

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

RPX CORPORATION, HUAWEI ENTERPRISE USA, INC., HUAWEI
TECHNOLOGIES USA, INC., HUAWEI TECHNOLOGIES CO., LTD.,
AND HUAWEI INVESTMENT AND HOLDING CO.,
Petitioner,

v.

PARITY NETWORKS, LLC,
Patent Owner.

Case IPR2018-00100
Patent 6,738,378 B2

Before DEBRA K. STEPHENS, BRYAN F. MOORE, and
KAMRAN JIVANI, *Administrative Patent Judges*.

MOORE, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C § 318(a)

I. INTRODUCTION

RPX Corporation, Huawei Enterprise USA, Inc., Huawei Technologies USA, Inc., Huawei Technologies Co., Ltd., and Huawei Investment and Holding Co. (collectively, “Petitioner”) request *inter partes* review of claims 1–19 of U.S. Patent No. 6,738,378 B2 (“the ’378 patent,” Ex. 1101) pursuant to 35 U.S.C. §§ 311 *et seq.* Paper 2 (“Pet.”). Petitioner relies on the testimony of Dr. Tai Lavian. Ex. 1105. Parity Networks, LLC (“Patent Owner”) filed a preliminary response. Paper 6 (“Prelim. Resp.”). In a January 12, 2018 Institution Decision, we determined that Petitioner had a reasonable likelihood of prevailing as to at least one of the challenged claims of the ’378 patent. Paper 7 (“Inst. Dec.”). Accordingly, we instituted an *inter partes* review pursuant to 37 C.F.R. § 42.108. Inst. Dec. 33.

Patent Owner filed a Patent Owner Response (Paper 10, “PO Resp.”) to which Petitioner filed a Reply (Paper 16, “Reply”). Patent Owner also filed a Sur-Reply. Paper 20. Petitioner relies on the testimony of Dr. Jose Luis Melendez. Ex. 2001.

Petitioner moved to strike the Sur-Reply. Paper 24. Patent Owner filed an opposition to the motion to strike. Paper 25. Both parties requested a hearing for oral argument (Papers 19 and 21), and a hearing was held January 23, 2019. See Paper 28 (“Tr.”).

A. Related Matters

A decision in this proceeding could affect or be affected by the following cases pending in the United States District Court for the Eastern District of Texas and involving the ’378 patent: *Parity Networks, LLC v. Huawei Technologies USA, Inc.*, C.A. No. 6:16-cv-01306-RWS-KNM (hereafter “the Litigation”); *Parity Networks, LLC v. Alcatel-Lucent USA,*

Inc., C.A. No. 6:16-cv-01367-RWS-KNM; *Parity Networks, LLC v. Juniper Networks, Inc.*, C.A. No. 6:17-cv-00495; and *Parity Networks, LLC v. Ericsson, Inc.*, C.A. No. 6:17-cv-00526.

In addition to the instant Petition, Petitioner concurrently filed a second petition for *inter partes* review of claims 1–19 in IPR2018-00097 (Pet. vi).

B. The '378 Patent

The '378 patent is directed to routing digital data by a router's central processing unit (CPU) through a data-packet network (DPN), such as the Internet. Ex. 1101, 1:10–15. Data is transmitted through such a network in transmission control protocol (TCP) and Internet protocol (IP) "packets." *Id.* at 1:30–35.

The '378 patent uses "network node" to refer to devices in the network, including packet routers and computers. Ex. 1101, 3:21–23. As each packet is routed through the network from its source to a destination, the packet is typically processed along the way by multiple routers, each of which processes the packet's header to route the packet through the network. *Id.* at 1:17–30.

The '378 patent asserts that in conventional routers, a network access controller (NAC) queued all packets to be processed by the CPU on a first-come, first-processed basis. Ex. 1101, 2:8–9. The '378 patent notes that this is disadvantageous. For example, according to the patent, there was no way for the CPU to determine priority. *Id.* at 2:5–11. The '378 patent explains that this made conventional routers susceptible to denial-of-service attacks where large volumes of erroneous packets were sent to overload the router's CPU so that it was unable to process legitimate and useful packets. *Id.* at

2:11–17. The '378 patent states that techniques were needed to enable “a router or server to quickly identify and sort data packets that require special handling by category and priority so that the CPU is able to adequately process legitimate and useful packets.” *Id.* at 2:20–24.

In one embodiment, the alleged invention includes one or more packet processors enabled to sort the specific types of packets into two or more categories for processing by the CPU. Ex. 1101, 2:32–34. In the '378 patent not all packets received by the router are for processing by the CPU and the “specific types of packets” to be sorted in this embodiment are packets destined for the CPU (i.e., CPU-destined packets). To prevent CPU overload and thwart the denial-of-service attack, this embodiment sorts those packets into three categories as follows:

1. Data packets from known and trusted data sources;
2. Data packets from data sources known to be suspect; and
3. Data packets falling between the first two categories.

Id. at 5:3–8.

The '378 patents states “[d]ata packets from known and trusted data sources include legitimate and legal packets from neighboring routers and secure nodes, as well as internally generated data packets. Data packets from sources that are suspect include those from sources known within the network to have previously caused damage to network components.” *Id.* at 5:8–13.

In addition to the packet processors for sorting the specific type of packets, this embodiment of the invention includes a set of hardware queues of the NAC for queuing the packets according to category. *Id.* at 2:38–40. The '378 patent states that “one queue may be implemented for packets

determined to be from known and trusted sources, another for packets determined to be from suspect sources, and a third for packets that fall between these two categories.” *Id.* at 5:36–39.

Finally, the CPU processes the queued packets according to category. *Id.* at 2:35–37. The ’378 patent explains that in this three-queue embodiment, “priority for processing by the CPU is assigned, with the highest priority for the queue holding the trusted packets, and the lowest priority for the queue holding the packets highly suspect.” *Id.* at 5:43–46.

C. Illustrative Claim

Independent claim 1, reproduced below, is illustrative of the claimed subject matter:

1. In a network node having one or more packet processors and at least one CPU required to process specific types packets for CPU processing, the system comprising:
 - one or more packet processors enabled to sort the specific types of packets into two or more categories for processing; and
 - a set of hardware queues of a network access controller (NAC) for queuing the packets according to category;characterized in that the CPU processes the queued packets according to category.

Ex. 1101, 7:65–8:10.

D. Instituted Grounds of Unpatentability

Petitioner asserts that claims 1–19 are unpatentable based on the following grounds:

Reference(s)	Basis	Claims challenged
Wilford ¹	§ 103	1–4, 6–9, 11–16, 18–19
Wilford and Ayres ²	§ 103	5, 10, 17

Pet. 10, 14–68.

II. DISCUSSION

A. Relevant Law

1. Evidentiary Burden

Petitioner bears the burden of proving unpatentability of the challenged claims, and the burden of persuasion never shifts to Patent Owner. *Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015). To prevail in its challenges, Petitioner must demonstrate by a preponderance of the evidence that the challenged claims are unpatentable. 35 U.S.C. § 316(e); 37 C.F.R. § 42.1(d).

2. Obviousness

A claim is unpatentable under 35 U.S.C. § 103(a) if the differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter

¹ US Patent No. 6,687,247 B1, issued Feb. 3, 2004 (“Wilford,” Ex. 1108).

² US Patent No. 6,738,371 B1, issued May 18, 2004 (“Ayres,” Ex. 1104).

pertains. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations including: (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of skill in the art; and, (4) where in evidence, objective indicia of non-obviousness (i.e., so-called secondary considerations), including commercial success, long-felt but unsolved needs, failure of others, and unexpected results.³ *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966).

3. Level of Skill

The level of ordinary skill in the art usually is evidenced by the references themselves. See *Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001); *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995); *In re Oelrich*, 579 F.2d 86, 91 (CCPA 1978). For an obviousness analysis, prior art references must be “considered together with the knowledge of one of ordinary skill in the pertinent art.” *In re Paulsen*, 30 F.3d 1475, 1480 (Fed. Cir. 1994) (quoting *In re Samour*, 571 F.2d 559, 562 (CCPA 1978)). Moreover, “it is proper to take into account not only specific teachings of the reference but also the inferences which one skilled in the art would reasonably be expected to draw therefrom.” *In re Preda*, 401 F.2d 825, 826 (CCPA 1968). That is because an obviousness analysis “need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ.” *KSR*, 550 U.S. at 418.

³ Patent Owner does not put forth evidence it alleges tends to show secondary considerations of non-obviousness in its Patent Owner Response.

Petitioner asserts a person of ordinary skill in the art of the subject matter of the '378 patent would have had “(a) a bachelor’s degree in computer science or electrical engineering, and four or more years of experience with networks and networking devices, or (b) a master’s degree in computer science or electrical engineering, and two or more years of experience with networks and networking devices.” Pet. 10 (citing Ex. 1105 ¶¶ 30–34). Patent Owner does not offer a proposed level of skill in its Patent Owner Response and does not address Petitioner’s assertions regarding the education or experience of an ordinarily skilled artisan at the time of the invention. *See generally* PO Resp. We adopt Petitioner’s articulation of the level of skill and acknowledge that the level of ordinary skill in the art is also reflected by the prior art of record. *See Okajima*, 261 F.3d at 1355; *GPAC Inc.*, 57 F.3d at 1579 (Fed. Cir. 1995); *Oelrich*, 579 F.2d at 91.

B. Claim Construction

In an *inter partes* review, we construe claim terms in an unexpired patent according to their broadest reasonable construction in light of the specification of the patent in which they appear. 37 C.F.R. § 42.100(b) (2017).⁴ Consistent with the broadest reasonable construction, claim terms are presumed to have their ordinary and customary meaning as understood by a person of ordinary skill in the art in the context of the entire patent

⁴ Our recently changed version of this Rule, which requires that we interpret claims in the same manner used in a civil action under 35 U.S.C. § 282(b), does not apply here because the Petition was filed before the effective date of the new Rule, November 13, 2018. *See Changes to the Claim Construction Standard for Interpreting Claims in Trial Proceedings Before the Patent Trial and Appeal Board*, 83 Fed. Reg. 51,340, 51,344 (Oct. 11, 2018).

disclosure. *In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007).

All challenged claims require a “hardware queue.” Petitioner argues “under BRI, the term ‘hardware queue’ should be interpreted to include hardware that is configurable by software to form a queue.” Pet. 42 (citing Ex. 1105 ¶ 210). Petitioner also suggests that an alternative definition of the term could “require special-purpose hardware circuitry to control at least some portion of the queue.” *Id.* at 44. Patent Owner does not define explicitly “hardware queue.” Nevertheless, Patent Owner’s declarant asserts “[t]he plain and ordinary meaning of a hardware queue, is well known,” i.e., that “such a queue would be designed by a circuit designer and formed to function as a queue as part of the manufacturing process of an electronic circuit,” and that it differs from a software queue “formed by software operating on general purpose or multi-purpose hardware.” Ex. 2001 ¶ 34. Additionally, Patent Owner’s argument in the Patent Owner Response suggests “hardware queue” means that the queue cannot be a “software queue” and “must be implemented in hardware.” PO Resp. 7, 9–12 (stating “Petitioners do not assert that the queue must be implemented in hardware”).

The scope of the claim term “hardware queue” was discussed in the Decision Denying Institution in IPR2018-00097. There, the Board stated the

Specification does not state that queues are “implemented” in software or that software “forms” a queue; rather, the specification says that the queues are “configured” by software which determines “which and how many” queues to use. Ex. 1001, 6:47–52. Thus, [we] clarify that software cannot implement or form the queues themselves but the software configures existing hardware queues, as described and claimed.

IPR2018-00097 Inst. Dec. 10–11 (in this record as Ex. 1118).

“[A] sound claim construction need not always purge every shred of ambiguity.” *Eon Corp. IP Holdings v. Silver Spring Networks*, 815 F.3d 1314, 1318 (Fed. Cir. 2016) (quoting *Acumed LLC v. Stryker Corp.*, 483 F.3d 800, 806 (Fed. Cir. 2007)). Here, we address the portions of the parties’ constructions necessary to resolve the dispute. We agree with Patent Owner’s suggestion that the term “hardware queue” is a queue implemented and formed in hardware. Also, consistent with the Specification, software can be used to configure the queue, i.e., determine “which and how many queues will be used, what the priority is to be, and the method by which packets are selected from each queue for the CPU.” Ex. 1001, 5:64–67, 6:47–52. This construction is consistent with Patent Owner’s argument in its Patent Owner Response and Petitioner’s alternate argument in its Reply (Reply 9–10).

We determine that it is not necessary to provide an express interpretation of any other term of the claims. *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) (only terms that are in controversy need to be construed, and only to the extent necessary to resolve the controversy); *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (applying *Vivid Techs.* in the context of an inter partes review).

C. Asserted Obviousness over Wilford

Petitioner contends claims 1–4, 6–9, 11–16, and 18–19 are unpatentable under 35 U.S.C. § 103(a) as obvious over Wilford. Pet. 14–56. Petitioner explains how Wilford allegedly describes all of the claim limitations. *Id.* As explained below, Petitioner has shown by a

preponderance of the evidence that claims 1–4, 6–9, 11–16, and 18–19 would have been obvious over Wilford.

1. Wilford (Ex. 1108)

Wilford is directed to a router that “provides low latency routing based on packet priority.” Ex. 1108, Abstract. The routing system 100 includes one or more linecards 110 (each having a respective “slot” number) that connect to a communications network 1 and process packets to/from the network for the router. *Id.* at 4:54–57. Each linecard has a packet processor, as shown in figure 1, reproduced below, that can determine how to route some packets and direct each of those packets to the appropriate output interface of the routing system through which the packet can be routed back out into the communications network 1, via the same or a different linecard. *Id.* at Abstract, 1:64–2:6, 5:17–35, 6:3–22, 7:39–8:18.

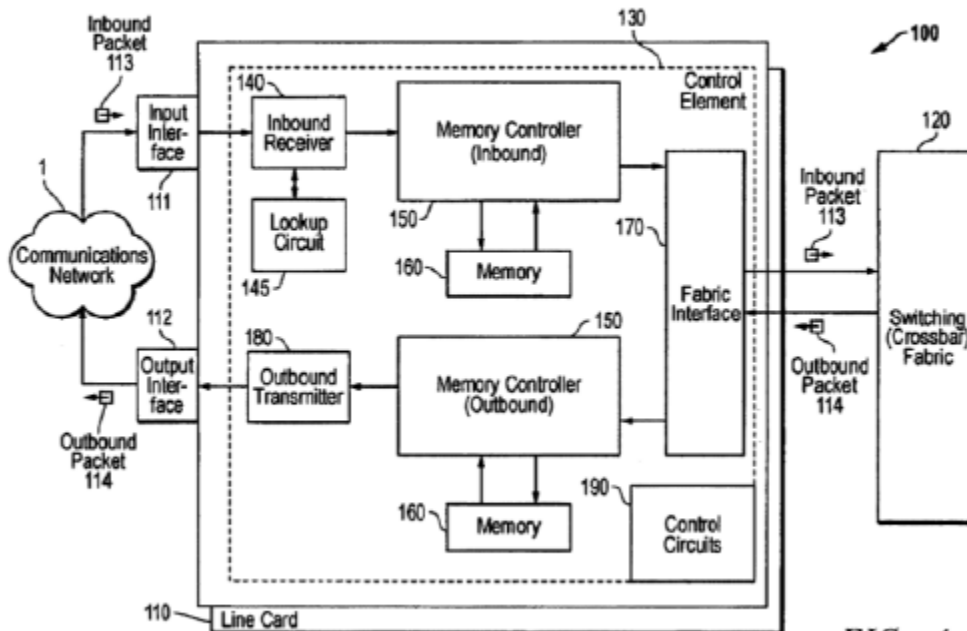
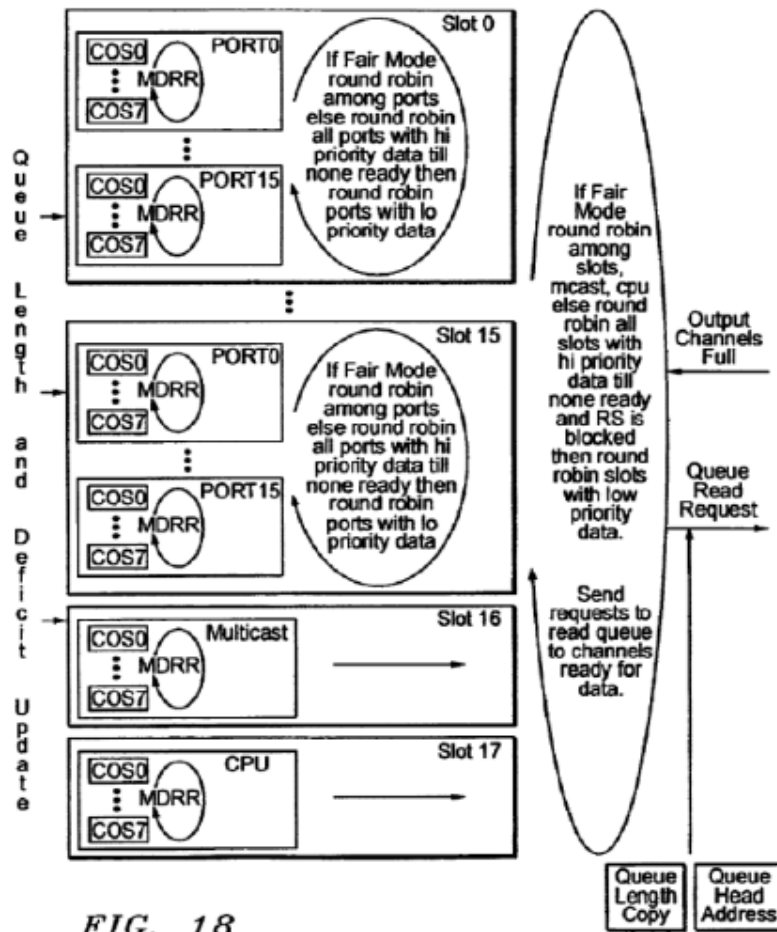


FIG. 1

As shown in figure 1 above, the packet processor on each linecard directs other inbound packets (e.g., those for which the packet processor cannot make a routing determination or those requiring special handling) to the linecard's CPU (the control circuits 190 in figure 1) for processing. Ex. 1108, Abstract; 1:64–2:6, 9:43–48, 52:65 (“Central Processing Unit (Control Circuits 190)”); 53:4–6.

A linecard inspects each inbound packet's header to determine the packet's class of service (CoS) level, which can be one of eight priority levels. Ex. 1108, 1:58–2:6, 9:10–16. As shown in figure 18, reproduced below, each output interface (identified by its slot and port number), as well as the linecard's CPU, has eight associated CoS queues (labeled COS0-COS7 in figure 18) corresponding to the eight different priority levels a packet header can specify. *Id.* at 28:28–31, 28:47–50.



Thus, as shown in figure 18 above, when the packet processor is able to make a routing determination for an inbound packet, the inbound packet is placed, by the linecards' memory controller (MCC), into one of the eight CoS queues for the determined output interface (identified by its slot and port number) and the particular CoS queue into which the packet is placed is the one corresponding to the CoS priority level specified in the packet's header. *Id.* at 17:10–12, 27:55–66, 28:47–50.

For example, when the packet processor is unable to make a routing determination for an inbound packet or the packet requires special handling by the CPU, the inbound packet is placed, by the linecards' memory

controller (MCC), into the appropriate one of the eight “slot 17” CoS queues for the CPU. *Id.* at 9:43–48, 17:10–12, 27:65–66, 28:47–50, 43:48–50.

The linecard’s CPU processes the packets queued in its eight CoS queues according to category, and gives preference to processing the highest priority packets. Ex. 1108, 34:51–36:45, 37:47–50. Wilford describes various techniques for selecting packets from the CPU’s eight CoS queues for processing by the CPU, including a Modified Deficit Round Robin (MDRR) technique in which one of the CoS queues (i.e., CoS queue #7) is “given high priority.” *Id.* at 34:54–67. In a low-delay mode, the MDRR technique ensures that the highest priority queue is given priority in being serviced over the low-priority queues, which guarantees “the lowest possible latency for the high-priority traffic.” *Id.* at 34:63–35:7, 35:53–36:45.

2. Analysis

a. *Claims 1, 8, and 13 - Preamble*

Claim 1 recites, in the preamble, “a network node having one or more packet processors and at least one CPU required to process specific types of packets, a system for managing the specific types of packets for CPU processing.” Claim 8 recites, in the preamble, “[a] network data router capable of categorizing data packets for CPU processing.” Claim 13 recites, in the preamble, “[a] method for processing CPU-destined data packets in a network node, the node having at least one data port for sending and receiving data, and at least one packet processor coupled to the node.” Based on the complete record before us, we are persuaded Petitioner has shown sufficiently that Wilford teaches these preambles of claims 1, 8, and 13.

For example, Petitioner contends an ordinarily skilled artisan would have understood Wilford's routing system 100 is a "*node*" in a communications network, and a "routing system" that routes packets (i.e., sending and receiving data). Pet. 37 (citing Ex. 1108, 1:12–21; 1:31–32). Petitioner contends Wilford's RX ASIC, which includes the inbound lookup unit (LU) (i.e., lookup circuit 225 and lookup memory 227), is a "*packet processor*" as claimed. *Id.* at 38 (citing Ex. 1105 ¶ 201). Petitioner also contends the RX ASIC reads and, via the LU, determines the nature of incoming data packets to routing device 100, *determines* packets "*destined for the CPU*," *categorizes* them, and directly processes other packets when the LU is able to determine their destination directly. *Id.* (citing Ex. 1108, 5:24–30 (the LU "does a destination routing lookup and a source address lookup, in one embodiment by means well-known in the art. . . . Routing information is provided as a set of outputs from lookup circuit 225 to inbound receiver 220 and to rate limiter 230."), 1:60–2:6 (explaining that these "well-known" "routing treatments" executed by the LU can include selecting output interfaces for packets, as well as determining CoS); Ex. 1105 ¶ 201).

Petitioner contends that, like the packet processor described in the '378 patent, Wilford's RX ASIC (including the LU) processes some packets directly by "switching" packets "that have a standard header" to their destinations, and directs other packets to the *CPU for processing*. *Id.* at 39 (citing Ex. 1108, 17:23–28 ("The LookUp Engine (LU) [in the RX ASIC] performs forwarding table lookups on packet headers that it receives from the RX ASIC. LU switches all IPv4 packets that have a standard header and directs all other IP packets to the Linecard processor (LCPU)."), Ex. 1105

¶ 201). Thus, Petitioner contends Wilford’s linecard CPU is required to process *specific types of packets*, i.e., packets requiring CPU intervention, such as packets with IP options, SRP control packets, ICMP packets, tag multicast packets, and exception packets. Pet. 29 (citing Ex. 1108, 5:61–64 (“Special packets, i.e., those with IP options, SRP control packets, ICMP packets addressed to this switch/router, some tag multicast packets and exception packets, are sent to the linecard’s CPU 440 for processing.”), 52:65–53:6 (“The primary tasks of the CPU are: Packet Processing: Any packet needing CPU intervention is transferred to the CPU through the CPU interface module.”), 17:23–28 (“LU switches all IPv4 packets that have a standard header and directs all other IP packets to the Linecard processor (LCPU).”), 7:39–41, 9:43–48, 11:16–45). Patent Owner does not specifically address the preambles. Based on the complete record before us, we are persuaded Petitioner’s contentions have shown Wilford teaches the recited preambles of claims 1, 8, and 13 and we adopt them as own.

b. Claims 1, 8, and 13 - Sorting and Hardware Queues

Claim 1 recites “one or more packet processors enabled to sort the specific types of packets into two or more categories for processing” and “a set of hardware queues of a network access controller (NAC) for queuing the packets according to category.” Claim 8 recites “characterized in that the at least one packet processor sorts packets destined for processing according to two or more categories or priorities into the two or more queues.” Claim 8 also recites “a set of two or more hardware queues of a network controller (NAC) for managing packets destined to the CPU for processing.” Claim 13 recites “sorting the CPU-destined packets into two or more hardware queues of a network controller (NAC) by category.” As explained below, Petitioner

has shown sufficiently that Wilford teaches these limitations of claims 1, 8, and 13.

i. sorting

Petitioner contends Wilford explains that there are eight different CoS levels. Pet. 40 (citing Ex. 1108, 9:10–14 (“[A] queuing system ... provide[s] eight distinct Class of Service (CoS) levels per interface.”); 34:42–47 (“MCC: 2064 queues, including: ... Local CPU: 1 port x 8 CoS/port=8 queues”); 43:49–50 (“One port with 8 COS queues running MDRR is dedicated to local CPU output.”)). Petitioner contends a person of ordinary skill in the art at the time of the invention would have understood these eight CoS levels to be “categories” because packets belong to one of the eight different CoS levels. *Id.* (citing Ex. 1105 ¶ 204).

Petitioner further asserts Wilford’s LU generates a buffer header (BHDR) for each packet which includes information that *categorizes the packets destined for the CPU* according to one of the eight CoS levels. Ex. 1108, 43:49–50 (“One port with 8 COS queues running MDRR is dedicated to local CPU output.”); 28:28–30; Ex. 1105 ¶ 205. Thus, according to Petitioner, the buffer header (BHDR) is used to *sort* the packets into the eight CoS queues for the CPU. *Id.* at 30:55–61; Ex. 1105 ¶ 205. Petitioner contends that CPU-destined data packets for processing using eight CoS queues “are fed into one high and one low priority channel dedicated [and ultimately a high or low priority FIFO/buffer] for the CPU output.” Pet. 41–42 (citing Ex. 1108, 43:49–52, 55–60). We address Patent Owners arguments regarding these limitations below. We adopt as our own Petitioner’s contentions as these limitations recited in claims 1, 8, and 13, with the following exception.

Petitioner states in its Reply that the Petition intended to support the limitation to “the CPU processes the queued packets according to category” in two ways, i.e. “the CPU **both** [1] processes packets in the low-priority channel according to CoS level (in the order selected by the MCC MDRR module from the 7 low15 priority queues), and [2] also process[es] packets in the high-priority channel/FIFO separately from those in the low-priority channel.” Reply 14–15 (citing Pet. 46). We rely only on Petitioner’s contention that processing the high priority channel separately from the low priority channel meets the limitation to sort “according to category.”

Claim 1 reads “characterized in that the CPU processes the queued packets according to category.” Claim 8 requires “the CPU takes [packets] from queues for processing.” Claim 13 recites “selecting packets from the queues for CPU.” Patent Owner asserts that to meet claims 1, 8, and 13, Wilford’s CPU must process, take, and/or select packets directly “as they are emptied from” the CoS queues. PO Resp. 13. We do not read the word “directly” into these claims nor do we find that the claims require the CPU actively take or select packets.

As Petitioner argues, in the ’378 patent’s “preferred embodiment” the NAC, not the CPU, selects packets from the queues and routes the CPU-destined packets to the CPU for processing, and Wilford works the same way. Reply 13 (citing Pet. 44–46). A construction that excludes the preferred embodiment is rarely correct. *See Arista Networks, Inc. v. Cisco Sys., Inc.*, 908 F.3d 792, 797 (Fed. Cir. 2018). Petitioner also argues that Patent Owner’s declarant testifies that “select” in claim 13 does not limit “selecting packets” to being performed by the CPU. Reply 12 (citing Ex. 2001, 204:23–205:9). The Wilford CPU does process, take, and/or

select packets that come from CoS queues that Petitioner alleges are the “hardware queues.” As stated above, CPU-destined data packets for processing using eight CoS queues “are fed into one high and one low priority channel dedicated [and ultimately a high or low priority FIFO/buffer] for the CPU output.” Pet. 41–42 (citing Ex. 1108, 43:49–52, 55–60). Thus, we are not persuaded by Patent Owner’s argument.

Patent Owner further asserts that Wilford does not process packets “according to category.” PO Resp. 13–17. Specifically, Patent Owner argues that

[i]n Wilford, as opposed to the ‘378 Patent, the packets are initially placed into software queues in DRAM by utilizing a queue manager to identify packet destinations utilizing packet headers. See Declaration of Dr. Melendez at ¶49, Wilford at 5:46-54. Once the queue manager determines the destination of the packets, the packets are dequeued and routed to either the fabric switch or to the CPU. The packets that are destined for the CPU are placed into a single queue for processing by the CPU. See Declaration of Dr. Melendez at ¶49, Wilford at 5:55-6:5, Fig. 32. Wilford thus undoes the queuing so that the CPU does not process the hardware queued packets.

PO Resp. 14. In other words, Patent Owner asserts that Wilford undoes the categorization of packets by dequeuing them and requeueing them in a single queue. We disagree.

The use of a low priority queue and a high priority queue is a sort by category according to the claims. Patent Owner suggests that priority is distinct from category. PO Resp. 16. Nevertheless, the Abstract of the ‘378 patent recites sorting packets into “categories of different priority,” apparently equating category and priority or at least indicating that a category can be assigned to a priority just as a particular class of service is assigned to priority in Wilford. Compare Ex. 1101, Abstract, with Ex. 1108,

34:51–36:45; 37:47–50. In particular, figure 3 of the patent recites “CPU processes packets according to priority of categories and classes” and the Specification later refers to categories as “categorical subclasses.” Ex. 1101, 6:53–57, Fig. 3. These citations further suggest class of service is within what was contemplated by the inventors and giving a priority to a class of service was also contemplated. Finally, as noted above in the summary of the ’378 patent, the Specification describes separating packets by category but then associates those categories with priorities. Ex 1101, 5:29–49. Thus, assigning a priority does not undo the categorization of packets.

Patent Owner also asserts that the high and low priority queues being in a “single” queue and identified by separate pointer queues is inconsistent with the claims. PO Resp. 17 (“[T]ransformation of the packets into a singular queue means the CPU is not processing the queued packets ‘according to category.’”). As explained below, we disagree.

It is important to focus on the exact claim language at issue here. Claim 1 recites “process[ing] the queued packets according to category.” Patent Owner does not construe “according to category.” Patent Owner asserts “treating packets as ‘high priority’ and ‘low priority’ is not processing the packets ‘according to category.’” The categories referenced in the claims are the categories that the packets were sorted into by the packet processor.” PO Resp. 16. In Wilford, the packets are sorted into eight CoS categories but are also sorted into one COS category treated as high priority and seven others, collectively treated as low priority. We determine that separating the packets into eight groups then further separating into two groups is consistent with creating “one or more categories” and processing according to those categories. A broad but reasonable reading of this claim

language is that “by category” only requires that one category is processed separate from another. Allowing the high priority and low priority packets to be processed separately, even if the low priority packets come from seven separate queues, would further the ’378 patent inventor’s stated goal of separating packets to preventing CPU overload. *See* Ex. 1001, 2:20–25. Thus, the high priority and low priority packets are in separate queues, even if the low priority packets are in seven unique queues.

Finally, claim 1 recites “sorting the specific types of packets into two or more categories for processing.” This is met by Wilford’s sorting into CoS categories. Claim 1 further recites “queueing packets according to category [in hardware queues].” The eight CoS hardware queues meet this limitation because they are separated into two categories: one high priority CoS queue and seven low priority CoS queues.

Additionally, as noted above, claim 1 does not preclude further queueing prior to processing the packets. In Wilford, the queue packets from the CoS queues are processed according to the categories, i.e. the packets from high priority CoS queues are processed separately from packets from the seven low priority CoS queues.

Claim 8 recites “sort[ing] packets destined for processing according to two or more categories or priorities into two or more queues.” Wilford’s eight CoS hardware queues meet this limitation as they are separated into two categories, i.e., one high priority CoS queue and seven low priority CoS queues. Claim 8 explicitly recites that priorities and categories are interchangeable.

Claim 13 recites “sorting the CPU-destined packets into two or more hardware queues . . . by category.” This is covered by Wilford’s sorting into CoS categories. The eight CoS hardware queues meet this limitation as they are separated into two categories, i.e., one high priority CoS queue and seven low priority CoS queues.

In sum, claims 1 and 8 recite sorting into two or more categories and claim 13 recites sorting into categories, thus none of the claims require more than two categories. Sorting into eight CoS categories and further sorting one of the CoS categories into a high priority category and sorting the other seven into a low priority category meet the claims’ language of sorting into two categories, i.e., low priority and high priority.

ii. hardware queues

Petitioner also contends “Wilford discloses that the CPU’s CoS queues are implemented using hardware configured by software, just as in the ‘378 patent, so the CPU’s CoS queues are hardware queues as claimed.” Pet. 44. Petitioner contends that Wilford explains that the CPU CoS queues (like the other sets of CoS queues) are implemented in the MCC ASIC using DRAM, which is hardware. Pet. 42–43 (citing Ex. 1108, 30:31–40 (“FIG. 14 shows a high-level view of the MCC’s packet handling path. The MCC makes accesses to SDRAM in two separate phases: an Input phase and an Output phase. The input phase is used for writing incoming packets to the SDRAM and the output phase is for reading packets out of the SDRAM.”); 29:49–30:20 (explaining how the queue structures are used to maintain the queue in SDRAM); 31:36–37 (“Cells are dequeued from the Input Cell FIFO and written into DRAM to the appropriate output queue.”); Ex. 1105 ¶ 211).

Further, Petitioner contends Wilford discloses that the MCC's CoS queues are configurable. A "globally configurable value called the 'Read Limit' is used as a target amount of data to read from a queue" (for efficiency of DRAM access). Pet. 43 (citing Ex. 1108, 33:59–61). Additionally, the QUANTUM "programmable value" of each queue is "written by the CPU at configuration time." *Id.* (citing Ex. 1108, 35:21–30; 28:65–67 (explaining that the amount of received packet buffering is configurable)). Petitioner contends a person of ordinary skill in the art at the time of the invention would have understood such disclosures of programmable values, including values written by the CPU, would be performed by software executed by the CPU. *Id.* (citing Ex. 1108, 27:54–28:15 (explaining that "features supported in the MCC ASIC" include "d) Support up to 2064 CoS output queues," "e) CoS up to 8 levels per output port," and "k) Processor interface for slow path packet routing, chip configuration, monitoring, and control.)); Ex. 1105 ¶ 212). We address Patent Owners arguments regarding these limitations below. We adopt as our own Petitioner's contentions as to these limitations recited in claims 1, 8, and 13 with the following exception.

The Petitioner states in the Petition that "[i]f 'hardware queue' is construed more narrowly to require special-purpose hardware circuitry to control at least some part of the queue, the CPU's CoS queues implemented by the MCC are also hardware queues under this narrower construction because these queues are controlled (written and read) by special purpose hardware." Pet. 44. We do not adopt that narrow construction and thus do not rely on the above contention by Petitioner.

We acknowledge that the main thrust of Patent Owner’s argument is that Wilford is similar to the S-Ma reference in IPR2018-00097. In IPR2018-00097, we found that the petitioner had not shown a reasonable likelihood of success in showing that S-Ma taught or suggested “hardware queues” as recited in claims 1, 8, and 13 of the ’378 patent. Ex. 1118, 15. Thus, according to Patent Owner, we must find that Wilford, which is allegedly similar to S-Ma, does not teach or suggest hardware queues in this IPR. PO Resp. 9–12. Patent Owner, relying on the Board’s treatment of the S-Ma reference in IPR2018-00097, argues that Wilford “dynamically creates software queues formed in generic SDRAM.” PO Resp. 10–11. Patent Owner asserts that “Petitioners allege that the use of software queues that are implemented on top of generic RAM memory is hardware, despite the fact that no hardware performs the function of a queue.” *Id.* at 9. Nevertheless, besides citing to S-Ma and IPR2018-00097, Patent Owner does not explain sufficiently what a “software queue” is or why “generic” SDRAM cannot form a hardware queue. Additionally, Patent Owner does not explain sufficiently how Wilford’s queues are allegedly dynamically created or why that would not meet the claim language. Also, Petitioner’s declarant admits that “Dynamic” Random Access Memory (DRAM) is a circuit. Ex. 2001 ¶ 12. Patent Owner does not address sufficiently why the MCC ASIC, in which the queue is implemented, is not hardware.

The Board’s decision in IPR2018-00097 does not require that any reference similar to S-Ma must also lack “hardware queues.” For example, if Wilford discloses “hardware queues” but is also similar or identical to S-Ma, it may be that the petitioner in IPR2018-00097 did not explain the S-Ma

reference sufficiently to meet its burden to show that S-Ma contains “hardware queues.”⁵

Nevertheless, the petition in IPR2018-00097 stated that in S-Ma “the system controller 116 . . . configure[s] the SDRAM 118 to form those queues” (Pet. 33) and/or “a [person of ordinary skill in the art] would have understood that S-Ma’s FE MAC 134 is hardware circuitry that controls and provides a hardware interface to the RSC’s voice and data queues” (Pet. 34). In the first alternative, as described by the petitioner, software (the system controller) is forming (implementing) the queue. In the other alternative, hardware (FE MAC) is controlling and interfacing with the queue but not forming the queue.

In contrast, here, as required by our construction, the Petition recites “Wilford describes how the MCC ASIC *implements* the CoS queues.” Pet. 23. Petitioner explains in its Reply that the Petition shows “the MCC ASIC implements the CoS queues by organizing the SDRAM block into queues and maintaining the head and tail pointers used to write to and read

⁵ We also are not bound by the result in the IPR2018-00097 institution decision because that result was set forth in a non-final institution decision. *See, e.g., Nestle USA, Inc. v. Steuben Foods, Inc.*, No. 2017–1193, slip op. at 1 (Fed. Cir. March 13, 2018) (applying collateral estoppel in claim construction context and noting that “[c]ollateral estoppel protects a party from having to litigate issues that have been *fully and fairly* tried in a previous action and adversely resolved against a party opponent”) (emphasis added). In that case, a preliminary decision was made without a full proceeding and Petitioner did not request rehearing. Importantly, in this IPR, the parties’ arguments refer to a different reference and are more developed—all post-institution briefs have been filed and oral argument has been held. As such, arguments were made in this IPR that were not considered in the ’097 IPR.

from the queues.” Reply 3. Thus, according to Petitioner, “both components that implement the CoS queues – the MCC ASIC and SDRAM – are indisputably hardware” and thus, “the CoS queues are implemented entirely in hardware.” Reply 3. We agree.

Patent Owner also suggests that hardware queues must be implemented using shift registers to be considered a hardware implementation. Sur-Reply 2. Nevertheless, Patent Owner does not specify anything in the Specification of the ’378 patent that would limit hardware queues to shift registers or a specific hardware structure. Thus, we are not persuaded by that argument.

Patent Owner additionally argues Petitioner’s alternate definition that a hardware queue “require[s] special purpose hardware circuitry to control at least some portion of the queue” is not the proper construction. PO Resp. 7. Additionally, Patent Owner argues Wilford does not meet that construction because the Board found in the ’097 IPR that S-Ma’s FE MAC interface did not meet that limitation. PO Resp. 11–12. We do not rely on this alternative construction so these arguments by Patent Owner are moot.

Patent Owner further argues that the alleged hardware queues in Exs. 1110 and 1111 are different from the queues in Wilford. Sur-Reply 2. Petitioner cites to Exs. 1110 and 1111 in its Reply to support its construction of the term “hardware queue” (Reply 6) and to refute Patent Owner’s apparent argument that dynamically created queues cannot be hardware queues (Reply 6–7). We do not rely on Exs. 1110 and 1111, which were not cited or relied on in the Petition, as a basis for our Decision so these arguments by Patent Owner are moot.

c. Claims 1, 8, and 13 - CPU Processing, Selecting, and Taking

Claim 1 recites that “the CPU processes queued packets according to category.” Claim 8 recites that “the CPU takes from queues for processing.” Claim 13 recites “selecting packets from the queues for CPU.” Petitioner has shown sufficiently that Wilford teaches these limitations of claims 1, 8, and 13.

Petitioner contends Wilford’s MCC MDRR module selects from the CoS queues and provides packets to the CPU (via the high- and low-priority channel/FIFOs) and thus, satisfies claim 13’s requirement of “selecting packets from the queues for CPU.” Pet. 46. Petitioner contends Wilford’s CPU 440 receives different categories (priorities) of packets through the two channels/FIFOs, and therefore the CPU both processes packets in the low-priority channel according to CoS level (in the order selected by the MCC MDRR module from the 7 low-priority queues), and also processes packets in the high-priority channel/FIFO separately from those in the low-priority channel. *Id.* Petitioner contends this meets claim 1’s requirement that “the CPU processes queued packets according to category.” *Id.* (citing Ex. 1105 ¶ 219).

Petitioner further contends Wilford’s MCC MDRR module that selects from the CoS queues and provides packets to the CPU, and Wilford’s CPU 440 that processes those packets selected by the MCC, also meet claim 8’s requirement that “the CPU takes from queues for processing.” Pet. 46–47 (citing Ex. 1105 ¶ 220). Petitioner argues that the ’378 patent specification nowhere describes the CPU itself performing the selection of packets from the queues for processing. Pet. 47 (citing Ex. 1105 ¶ 220). Thus, according to Petitioner claim 8 “must be interpreted under BRI to

cover the CPU ‘taking’ packets from the queues for processing by receiving those packets from the NAC after the NAC has selected the packets for processing and sent them to the CPU, as that is what the ‘378 patent specification describes.” Pet. 47 (citing Ex. 1101, 1:54–56; 6:36–38; 6:47–65; 7:39–51; Ex. 1105 ¶ 220. We agree.

Patent Owner’s argument regarding selecting, taking, and processing by category are discussed above in the section regarding the sorting limitation. As discussed above, we are not persuaded by Patent Owner’s arguments in this regard.

d. Claims 2–4, 9, 14, 15

Claims 2–4, 9, 14, 15 of the ’378 patent include limitations to either “the network is the Internet network” or “the network node is one of a packet router, a computer, or an information server.” Patent Owner does not argue claims 2–4, 9, 14, and 15 separately. Petitioner asserts “an obvious use of Wