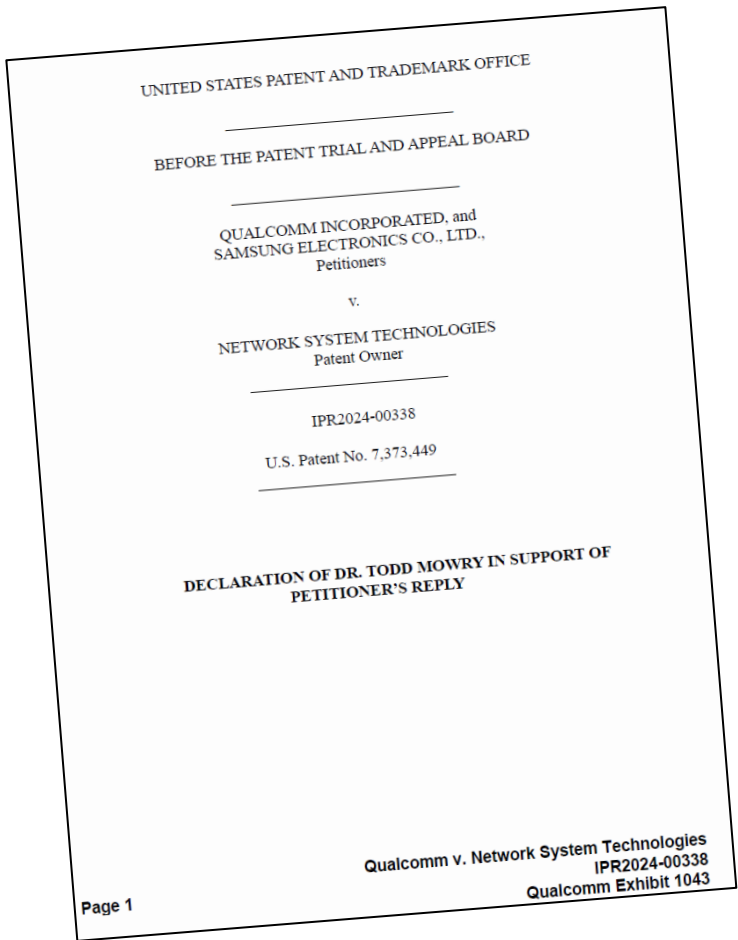
‘first module.’” Paper 14, 25. For the reasons discussed below, this construction is inconsistent with intrinsic evidence and the understanding of a POSITA. The claimed first module and communication manager (“CM”) should be construed as separate and distinct claim elements. EX2070, ¶114.

[. . .]

PO Response (Paper 35), 13

Drake and the claimed “first module”

Dr. Mowry testimony regarding new claim construction:

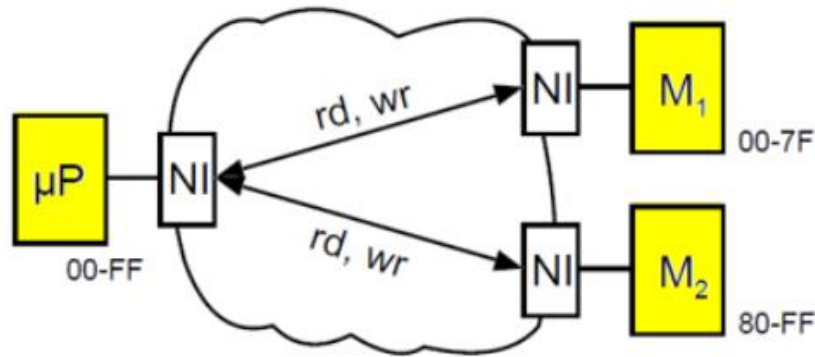


7. I have not been asked to provide an opinion on this claim construction issue. I also note that it would not affect my opinions. As I stated in my First Declaration and also explain below, the claims are met by a combination of Goossens2002 with Drake’s QoS teachings, and also Goossens2003 with Drake’s QoS teachings. In each of the combinations, the QoS components (including the communication manager) are not positioned in the first module but rather where the Goossens references already teach to put QoS functionality: outside of the modules, with the NOC. As I explain further below, I also point to “first module” and “communication manager” components in the prior art that are separate and distinct.

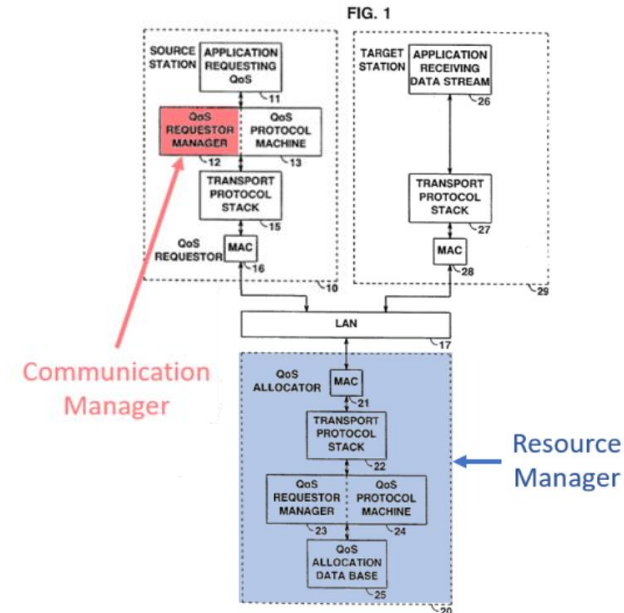
Ex-1043, ¶7

Goossens2003 + Drake

Goossens2003



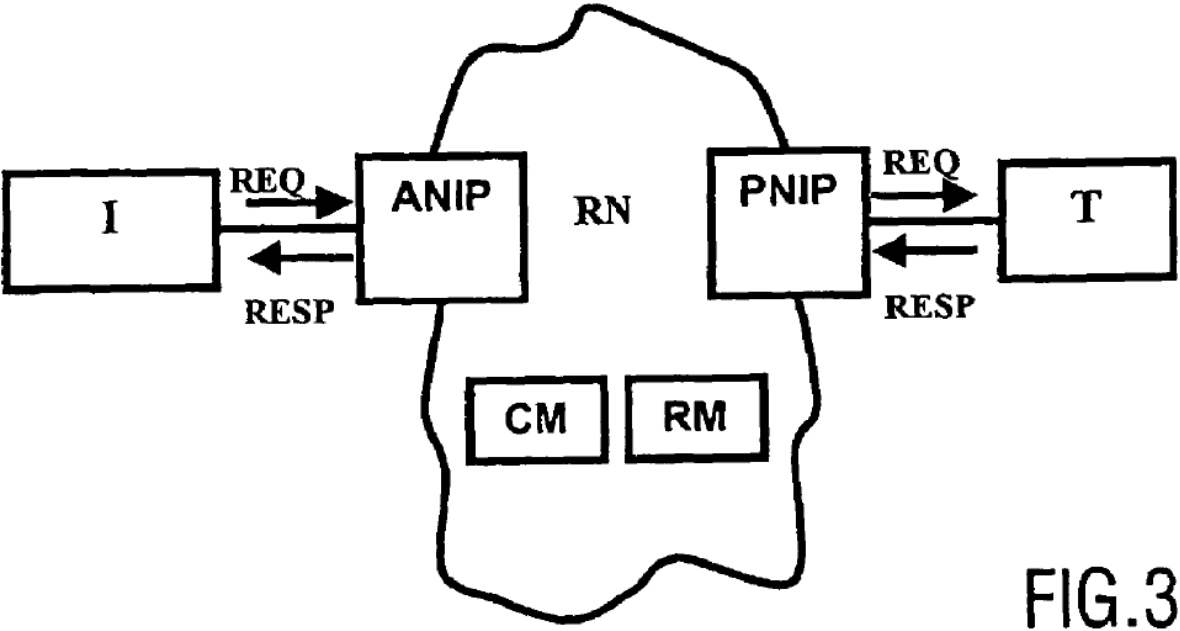
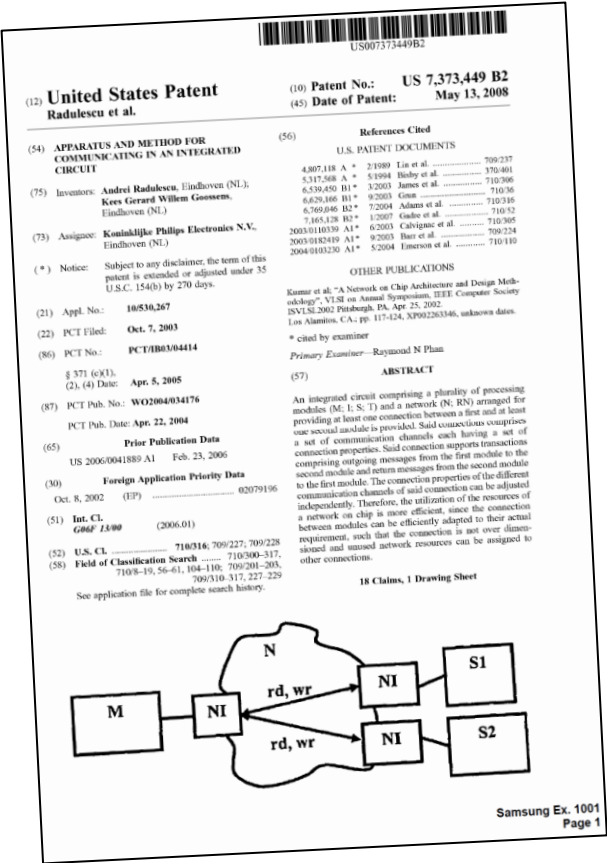
Drake teachings (composite annotations)



“A POSITA would have understood that using Drake’s QoS management process would improve the performance and efficiency of Goossens2003’s NOC by ensuring that connections established by IPs are able to provide the required services A POSITA would have understood that mature networking technologies like those described by Drake could be usefully applied in the context of newer networks on chip.”

Petition (-01081 IPR, Paper 1), 52-53

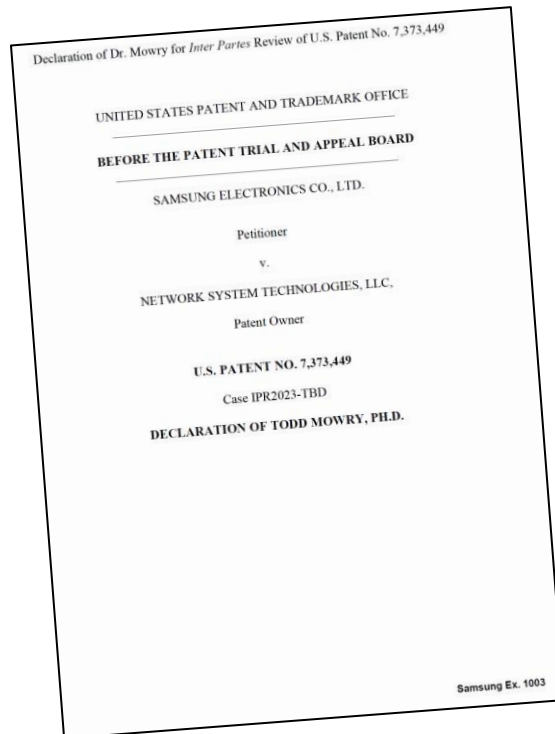
The '449 Patent



Ex-1001, Fig. 3, see also id. 9:66-10:64

Technology and Motivations

Dr. Mowry testimony (Ex-1003):

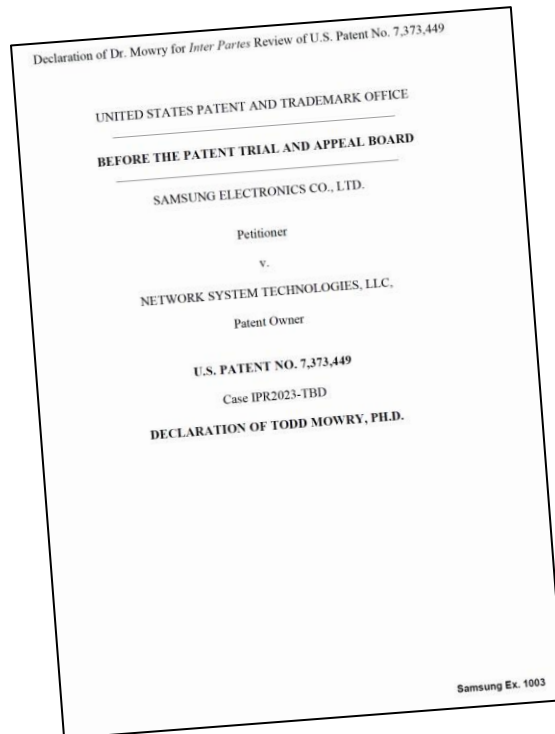


30. Computer systems are comprised of a number of different components, including processors (e.g., CPUs, GPUs, etc.), memory (e.g., SRAM, DRAM, etc.) and various input/output (I/O) devices. As the capacity of integrated circuit (IC) technology has improved exponentially over the years, it has enabled larger portions of computer systems to be integrated onto single chips. For example, in the early

31. As IC capacity has continued to increase well beyond what is needed to implement a single processor, it has enabled “Systems-on-Chip” (SoC) designs that include even more system components on a single chip; usually some combination of processors (often including multiple CPUs and possibly accelerators such as GPUs), portions of the memory hierarchy, and possibly I/O devices or controllers. SoCs were well-known by the late 1990s, and are integrated on a silicon die. *See, e.g.,* Ex-1014 (Jantsch) at vii (“During the 1990s more and more processor cores and large reusable components have been integrated on a single silicon die, which has become known under the label System on Chip (SoC).”); Ex-1008 (Wolf) at 1 (“An SoC is an integrated circuit that implements most or all of the functions of a complete electronic system.”); Ex-1009 (Badawy) at 1 (“With the increase in the number of logic gates that can be implemented on a single chip, various functionalities ... can be integrated into a single silicon chip, realizing an entire system on chip (SoC)”). Thus, SoCs can also be referenced as System on Silicon.

Technology and Motivations

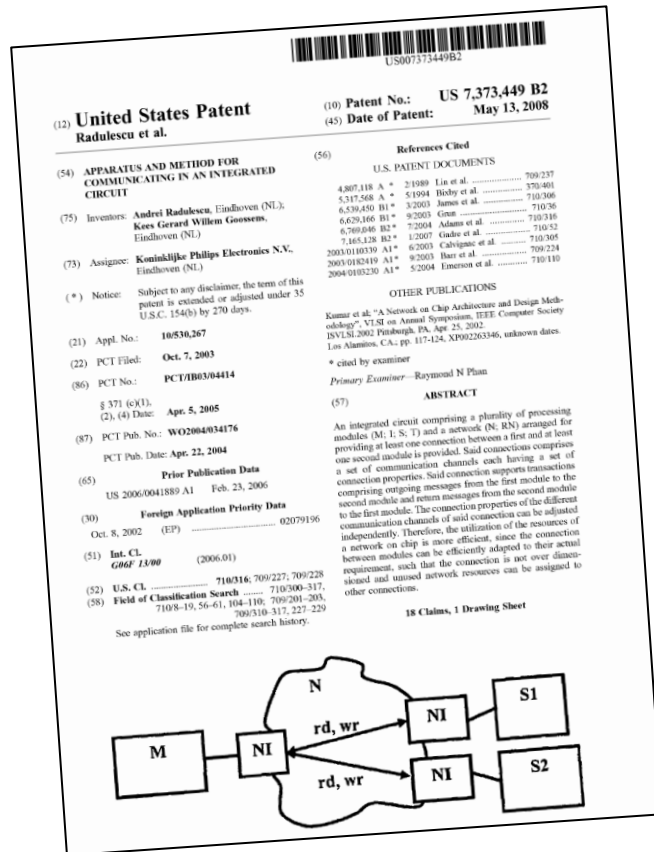
Dr. Mowry testimony (Ex-1003):



33. By integrating the interconnection network fully onto a chip, NoC designs can potentially provide lower cost, lower latency, and higher bandwidth relative to off-chip designs. To take full advantage of these potential benefits and to provide scalability as the number of on-chip components increases, NoCs have borrowed many design features from scalable interconnection networks that were designed for large-scale parallel machines in the 1990s. See Ex-1015 (Culler) at 749–828; Ex-1010 (Radulescu) at 277 (“Networks have been the subject of research for decades, both in the context of local and wide area networks (computer networks) [], and as an interconnect for parallel machines []. Both are very much related to on-chip networks, and many of the results in those fields are also applicable on chip.”);

Technology and Motivations

'449 patent:



Most of these topics have been already the subject of research in the field of local and wide area networks (computer networks) and as an interconnect for parallel machine interconnect networks. Both are very much related to on-chip networks, and many of the results in those fields are also applicable on chip. However, NoC's premises are different from off-chip networks, and, therefore, most of the

(Ex-1001, 1:61-67)

Only documents PO’s expert cites

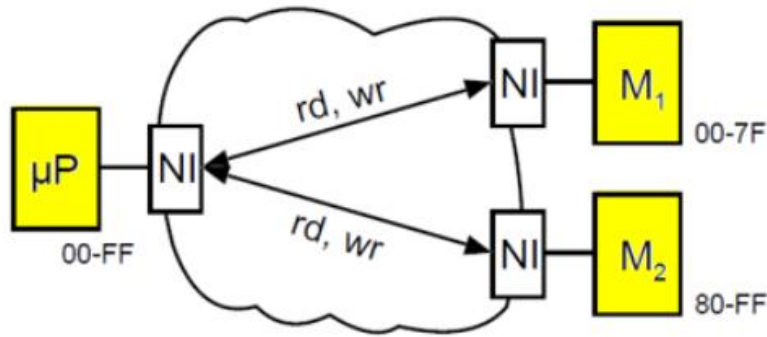
Exhibit No.	Description
2045	Plaintiff Network System Technologies, LLC’s Supplemental Claim Construction Brief, <i>Network System Technologies, LLC v. Qualcomm Incorporated et al.</i> , Civil Action No. 1:22-cv-1331-DAE (W.D. Tex.), Dkt. 106 (Mar. 1, 2024)
2046	Defendants’ Supplemental Invalidity Contention Cover Pleading, <i>Network System Technologies, LLC v. Qualcomm Incorporated et al.</i> , Civil Action No. 1:22-cv-1331-DAE (W.D. Tex.) (Jan. 23, 2024)
2047-2049	RESERVED
2050	Defendants’ Invalidity and Ineligibility Contentions And Disclosures Pursuant to Patent Rule 3-3 Cover Pleading, <i>Network System Technologies, LLC v. Samsung Electronics Co., Ltd., et al.</i> , Civil Action No. 2:22-cv-00481-JRG (E.D. Tex.) (July 11, 2023)
2051-2061	RESERVED
2062	Statutory Disclaimer dated March 25, 2024
2063-2069	RESERVED
2070	Declaration of Dr. Nader Bagherzadeh
2071	<i>Curriculum vitae</i> of Dr. Nader Bagherzadeh
2072	A survey of Research and Practices of Network on Chip
2073	Routing Algorithms for On Chip Networks
2074	McGraw Hill Dictionary
2075	IEEE Dictionary
2076	Supported Token-Ring Functions – IBM Documentation
2077	Token Ring Solutions
2078	Transcript of Nov. 4, 2024 Deposition of Dr. Todd Mowry
2079	Transcript of Nov. 15, 2024 Deposition of Sean O’Bryan

Drake's QoS-12 component not “redundant”

Goossens2003

+

Drake teachings



Ex-1003 (Dr. Mowry Decl.), ¶170
(annotations of Goossens2003, Fig. 4.5)

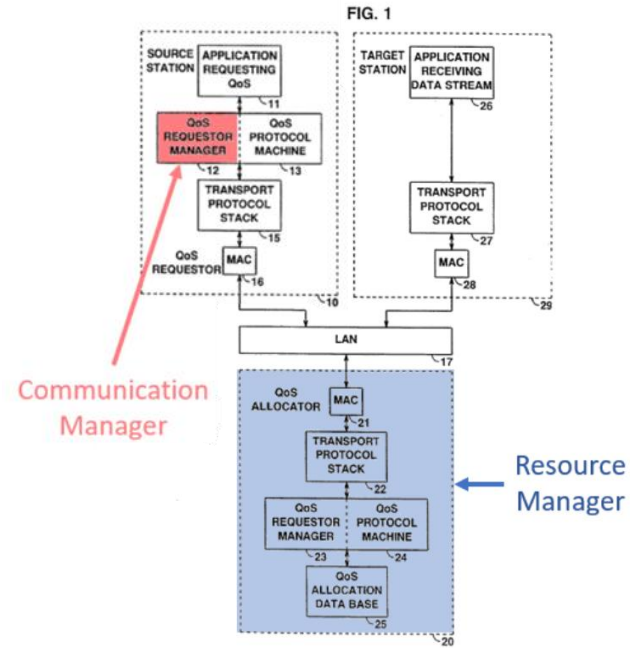


Exhibit 1003 (Dr. Mowry Decl.), ¶¶93, 121
(composite annotations of Drake, Fig. 1)

Drake’s QoS-12 component not “redundant”

“The Board stated that ‘[i]f . . . Kabemoto already addresses [the] problem [of cache coherency] through the use of a known technique similar to that of Bauman's, [it] fail[ed] to see why one of ordinary skill in the art would regard Bauman's technique as an obvious improvement to Ka-bemoto.’ But the Board's reasoning belies its conclusion. That Kabemoto and Bauman address the same problem and that Bauman's cache was a known way to address that problem is precisely the reason that there's a motivation to combine under *KSR* and our precedent.

And contrary to the Board's suggestion, Intel never had to show that replacing Kabemoto's secondary cache with Bauman's secondary cache was an ‘improvement’ in a categorical sense. Intel just had to show that Bauman's secondary cache was a ‘suitable option’ to replace Kabemoto's secondary cache.

Intel Corp. v. PACT XPP Schweiz AG, 61 F.4th 1373, 1380-81 (Fed. Cir. 2023) (internal citations omitted)



Drake's QoS-12 component not "redundant"

Dr. Mowry testimony:

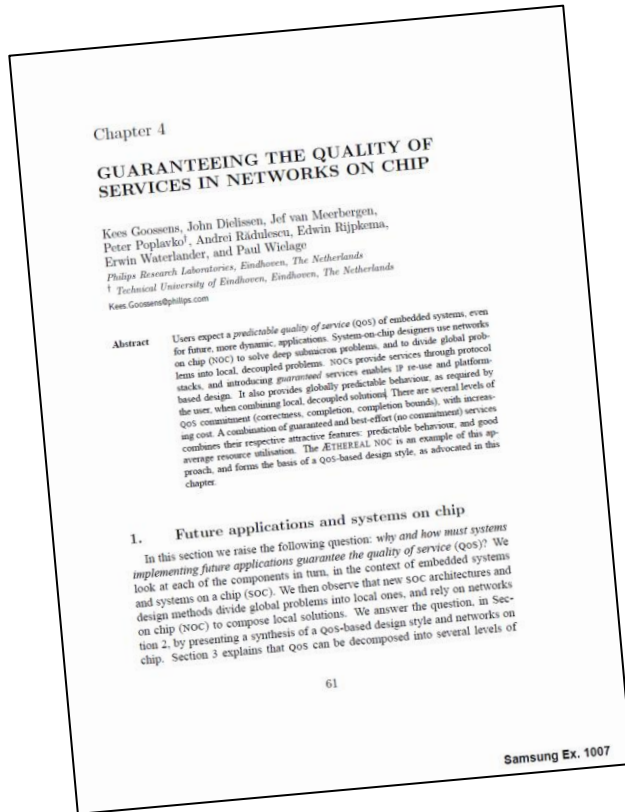
51. In addition, in the combination, the "QoS requestor manager" would not be redundant of the functioning of Goossens2003's IP blocks. As I discussed above, network interfaces "decouple computation from communication" by enabling their host/IP module to communicate without requiring specific knowledge of the low-level network details. A POSITA would understand that components such as the QoS requestor manager 12 would desirably be included in the NOC of Goossens2003.

Ex-1043, ¶¶51-52

52. A POSITA would understand that Goossens2003's NOC would desirably include a component to take the signal from the "re-use[d]" IP block and communicate with the network. Exhibit 1007, Abstract. Different re-used IPs would typically use different protocols to describe the type of service it wanted to send its data over. Exhibit 1010 at 1. Therefore, Goossens2003's NOC would have been expected to use a component such as Drake's QoS requestor manager 12 to assemble/format a request that could be understood by the network. That is exactly what Drake teaches the QoS requestor manager 12 is for. Exhibit 1006, 4:33-35 ("components 12 and 13 assemble a request"), 4:62-65 ("QoS requestor manager 12 interacts with application 11...to format the QoS request"). A POSITA would not expect such an interfacing component would already be included in the "re-used" IP block, which does the computation and sends messages without regard to the communication system it is using. The fact that this specific component was not disclosed in Goossens2003 is the result of Goossens2003's disclosure not including the details of the QoS components. A POSITA would not expect that Goossens2003's network would be able to perform its QoS functionality without a component to assemble the QoS request from the different re-used IP protocols.

Drake's QoS-12 component not “redundant”

Goossens2003 (Exhibit 1007):



Abstract

Users expect a *predictable quality of service* (QoS) of embedded systems, even for future, more dynamic, applications. System-on-chip designers use networks on chip (NOC) to solve deep submicron problems, and to divide global problems into local, decoupled problems. NOCs provide services through protocol stacks, and introducing *guaranteed* services enables IP re-use and platform-based design. It also provides globally predictable behaviour, as required by the user, when combining local, decoupled solutions. There are several levels of

Drake's QoS-12 component not “redundant”

Dr. Mowry testimony:

67. Second, as I also explained above, this argument ignores the role of interfaces in networking (on-chip and off-chip). While it is true that the QoS originates from the requesting module (for example application 11 in Drake's source station), a POSITA at the time knew it was good design practice to raise the level of abstraction for requests so that the requesting module did not need to be redesigned for different implementations. A POSITA would have recognized that Drake's QoS requestor manager 12 was an interface that understood how to forward the QoS request from the requesting module to the resource manager, as that is what Drake explains requestor manager 12 did (e.g., formatting the request appropriately). Exhibit 1006, 4:33-38. A POSITA would understand that the QoS requestor manager 12 would have this same function in the Goossens references. In fact, the Goossens references themselves explain that “network interfaces” are used to interface with the (interchangeable/reusable) IP modules. Exhibit 1007 (Goossens2003), 74 (“Both networks contain two components: routers and network interfaces ... network interfaces are required to bridge the network layer and transport layer”), 76-77 (“IPs negotiate with network interfaces to obtain connections with certain properties.”), Fig. 4.5; Exhibit 1005 (Goossens2003), §3, §4.

Ex-1043, ¶67
Slide 42

Drake's QoS “requestor manager” 12

Dr. Mowry testimony:

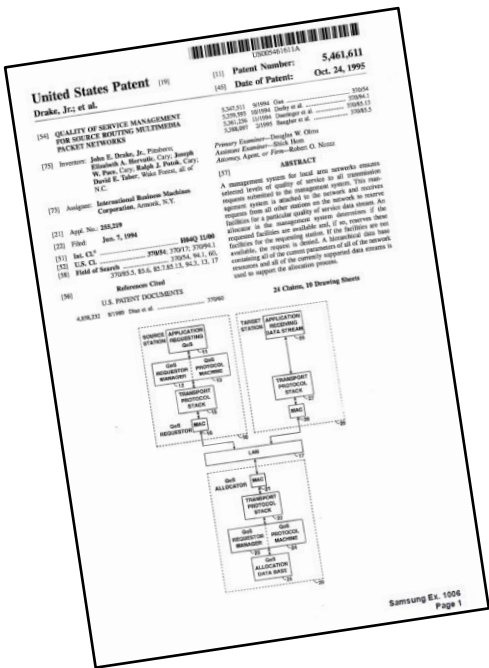
28. A POSITA would have understood Drake to teach that the QoS requestor manager 12 is part of the network interface components for source station 10. This would have been understood because (A) network interfaces were commonly included as part of a host in an off-chip LAN system (*see* Exhibit 2070, ¶77); (B) Drake specifically teaches that the functionality of its QoS request manager 12 includes “assembl[ing]” and “format[ing]” the QoS request to enable communication with networking components like the QoS allocator 20 (Exhibit 1006, 4:32-37, 4:62-65), which a POSITA would recognize was interfacing; (C) the surrounding components 13-16 of source station 10 are also network interface-type components involved in enabling communication with network components in the LAN (Exhibit 1006, 4:32-47); and (D) the location of the components 12 through 16—between the application 11 and LAN 17—is the location of a network interface (Exhibit 1006, FIG. 1). Thus, a POSITA would understand Drake to teach the QoS requestor manager 12 in the network interface.

Ex-1043, ¶128

Slide 43

Drake's QoS “requestor manager” 12

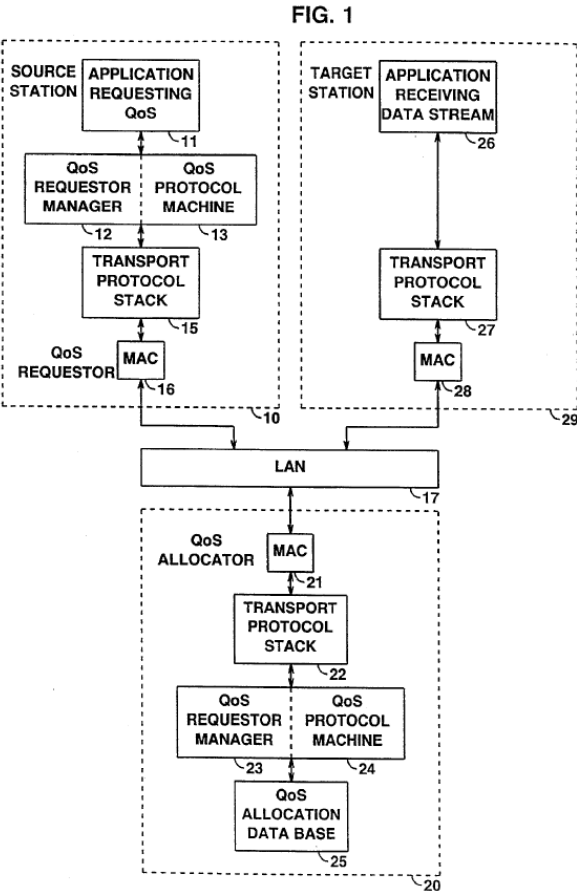
Drake:



transport protocol stack, 15 to the target station 29. Source station 10 also includes a QoS requestor manager 12 and a QoS protocol machine 13. Together, components 12 and 13 assemble a request to reserve a QoS connection for a multi-media data stream on that path through the LAN 17 between source station 10 and target station 29. Using a

[...]

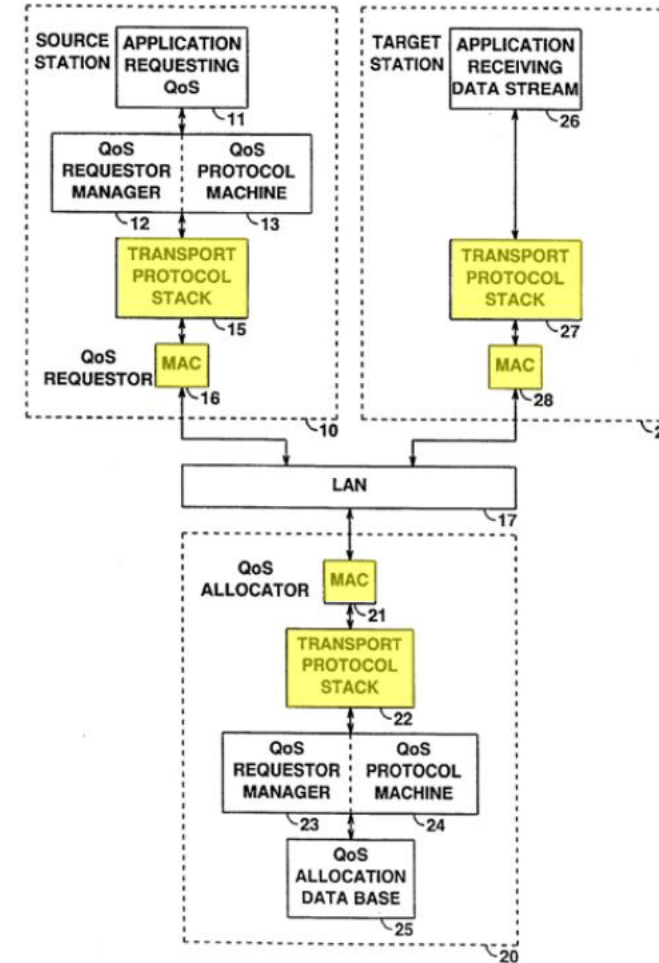
It should be noted that application 11 may simply be a file server application which causes multimedia data to be transported from a multimedia data base to LAN 17. One such server application is the IBM OS/2 LAN Server program described in IBM Manual S04G-1046. The QoS requestor manager 12 interacts with application 11 using internal messages as is well known in the art and utilizes QoS protocol machine 13 to format the QoS request.



Ex-1006, 4:32-37, 4:62-65 and Fig. 1

Drake's QoS-12 component not “redundant”

PO Sur-Reply, Paper 55, p. 10, arguing “QoS-12 is not a standard part included in all stations attached to LAN 17 (see target station 29)”



Drake's QoS-12 component not "redundant"

Dr. Mowry testimony:

14 Q Okay. Now I want to focus on the target station
15 29 in Drake. There are no QoS components shown in target
16 station 29 in Figure 1 of Drake; correct?

17 A It is true that target station 29 in Figure 1
18 does not contain QoS requestor manager 12 or QoS protocol
19 machine 13, and the reason for that is they are simply
20 illustrating what's necessary to send something from a
21 particular source to a particular target.

22 In practice, if the target station was also able
23 to initiate quality of service communication requests,
24 then it would also contain those same components.

25 But they only need to be actively used by the

Page 44

1 source of a communication that involves quality of
2 service. And that's why they are shown -- that's why
3 I believe they're illustrated only in source station 10.

Page 45

1 A Yes, that's -- that's correct, because --
2 because the target station in this example is only
3 receiving communication. It does not need to actively
4 use any quality of service components.

5 So for that reason, I believe that that's the
6 reason why it's not illustrated in target station 29.

7 However, as I stated before, if -- in many cases
8 these things would be designed symmetrically. So if
9 target station 29 is able to initiate quality of service
10 communication with another module, then it would also
11 have QoS requestor manager 12 and QoS protocol machine
12 13.

13 But as the target only, it does not need to use
14 such components, and I believe that that's the reason why
15 they are not shown in -- in box 29 in Figure 1.

Ex-2088, 43:14-44:3, 45:1-15

Interfacing: off-chip LAN vs. on-chip

Off-chip LAN – network interface/network adapter components within host/computer unit

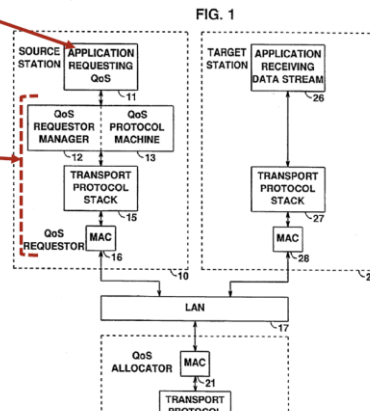
Dr. Mowry:

28. A POSITA would have understood Drake to teach that the QoS requestor manager 12 is part of the network interface components for source station 10. This would have been understood because (A) network interfaces were commonly included as part of a host in an off-chip LAN system (*see* Exhibit 2070, ¶77); (B) Drake specifically teaches that the functionality of its QoS request

(Ex-1043, ¶28)

Application 11 “requests a QoS connection,” and also transmits the multimedia data stream (for example, Exhibit 1006, 4:27-32, 4:48-50)

Network interface “components” (for example, Exhibit 1006, 4:34-35)



(Ex-1043, ¶15)

PO's Expert:

77. Another difference between NoCs and traditional computer networks is the modules that are communicating through each network. Traditional computer networks connect servers with personal computers or personal computers with other personal computers, for example. Embedded within these computers are specific network adapters that connect the computer to the network. Including network adapters (i.e., network interfaces) in a module on an SoC is highly disfavored for several reasons, one of which being that it prevents module re-use.

(Ex-2070, ¶77)

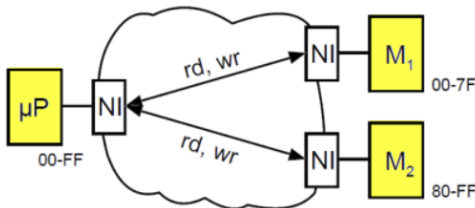
Interfacing: off-chip LAN vs. on-chip

On-chip – network interface outside of IP modules

Dr. Mowry:

27. A POSITA would have understood that a “difference” between off-chip and on-chip network interfaces was that network interfaces in a traditional off-chip network were typically part of the host (e.g., a server), whereas in a NoC they were separate from the IP module (as the Goossens references show). Exhibit 2070, ¶77. A POSITA would also have understood that this “difference” is merely box drawing exercise. The network interface in both off-chip and on-chip network is functionally disposed in the same location – between the communicating component (e.g., application 11 in Drake; IP module in Goossens2002/Goossens2003) and the rest of the network. Each of Drake, Goossens2002, and Goossens2003 references the use of network interfaces to facilitate communication between users of the network and the network. Exhibit

(Ex-1043, ¶127)



(Ex-1003, ¶170)

PO's Expert:

78. It is impractical to design a module on a NoC with a dedicated network adapter (i.e., network interface) because, if changes are made to the network, the module would similarly have to be changed. As a solution, SoC designers have decoupled the modules and the network adapters, thereby allowing the modules to merely communicate with a network adapter that is not integrated on the module.

(Ex-2070, ¶178)

25 Q. Where are the network interfaces in Goossens
1 located?
2 A. There's certainly not in the IPs, but they are
3 somewhere else.
4 Q. Are they in the network?
5 A. Could be.
6 Q. But you don't know?
7 A. It doesn't have to -- it doesn't say, but it
8 could be, yes.

(Ex-1037, 80:25-81:8)

Interfacing: off-chip LAN vs. on-chip

Dr. Mowry testimony:

[. . .]

strongly disagree. A POSITA would have been familiar with this “difference,” but that would have motivated a POSITA to place the communication manager (such as Drake’s QoS requestor manager 12) *outside* the IP module in the network-on-chip context, not inside it. In fact, this known “difference” between off-chip networks and on-chip networks is the reason why Drake showing the QoS requestor manager 12 located in source station 10 does not teach to include the QoS requestor manager 12 inside the IP module. I explain this in the following paragraphs.

26. As Dr. Bagherzadeh appears to agree that the general understanding in the art was that the network interfaces in an on-chip network context paralleled the network interfaces of a traditional off-chip network. Exhibit 2070, ¶77. That is, the network interfaces of a NoC were known to perform the same general function

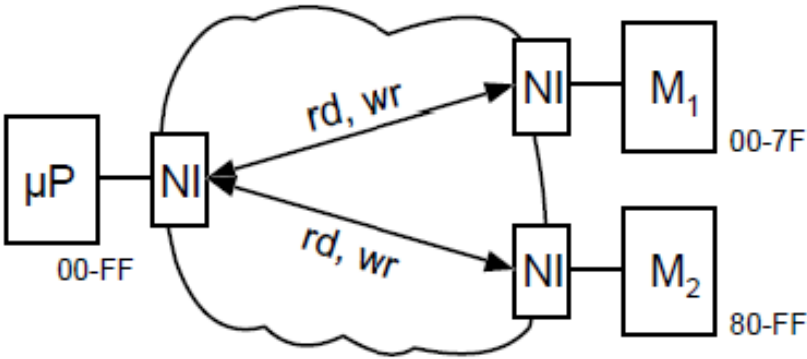
[. . .]

Ex-1043 ¶¶25-26, 61; see also *id.* ¶¶52-57

61. Dr. Bagherzadeh also asserts that Drake and Goossens2003 “teach completely opposite solutions” because QoS requestor manager 12 is in the source station 10 of Drake. Exhibit 2070, ¶262. As discussed above in paragraphs 25-29, 38-39, 52, and 54 above, Drake’s inclusion of the QoS requestor manager 12 in the source station 10 teaches a POSITA that a natural, obvious place to put it would be in the network interface of Goossens2003. POSITAs did not typically view a “network,” “NoC” or “LAN” as having rigid boundaries like suggested by Dr. Bagherzadeh. To the contrary, POSITAs looked at components and how they interact, which a POSITA understood was *the same* for both on-chip and off-chip network. For an off-chip network: the components included an application, which interacted with a network interface, which interacted with the transport components to transport a message. And for an on-chip network: the components also included an IP module, which interacted with a network interface, which interacted with transport components to transport a message. Because a POSITA

Purported “space constraints”

Goossens2003 (Ex-1007):



Purported “space constraints”

Dr. Mowry testimony:

Even if off-chip networks may have larger components, that would not mean that the general teachings of off-chip networks are not applicable, and Dr. Bagherzadeh provides no specific reason why a difference in size would render all teachings “impossible.”

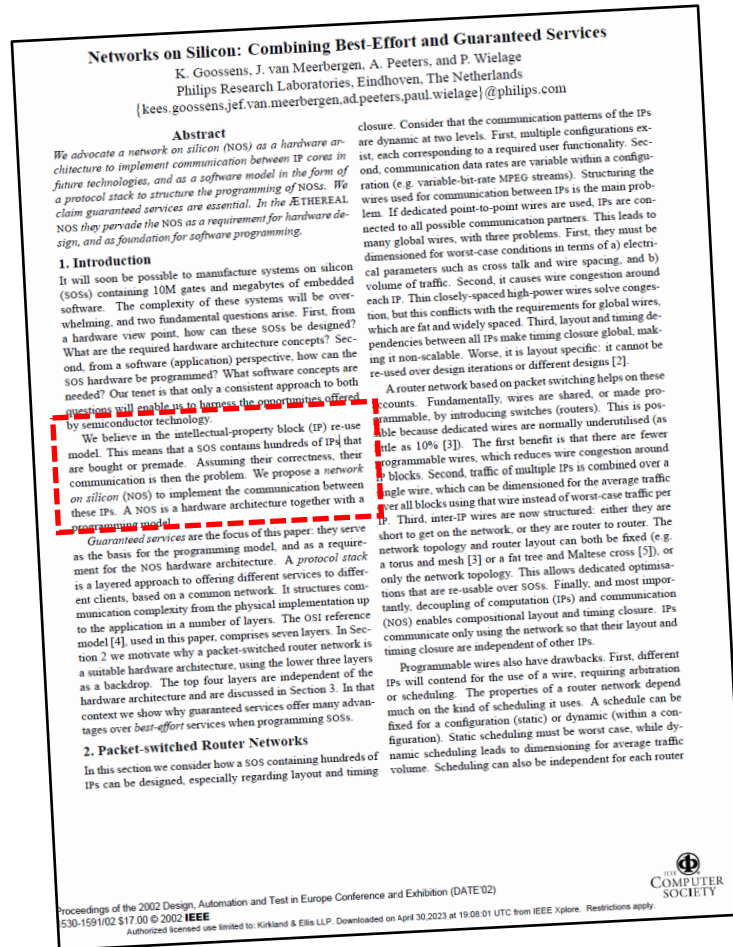
21. In fact, Dr. Bagherzadeh and I agree that essentially everything integrated onto a chip is just a smaller version of the off-chip component. For example, Dr. Bagherzadeh acknowledges that a “system-on-chip,” which is the focus of Goossens2002, Goossens2003, and the ’449 patent, is a larger system implemented in smaller form onto a chip. Exhibit 2070, ¶62 (“A system-on-chip (‘SoC’) is an integrated circuit in which *the components of a system are integrated on the chip*, including memories, processors, interfaces, and logic.”) (emphasis added). I similarly stated in my First Declaration: “[a]s the capacity of integrated circuit (IC) technology has improved exponentially over the years, it has enabled larger portions of computer systems to be integrated onto single chips.” Exhibit 1003, ¶30 (emphasis added); *see also id.*, ¶31 (citing references). This applies equally to implementing components from networks, such as the components in Drake. For example, the ’449 patent, Dr. Bagherzadeh, Goossens2002, and

22. Specific to the combinations of Drake with Goossens2002 or Goossens2003, Dr. Bagherzadeh relies on this “size” difference to suggest the proposed combination is faulty because “implementing a communication component outside of the IPs and NOS would require space and resources,” such as a separate circuit with its own network interface. Exhibit 2070, ¶¶263, 188. A POSITA knew that implementing a component anywhere on a chip (or off-chip) requires space and resources, and anything that communicates with a network typically required a network interface. These general facts do not mean that a POSITA would not be motivated to implement a component in a NOC, as Dr. Bagherzadeh appears to be suggesting. Moreover, as I noted above, Goossens2002 specifically explains that a system-on-silicon (SOS/SOC) can contain “hundreds of IPs” (Exhibit 1005, 1), indicating that space and resources (particularly for a chip containing only a small number of IPs) would not have been a dissuading factor.

Ex-1043, ¶¶20-23

Purported “space constraints”

Goossens2002 (Ex-1005):



We believe in the intellectual-property block (IP) re-use model. This means that a SOS contains hundreds of IPs that are bought or premade. Assuming their correctness, their communication is then the problem. We propose a *network on silicon* (NOS) to implement the communication between these IPs. A NOS is a hardware architecture together with a programming model.

Drake does not teach away

PO contends “*Drake teaches that it would be complex to implement QoS in the network,*” citing Ex-1006, 2:57-3:3

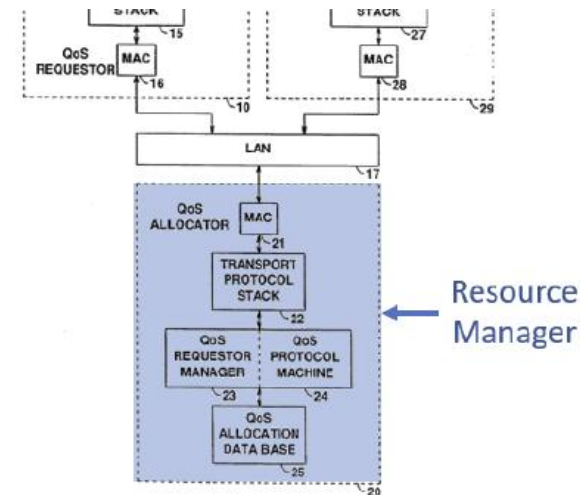
Ex-1006 (Drake), 2:57-3:3:

invention. A major advantage of the present invention is the ability to provide quality of service management in bridged LAN environments using all of the existing connection
60 set-up procedures, thereby providing such QoS management with little added complexity. It is possible, of course, to create in the QoS management system a route selection mechanism if such a mechanism does not already exist in the LANs themselves.

65 In accordance with another feature of the present invention, the quality of service management of a local area network is almost entirely concentrated in a single QoS management station on the LAN, thereby significantly reducing the complexity of introducing QoS management into existing LAN networks. The QoS management station

Ex-1006 (Drake), 4:66-67:

QoS allocator 20 is a station attached to LAN 17 which manages the QoS requests for the entire LAN 17. Like other



Drake does not teach away

PO contends “*Drake teaches that it would be complex to implement QoS in the network,*” citing Ex-1006, 2:57-3:3

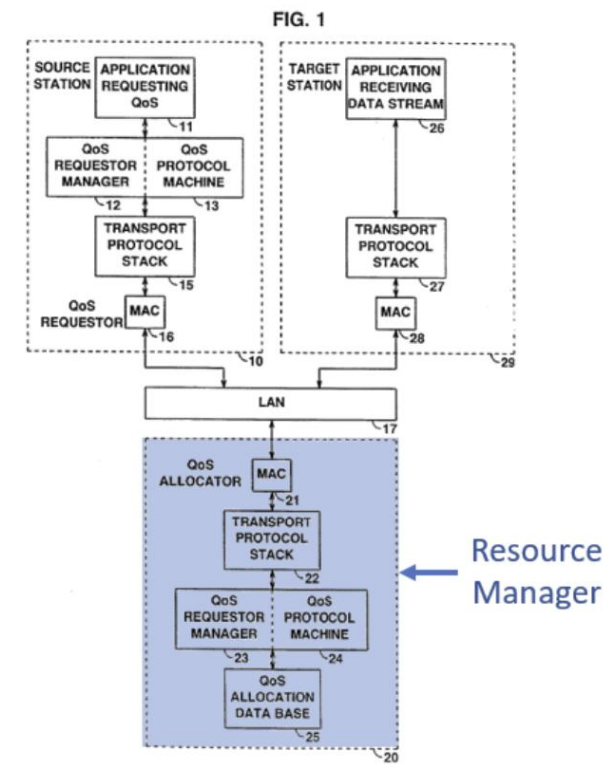
Dr. Mowry testimony:

doing so. Additionally, as I discussed above, in my opinion, a POSITA would have looked to Drake’s disclosures regarding a QoS allocator for details regarding the implementation of resource management in Goossens2003’s NOC, and would have been motivated to implement the functionality of Drake’s QoS allocator in a circuit on the same integrated circuit chip as Goossens2003’s IP blocks and NOC. As I

Ex-1003, ¶173

and the resource manager could go anywhere on the chip that is connected to the network (Exhibit 2078, 212:20-213:3), just as Drake shows and which is consistent with Goossens2003 and Goossens2002’s teaching of providing quality of service management separate from the IP modules.

Ex-1043, ¶¶38, see also id. 45



Ex-1003, ¶93

Communication Manager – NI a “natural” place to implement

Dr. Mowry testimony (first declaration):

184. Claim 2 depends from claim 1, which is obvious for the reasons that I explained above. Additionally, in my opinion, Goossens2003 in view of Drake renders this limitation obvious.

185. For example, Goossens2003 describes that its system includes “network interfaces,” and explains that that “end-to-end” connection properties—those properties which relate to the interactions between the master and slave modules, including “reordering, transaction completion, and flow control”—“are implemented by the network interfaces.” Ex-1007 (Goossens2003) at 76–77. Additionally, Goossens2003 further describes that “IPs negotiate with network interfaces to obtain connections with certain properties.” Ex-1007 (Goossens2003) at 77. In my opinion, this is just like the interaction between the modules and communication manager described in the ‘449 patent. Ex-1001 (‘449 patent) at 10:3–54.

Ex-1003, ¶¶184-185

Dr. Mowry testimony (first deposition):

10 Q. So here you wrote, “Additionally, in my
11 opinion, a POSITA would have been motivated to do so by
12 implementing the functionality of Drake’s QoS requestor
13 manager as a circuit on the same integrated circuit chip
14 as Goossens2002’s IP blocks and NOS.” Do you see that?
15 A. Yes.

[. . .]

24 Q. You have not explicitly defined in paragraph
25 122 where Drake’s QoS requestor manager would be

1 implemented as a circuit on the integrated circuit chip
2 in Goossens2002?

3 MR. KOHLHEPP: Object to form.

4 THE WITNESS: Well, there are -- I’m referring
5 to the teachings of Drake, and there would be several --
6 I mean, there would be several options for where this
7 might be implemented. It’s shown here in the network
8 interface. That would be a natural place to put it. I

Page 204

Ex-2078, 203:10-204:9

Dr. Mowry testimony (second declaration):

of a component to carry out that functionality. Exhibit 1003, ¶¶159, 173. I also explained that Goossens2003 similarly discloses communication management that is “implemented by” Goossens2003’s network interfaces.³ Exhibit 1003, ¶¶185-186; Exhibit 1007, 76-77. Thus, Drake and the Goossens references, as well as the general understanding in the art, would all teach a POSITA that a natural place to put components corresponding to the network interface (such as Drake’s QoS requestor 12), would have been in the network interfaces of the Goossens references. And in the network-on-chip context of Goossens2003, the interfaces at with the NOC, outside of the IP modules (and the same is true for Goossens2002).

33 and 102). Implementing Drake’s QoS components and data path in Goossens2003 or Goossens2002’s NOC architecture would have been routine and straightforward. For example, as I explained at my deposition a natural and obvious place to implement the communication manager is in Goossens’ network interface (which is where Goossens itself teaches to put communication management functionality, see Exhibit 1003 at ¶¶185-186) (Exhibit 2078, 204:7-8).

Ex-1043, ¶¶29, 38

Resource Manager – in a circuit on the same IC chip

Dr. Mowry testimony:

doing so. Additionally, as I discussed above, in my opinion, a POSITA would have looked to Drake's disclosures regarding a QoS allocator for details regarding the implementation of resource management in Goossens2003's NOC, and would have been motivated to implement the functionality of Drake's QoS allocator in a circuit on the same integrated circuit chip as Goossens2003's IP blocks and NOC. As I

Ex-1003, ¶173

and the resource manager could go anywhere on the chip that is connected to the network (Exhibit 2078, 212:20-213:3), just as Drake shows and which is consistent with Goossens2003 and Goossens2002's teaching of providing quality of service management separate from the IP modules.

Ex-1043, ¶¶38, see also *id.* 45

Reasonable expectation of success

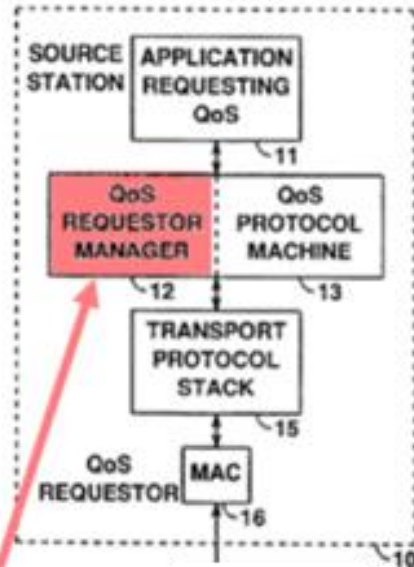
Petitioner's showing:

- Petition, pp.18-20 (citing Dr. Mowry, Ex-1003, ¶¶80-82, Ex-1005, Ex-1006, Ex-1001)
- Petition, pp.52-54 (citing Dr. Mowry, Ex-1003, ¶¶164-166, Ex-1007, Ex-1006, Ex-1001)
- Reply (Paper 43, 13-14) (citing Dr. Mowry, Ex-1043 ¶¶33-39)

PO Response (Paper 35), 30:

Spine., 567 F.3d at 1327. Moreover, under Petitioner's proposed definition, it would not be within the routine skill for a POSITA to implement Drake's QoS requestor manager in Goossens2002's NOS. EX2070, ¶¶201-210. Drake teaches that it would be complex to implement QoS in the network. EX1006, 2:57-3:3; EX2070, ¶206. It would be orders of magnitude more complex to implement the QoS requestor manager in the Goossens2002 NOS than the LAN in Drake. EX2070, ¶206. A POSITA under Petitioner's definition would not have a reasonable expectation of success in combining Goossens2002 and Drake in this manner. EX2070, ¶201-210.

Drake's QoS teachings – Communication Manager



Communication
Manager

118. For example, Drake discloses a “communication manager” in the form of a “QoS requestor manager.” Ex-1006 (Drake) at 4:32–37; *see also* Ex-1006 (Drake) at 4:62–65, 5:5–8, Fig. 1. Drake explains there are QoS requestor managers that are associated with source stations, and that the QoS requestor managers manage communications between modules by “assembl[ing] a request to reserve a QoS connection for a multi-media data stream ... between source station 10 and target station 29,” which request is sent to the QoS allocator (*i.e.*, resource manager). *Id.*

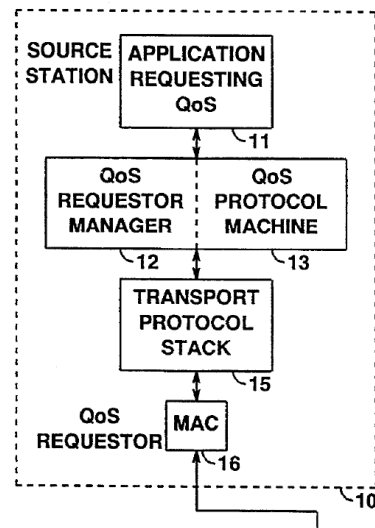
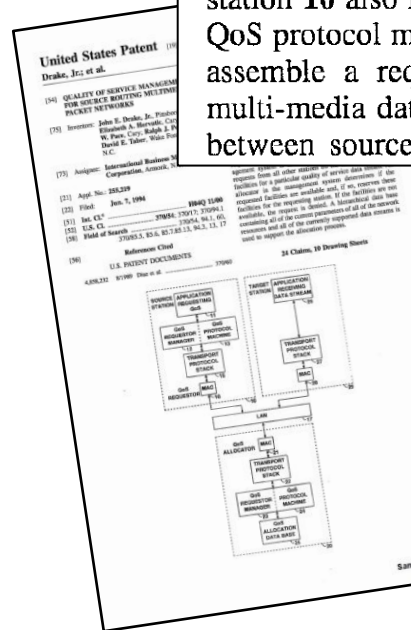
119. Indeed, this is consistent with the disclosure of the communication manager described in the specification of the '449 patent, which explains that “[t]he communication manager CM request[s] a connection with a set of properties between two modules from the resource manager RM.” Ex-1001 ('449 patent) at 10:19–24.

Ex-1003 (Dr. Mowry Decl.), ¶¶118-119;
see also id. ¶¶ 120-122, 130, 187, 195

Drake teaches separate QoS “components”

Drake:

More particularly, station 10 is the source of a desired multimedia transmission and comprises an application 11 which requests a QoS connection to the destination or target station 29. Application 11 also is responsible for causing the transmission of the multimedia data stream, via the LAN transport protocol stack, 15 to the target station 29. Source station 10 also includes a QoS requestor manager 12 and a QoS protocol machine 13. Together, components 12 and 13 assemble a request to reserve a QoS connection for a multi-media data stream on that path through the LAN 17 between source station 10 and target station 29. Using a



Ex-1006, 4:26-37, Fig. 1

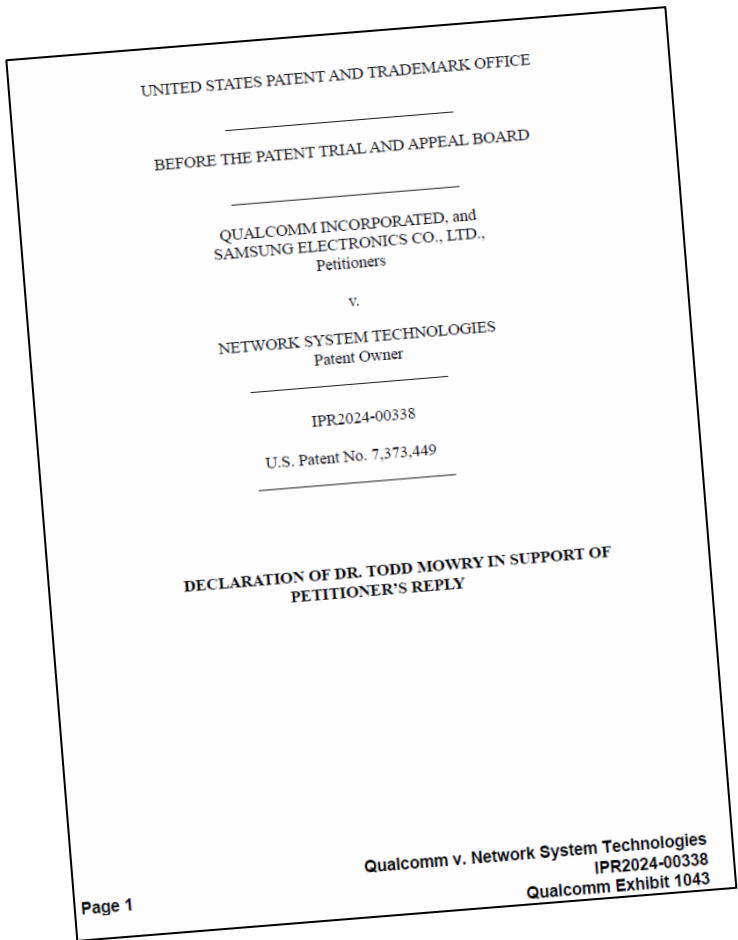
Dr. Mowry testimony:

31. Dr. Bagherzadeh appears to suggest that everything inside of source station 10 is software running on a common processing module. That is not what a POSITA would understand Drake to disclose. A POSITA would understand Drake to be disclosing and describing application 11 and QoS requestor manager 12 as separate components. See, for example, Exhibit 1006 (Drake) at FIG. 1 (showing application 11 and requestor manager 12 as separate components with arrows between them showing the data path); 4:34 (referring to “components 12 and 13”). A POSITA would understand from this description that QoS requestor manager 12 is already separate from application 11 in Drake, and there is no reason it could not be implemented apart from source station 10. A POSITA would have further understood that it was common design practice to have the network interface components (e.g., QoS requestor manager 12) as separate components from the main processing module (for example, application 11) as I explained above. *Supra*, ¶¶15, 27-28. This is exactly what Drake shows.

Ex-1043, ¶31, see also *id.* ¶¶56-57, 118-119

Drake and the claimed “first module”

Dr. Mowry testimony regarding new claim construction:

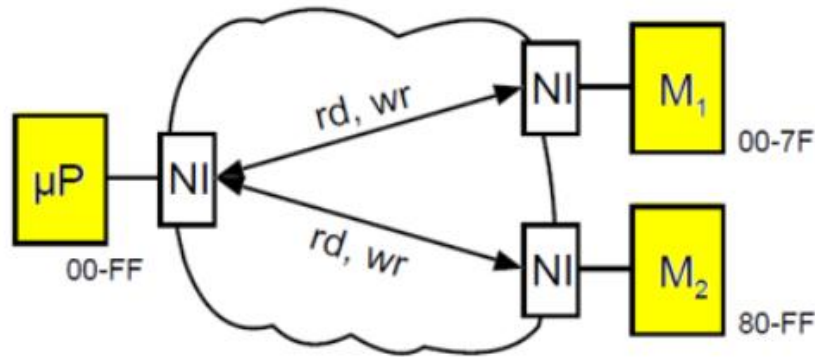


7. I have not been asked to provide an opinion on this claim construction issue. I also note that it would not affect my opinions. As I stated in my First Declaration and also explain below, the claims are met by a combination of Goossens2002 with Drake’s QoS teachings, and also Goossens2003 with Drake’s QoS teachings. In each of the combinations, the QoS components (including the communication manager) are not positioned in the first module but rather where the Goossens references already teach to put QoS functionality: outside of the modules, with the NOC. As I explain further below, I also point to “first module” and “communication manager” components in the prior art that are separate and distinct.

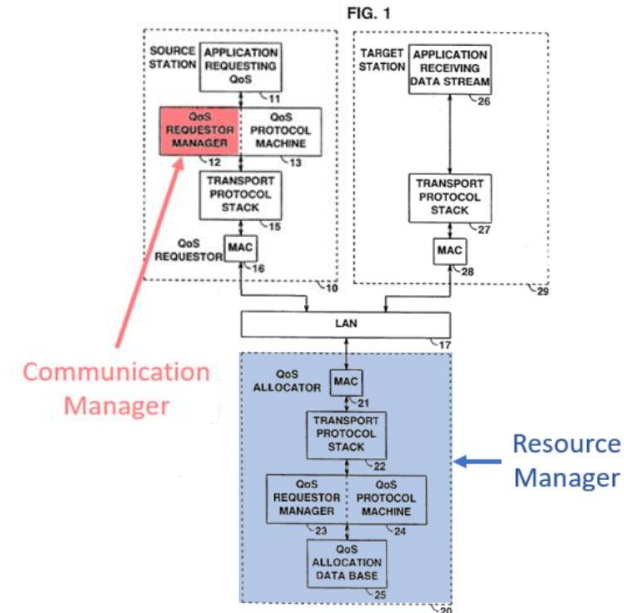
Ex-1043, ¶7

Goossens2003 + Drake

Goossens2003



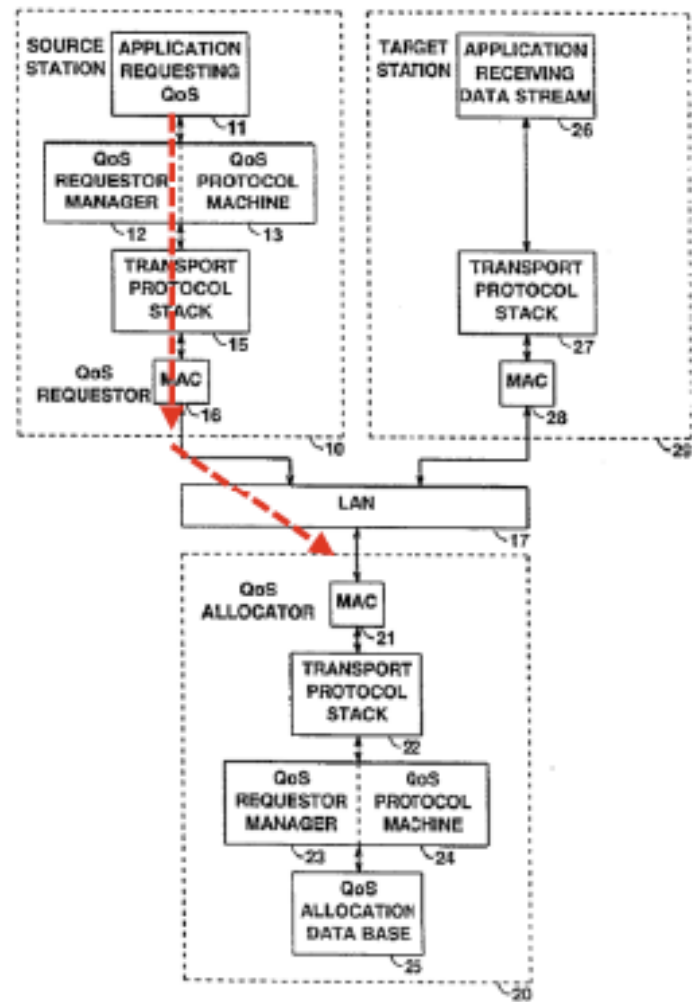
Drake teachings (composite annotations)



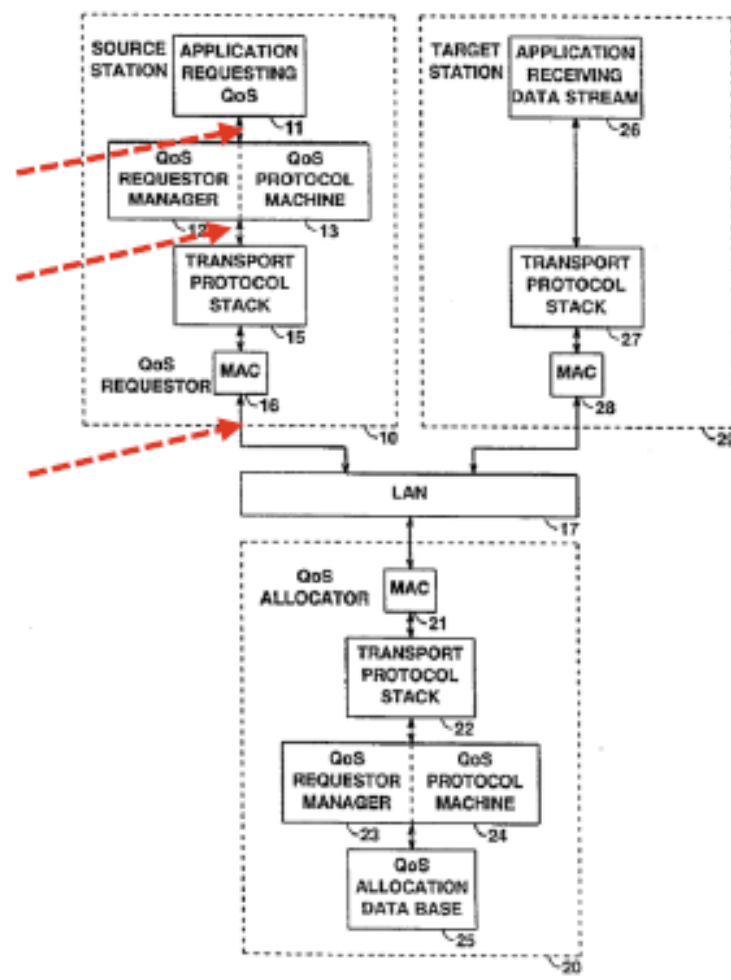
“A POSITA would have understood that using Drake’s QoS management process would improve the performance and efficiency of Goossens2003’s NOC by ensuring that connections established by IPs are able to provide the required services A POSITA would have understood that mature networking technologies like those described by Drake could be usefully applied in the context of newer networks on chip.”

Petition (-01081 IPR, Paper 1), 52-53

Drake's QoS teachings: data path



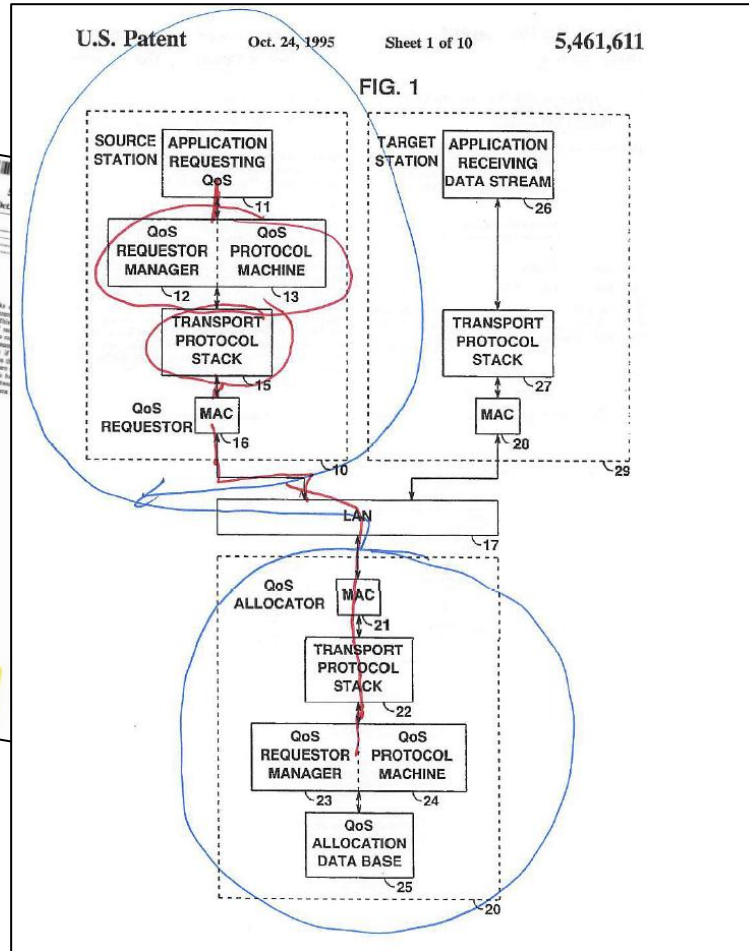
Ex-1043 (Dr. Mowry Decl.), ¶183



Ex-1043 (Dr. Mowry Decl.), ¶190

Ex-1036, 2

PO's expert agrees:



PO's expert agrees return path the same:

22 Q. So the message does go from QoS allocator 20 to
23 the application 11. Whether it ultimately goes past
24 that is something else -- is another question. But it
25 does go from QoS allocator 20 to application requesting

60

1 QoS?

2 A. Yes.

3 Q. Okay. What is the path that is disclosed by
4 Drake for that message going to application requesting
5 QoS?

MR. CHRISTOPHER: Objection to form.

THE WITNESS: It is the reverse of this path
that we have.

9 Q. The path going from the QoS allocator back to
10 the application requesting QS is the same as the red
11 path that you just drew, just in reverse; right?

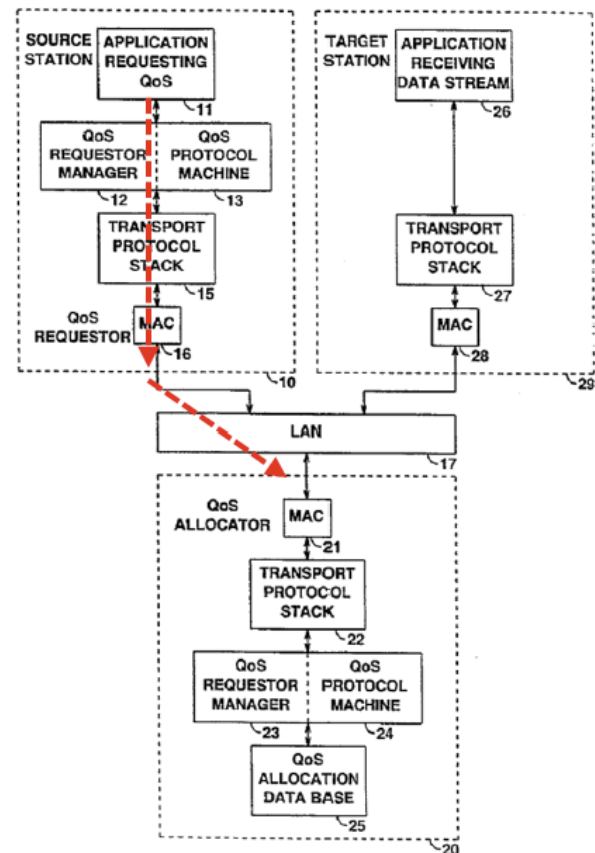
12 A. Explicitly, that would be the path to return,
13 yes.

Ex-1037, 59:22-60:8, 61:9-13

Drake's QoS teachings

Claim 6[b]/10[c]: “said communication manager (CM) is adapted to forward said request (REQ) for a connection with communication channels each having a specific set of connection properties to said resource manager (RM)”

Drake:



Ex-1043 (Dr. Mowry Decl.), ¶83

transport protocol stack, 15 to the target station 29. Source station 10 also includes a QoS requestor manager 12 and a QoS protocol machine 13. Together, components 12 and 13 assemble a request to reserve a QoS connection for a multi-media data stream on that path through the LAN 17 between source station 10 and target station 29. Using a [...]

stream transmitted from application 11 in station. 10. The overall function of allocator 20 is to receive the QoS request from station 10, examine the resources of LAN 17, and determine if a path exists in LAN 17 between station 10 and station 29 which satisfies the requested level of QoS. If the QoS exists, it is assigned to this QoS data stream. If the QoS is not available, station 10 is notified of the lack of an adequate QoS path to the intended destination.

It should be noted that application 11 may simply be a file server application which causes multimedia data to be transported from a multimedia data base to LAN 17. One such server application is the IBM OS/2 LAN Server program described in IBM Manual S04G-1046. The QoS requestor manager 12 interacts with application 11 using internal messages as is well known in the art and utilizes QoS protocol machine 13 to format the QoS request.

Ex-1006 (Drake),
4:32-37, 4:50-54, 4:62-65

Drake's QoS teachings

Claim 6[b]/10[c]: “*said communication manager (CM) is adapted to forward said request (REQ) for a connection with communication channels each having a specific set of connection properties to said resource manager (RM)*”

Dr. Mowry testimony:

83. Neither of these assertions would make sense to a POSITA and neither is supported by Drake. As to the first—that because Drake describes the request received by the QoS allocator 20 as coming from “source station 10,” a POSITA would understand that it was *not* forwarded by the QoS requestor manager 12 *in* source station 10—Drake shows the opposite. Drake does not disclose an alternative path, where the request somehow bypasses QoS requestor manager 12. To the contrary, Drake’s Figure 1 shows one path—the request goes through (is forwarded by) QoS requestor manager 12, and then ultimately is received by QoS allocator 20. The various lines with arrows illustrate the data path in this chain (i.e. 11↔12/13, 12/13↔15, 15↔16, 16↔17, etc.).

Ex-1043, ¶¶83, 85, see also *id.* ¶¶ 83-87

85. As to Dr. Bagherzadeh’s second assertion, that if a POSITA were to understand that any particular component in source station 10 was forwarding the request, it would be “transport protocol stack 15 and/or MAC entity 16, not QoS requestor manager 12” (Exhibit 2070, ¶288)—Drake does not support that either. A POSITA would understand the role of these components (protocol stack 15 and MAC entity 16) is to successfully transfer a data message that was received from QoS requestor manager 12 across the interconnect (LAN 17 in this case) to a destination. So, to the extent forwarding the request is just passing the message along, then every step in the chain in source station 10 is forwarding the message, including QoS requestor manager 12 and also then entities 15, and then 16. A

POSITA would understand that for the request to ultimately arrive at QoS allocator 20 from source station 10, the QoS requestor manager 12 has to forward it.

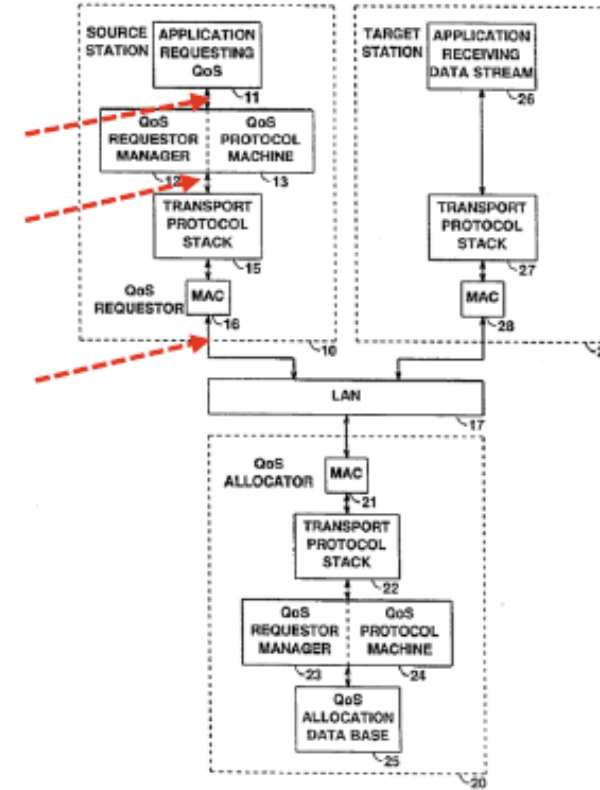
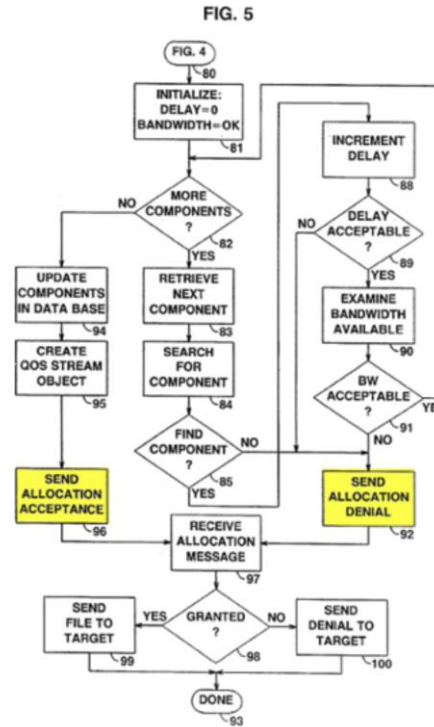
[. . .]

However, Drake teaches that the QoS requestor manager 12 realizes that it must first send a message somewhere else: to allocator station 20. Therefore, QoS requestor manager 12 is also forwarding in the sense of redirecting communication, in that it realizes the request should be sent to 20 (not 29).

Drake's QoS teachings

Claim 6[c]/10[e]: “*said resource manager (RM) responding with the availability of the requested connection to said communication manager (CM)*”

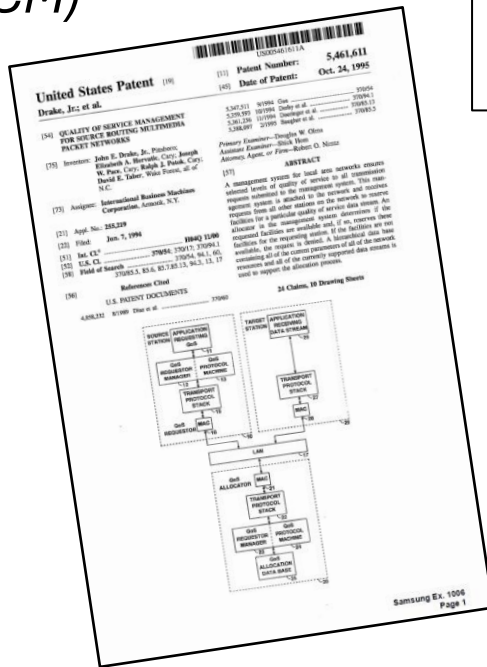
Drake:



Ex-1003 (Dr. Mowry Decl.), ¶¶133 , 197 Ex-1043 (Dr. Mowry Decl.), ¶¶88-93

Drake's QoS teachings

Claim 6[c]/10[e]: “said resource manager (RM) responding with the availability of the requested connection to said communication manager (CM)”



Drake:

station 29 which satisfies the requested level of QoS. If the QoS exists, it is assigned to this QoS data stream. If the QoS is not available, station 10 is notified of the lack of an adequate QoS path to the intended destination.

It should be noted that application 11 may simply be a file server application which causes multimedia data to be transported from a multimedia data base to LAN 17. One such server application is the IBM OS/2 LAN Server program described in IBM Manual S04G-1046. The QoS requestor manager 12 interacts with application 11 using internal messages as is well known in the art and utilizes QoS protocol machine 13 to format the QoS request.

Before proceeding to a description of the flow charts of FIGS. 4 through 11, the contents of the various messages exchanged between the stations of the local area network of FIG. 1 necessary to implement quality of service multimedia path management will be discussed. These messages include: the QoS allocation request, QoS allocation reply, QoS de-allocation announcement, QoS allocation change request, QoS change reply messages, QoS refresh announcement, allocation present announcement requestor present announcement, and requestor present reply messages. The contents of each of these messages are described below:

granted. An AllocationReply message is then sent from allocator station 20 (FIG. 1) to source station 10, indicating that the request has been granted. The source station 10, in turn, receives the AllocationReply message in box 183, and box 184 is entered to start an allocation refresh timer in source station 10 which is unique to this request.

Ex-1006, 4:55-65, 8:20-29, 14:22-27

Drake's QoS teachings

Claim 6[c]/10[e]: “*said resource manager (RM) responding with the availability of the requested connection to said communication manager (CM)*”

Dr. Mowry testimony:

90. Drake does not just disclose a black box source station 10. Instead, Drake shows and describes the components in source station 10, and the data path through those components. A POSITA would understand that the only way Drake's QoS works is for the reply regarding the QoS request to get back to application 11 in source station 10—the entity that originally made the QoS request. And Drake shows that to get back to application 11, the reply is first sent to QoS requestor manager 12, which Drake teaches is the interface with application 11. *E.g.*, Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). This is also clearly shown in Drake's Figure 1 (double-headed arrows for the data path, indicating to a POSITA that the reply comes back through the same path as the request):

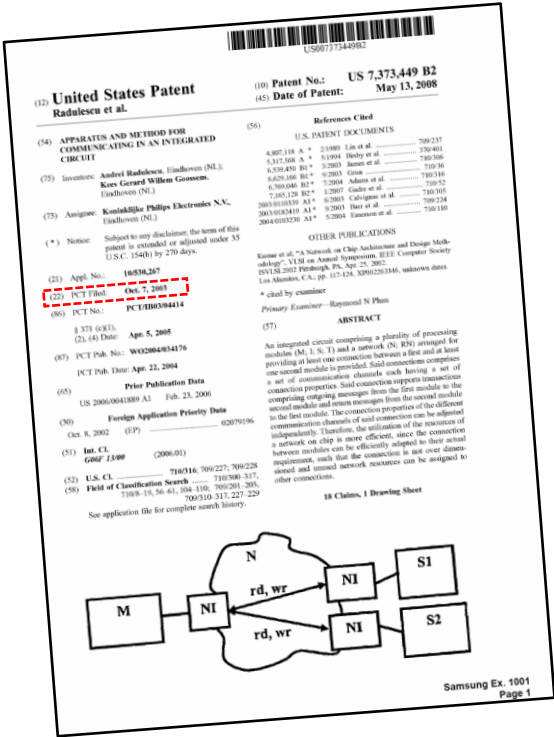
92. As I explained above Figure 1 shows the data path for communication (the lines with arrows at each end), and this indicates to a POSITA that the reply message goes back to component 12 before anything goes back to component 11. Exhibit 1006, Fig. 1. And Drake also expressly explains that QoS requestor manager 12 is the thing that interfaces with application 11—a POSITA would understand this is true for messages going in both directions. *See* Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). Moreover, there is no teaching in Drake of an alternative reply message route that bypasses the QoS requestor manager 12. Dr. Bagherzadeh states that the reply “could have been sent to application 11 or some component not depicted in Drake's Figure 1” (Exhibit 2070, ¶294), but that is not what Drake shows. Drake shows the reply message traveling back through QoS requestor manager 12 and QoS Protocol machine 13. *See* Exhibit 1040 at 60:3-61:13.

Ex-1043, ¶¶90, 92; *see also id.* ¶¶88-93

**Goossens2003, Goossens2002:
Printed Publication Prior Art**

Goossens2003, Goossens 2002 Are Prior Art

Priority date: **October 7, 2003**



(54) **APPARATUS AND METHOD FOR COMMUNICATING IN AN INTEGRATED CIRCUIT**

(75) Inventors: **Andrei Radulescu**, Eindhoven (NL);
Kees Gerard Willem Goossens,
Eindhoven (NL)

(73) Assignee: **Koninklijke Philips Electronics N.V.**,
Eindhoven (NL)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 270 days.

(21) Appl. No.: **10/530,267**

(22) **PCT Filed:** **Oct. 7, 2003**

(86) PCT No.: **PCT/IB03/04414**

§ 371 (c)(1),
(2), (4) Date: **Apr. 5, 2005**

(87) PCT Pub. No.: **WO2004/034176**

PCT Pub. Date: **Apr. 22, 2004**

(65) **Prior Publication Data**

US 2006/0041889 A1 Feb. 23, 2006

(30) **Foreign Application Priority Data**

Oct. 8, 2002 (EP) 02079196

Goossens2003 Is Prior Art

Declaration of UT Librarian Sean O'Bryan: Textbook catalogued by **April 2003** (Ex-1027)

DECLARATION OF THE UNIVERSITY OF TEXAS AT AUSTIN
LIBRARIES REGARDING THE "NETWORKS ON CHIP" REFERENCE

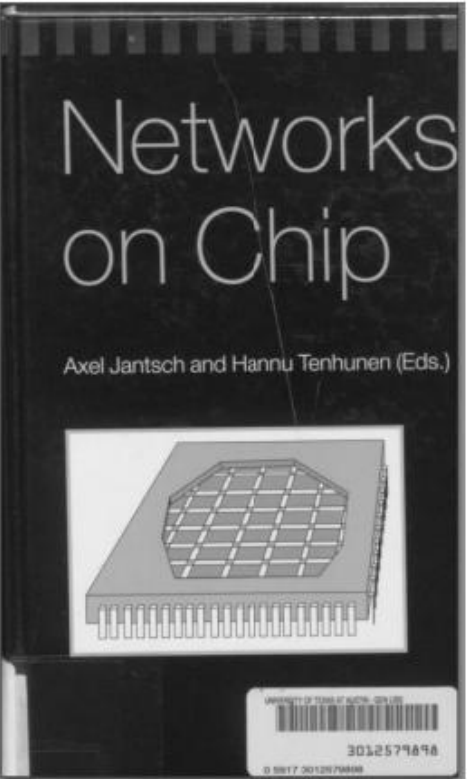
OPEN RECORDS REQUEST NO. R005527-113023

I, Sean O'Bryan, declare as follows:

Exhibit B

1. I am over the age of 18, have never been convicted of moral turpitude, and am legally competent to provide personal knowledge of the matters stated here.
2. I am a librarian at University of Texas Libraries at the University of Texas, Austin, TX.
3. I have been employed by UT Libraries for 22 years.
4. I am familiar with the routine record-keeping practices of the University of Texas Libraries.
5. The UT Libraries' records that are regularly updated reflect that "Networks on Chip" edited by Axel Jantsch and Hannu Tenhunen is in the UT Libraries' collection. It was catalogued on April 16, 2003.
6. A true and accurate copy of the cataloging record is attached as Exhibit A.
7. At the time of the acquisition of "Networks on Chip", the University typically made newly catalogued items available for cataloging.
8. I have reviewed the portions of "Networks on Chip" that are in the UT Libraries' collection and they reflect a true and accurate copy of the book as it exists in the UT Libraries.

* Note: In January 2020, UT Libraries completed a new Library Services Platform (ExLibris' Alma) and migrated our catalog of record. Prior to January 2020, I was employed as our cataloger. Prior to 2007, UT Libraries developed a catalog system (UTCAT/UTNetCat). In 2020, UT Libraries required migration to the new Alma platform.



Declaration of Librarian Sylvia Hall-Ellis, Ph.D: Textbook catalogued by **May 2003** (Ex-1035)

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

QUALCOMM INCORPORATED,
Petitioner*,

v.

NETWORK SYSTEM TECHNOLOGIES, LLC,
Patent Owner.

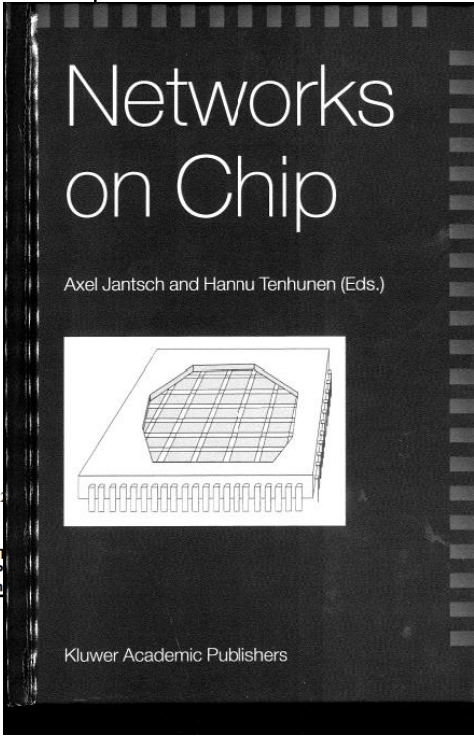
Case IPR2024-00338
Patent No. 7,373,449 B2

DECLARATION OF SYLVIA HALL-ELLIS, PH.D.

* Qualcomm Incorporated filed a petition and a motion for joinder in IPR 01081 and has been joined as a petitioner in this proceeding.

Page 1

Qualcomm v. Network System Technologies, LLC
IPR 01081
Qualcomm



Goossens2002 Is Prior Art

Goossens2002 publication (Ex-1005)

Declaration of Gordon MacPherson, IEEE records custodian (Ex-1025)

Networks on Silicon: Combining Best-Effort and Guaranteed Services

K. Goossens, J. van Meerbergen, A. Peeters, and P. Wielage
Philips Research Laboratories, Eindhoven, The Netherlands
[kees.goossens,jef.van.meerbergen,al.peeters,paul.wielage]@philips.com

Abstract

We advocate a network on silicon (NOS) as a hardware architecture to implement communication between 1P cores in future technologies, and as a software model in the form of a protocol stack to structure the programming of NOSs. We claim guaranteed services are essential. In the ETHERREAL NOS they pervade the NOS as a requirement for hardware design, and as foundation for software programming.

1. Introduction

It will soon be possible to manufacture systems on silicon (SOSs) containing 10M gates and megabytes of embedded software. The complexity of these systems will be overwhelming, and two fundamental questions arise. First, from a hardware view point, how can these SOSs be designed? What are the required hardware architecture concepts? Second, from a software (application) perspective, how can the SOS hardware be programmed? What software concepts are needed? Our tenet is that only a consistent approach to both questions will enable us to harness the communication offered

closure. Consider that the communication patterns of the 1Ps are dynamic at two levels. First, multiple configurations exist, each corresponding to a required user functionality. Second, communication data rates are variable within a configuration (e.g. variable-bit-rate MPEG streams). Structuring the wires used for communication between 1Ps is the main problem. If dedicated point-to-point wires are used, 1Ps are connected to all possible communication partners. This leads to many global wires, with three problems. First, they must be dimensioned for worst-case conditions in terms of a) electrical parameters such as cross talk and wire spacing, and b) volume of traffic. Second, it causes wire congestion around each 1P. Thin closely-spaced high-power wires solve congestion, but this conflicts with the requirements for global wires, which are fat and widely spaced. Third, layout and timing dependencies between all 1Ps make timing closure global, making it non-scalable. Worse, it is layout specific: it cannot be re-used over design iterations or different designs [2].

A router network based on packet switching helps on these

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 \$17.00 © 2002 IEEE



Guaranteed services are the focus of this paper: they serve as the basis for the programming model, and as a requirement for the NOS hardware architecture. A protocol stack is a layered approach to offering different services to different clients, based on a common network. It structures communication complexity from the physical implementation up to the application in a number of layers. The OSI reference model [4], used in this paper, comprises seven layers. In Section 2 we motivate why a packet-switched router network is a suitable hardware architecture, using the lower three layers as a backdrop. The top four layers are independent of the hardware architecture and are discussed in Section 3. In that context we show why guaranteed services offer many advantages over best-effort services when programming SOSs.

2. Packet-switched Router Networks

In this section we consider how a SOS containing hundreds of 1Ps can be designed, especially regarding layout and timing

1P. Thus, inter-1P wires are now structured: either they are short to get on the network, or they are router to router. The network topology and router layout can both be fixed (e.g. a torus and mesh [3]) or a fat tree and Maltese cross [5]), or only the network topology. This allows dedicated optimisations that are re-usable over SOSs. Finally, and most importantly, decoupling of computation (1Ps) and communication (NOS) enables compositional layout and timing closure. 1Ps communicate only using the network so that their layout and timing closure are independent of other 1Ps.

Programmable wires also have drawbacks. First, different 1Ps will contend for the use of a wire, requiring arbitration or scheduling. The properties of a router network depend much on the kind of scheduling it uses. A schedule can be fixed for a configuration (static) or dynamic (within a configuration). Static scheduling must be worst case, while dynamic scheduling leads to dimensioning for average traffic volume. Scheduling can also be independent for each router

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 \$17.00 © 2002 IEEE
Authorized licensed use limited to: Kirkland & Ellis LLP. Downloaded on April 30, 2023 at 19:08:51 UTC from IEEE Xplore. Restrictions apply.

COMPUTER SOCIETY

Samsung Ex. 1005

DocuSign Envelope ID: 4AC38FEB-58BD-49FA-A891-80215286816F

IEEE
Advancing Technology
for Humanity

DECLARATION OF GORDON MACPHERSON

I, Gordon MacPherson, am over twenty-one (21) years of age. I have never been convicted of a felony, and I am fully competent to make this declaration. I declare the following to be true to the best of my knowledge, information and belief:

1. I am Director, Board Governance & Policy Development of The Institute of Electrical and Electronics Engineers, Incorporated ("IEEE").
2. IEEE is a neutral third party in this dispute.
3. I am not being compensated for this declaration and IEEE is only being reimbursed for the cost of the article I am certifying.
4. Among my responsibilities as Director, Board Governance & Policy Development, I act as a custodian of certain records for IEEE.
5. I make this declaration based on my personal knowledge and information contained in the business records of IEEE.
6. As part of its ordinary course of business, IEEE publishes and makes available technical articles and standards. These publications are made available for public download through the IEEE digital library, IEEE Xplore.
7. It is the regular practice of IEEE to publish articles and other writings including article abstracts and make them available to the public through IEEE Xplore. IEEE maintains copies of publications in the ordinary course of its regularly conducted activities.
8. The article below has been attached as Exhibit A to this declaration:

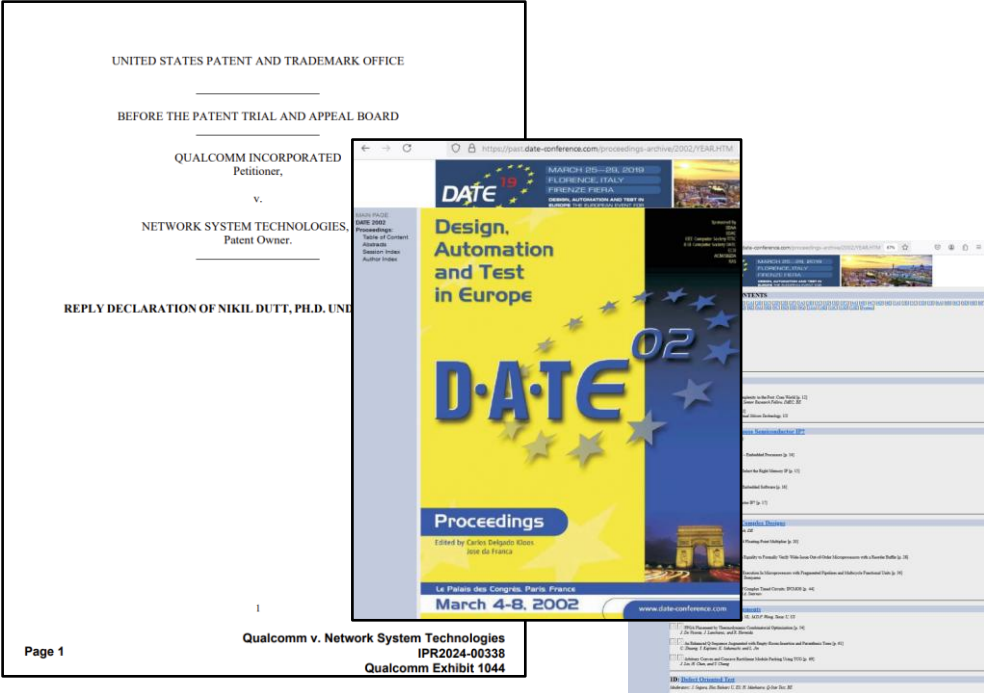
A. K. Goossens, et al.; "Networks on silicon: combining best-effort and guaranteed services", published in Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition, date of conference March 04-08, 2002.
9. I obtained a copy of Exhibit A through IEEE Xplore, where it is maintained in the ordinary course of IEEE's business. Exhibit A is a true and correct copy of the Exhibit, as it existed on or about December 5, 2023.

445 Hoes Lane Piscataway, NJ 08854

Samsung Ex. 1025
Page 1

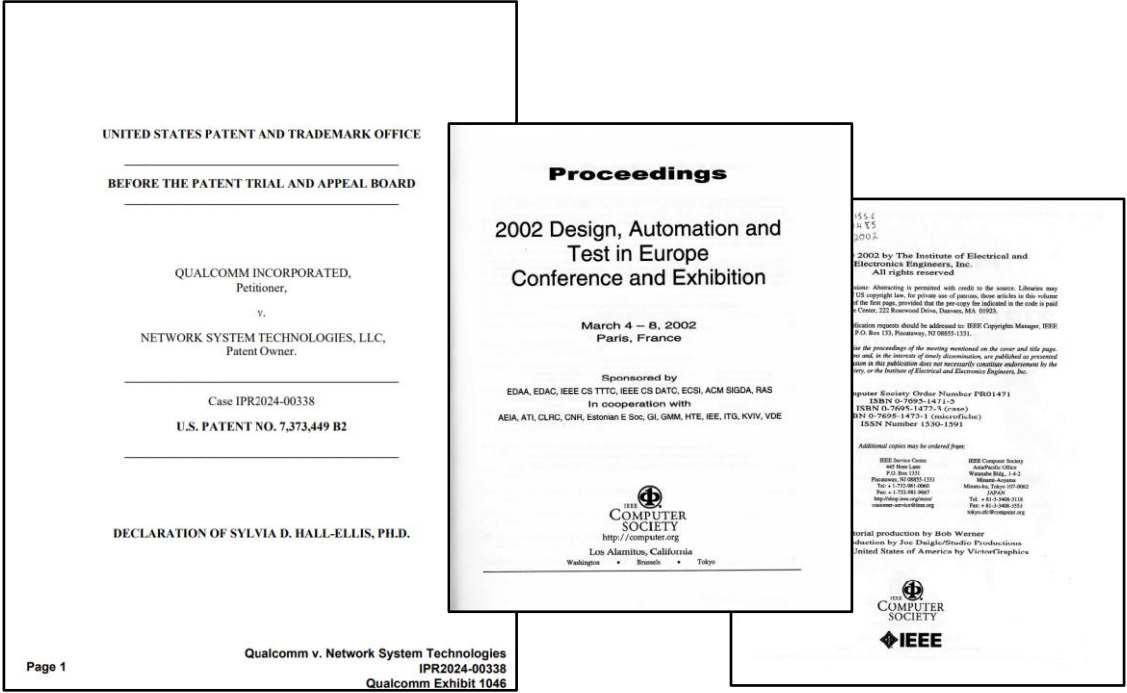
Goossens2002 Is Prior Art

Declaration of Prof. Nikil Dutt, Ph.D
(attended DATE 2002) (Ex-1044)



(Ex-1039 – DATE Archives)

Declaration of librarian Sylvia D. Hall-Ellis, Ph.D. (Ex-1046)



(Ex-1047 – Linda Hall Library copy (Kansas City, MO))

Goossens2002 Is Prior Art

Library of Congress Record (Ex-1048)

LIBRARY

CATALOG

Search Navigation

< 6 of 10 >

BOOK

Design, Automation, and Test in Europe Conference and Exhibition 2002 : proceedings ...

Full Record

MARC Tags

Meeting name

Design, Automation, and Test in Europe Conference and Exhibition (2002 : Paris, France)

Main title

Design, Automation, and Test in Europe Conference and Exhibition 2002 : proceedings : Paris, France, March 4-8, 2002 / sponsors, EDAA ... [et al.] ; in cooperation with AEIA ... [et al.]

Published/Created

Los Alamitos, Calif. : IEEE Computer Society, c2002.

Request this Item

LC Find It

Links

>

Links

Restricted to IEEE Xplore subscribers. <http://ieeexplore.ieee.org/servlet/opac?punumber=7834>

More Information

>

LCN Permalink

<https://lccn.loc.gov/2003268017>

Description

xiv, 1147 p. : ill. ; 28 cm.

ISBN

0769514715

LC classification

TK7870 .D467 2002

Cover title

DATE '02

Related titles

Design, Automation and Test in Europe Conference and Exhibition, 2002, proceedings.

Related names

EDA Association.
IEEE Computer Society

<https://catalog.loc.gov/vwebv/holdingsinfo?searchid=32959&recPointer=5&recCount=25&bibid=13065693>

1/2

U.S. Copyright Office Record (Ex-1038)

Copyright

United States Copyright Office

Try the Copyright Public Records System (CPRS) pilot with enhanced search features and filters.

Help Search History Titles Start Over

Public Catalog

Copyright Catalog (1978 to present) at DC4
Search Request: Left Anchored Title = design, automation and test in europe
Search Results: Displaying 6 of 7 entries

previous next

Labeled View

2002 design, automation, and test in Europe conference and exhibition ...

Type of Work: Text
Registration Number / Date: TX0005625531 / 2002-10-03
Title: 2002 design, automation, and test in Europe conference and exhibition : proceedings, March 4-8, 2002, Paris, France.
Imprint: Los Alamitos, CA : IEEE Computer Society, c2002.
Description: 1147 p.
Notes: Cover tit.: DATE '02, proceedings, edited by Carlos Delgado Kloos, Jose DaFranca.
Copyright Claimant: IEEE (employer for hire) Design, automation, and test in Europe conference and exhibition proceedings.
Date of Creation: 2002
Date of Publication: 2002-03-04
Variant title: 2002 design, automation, and test in Europe conference and exhibition : proceedings
Other Title: DATE '02, proceedings
Design, automation, and test in Europe conference and exhibition proceedings
Names: Kloos, Carlos Delgado
DaFranca, Jose
IEEE

previous next

Save, Print and Email (Help Page)

Select Download Format Full Record Format for Print/Save
Enter your email address: Email

Help Search History Titles Start Over

Contact Us | Request Copies | Get a Search Estimate | Frequently Asked Questions (FAQs) about Copyright | Copyright Office Home Page | Library of Congress Home Page

Reply (Paper 43), 26-28

Slide 74

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE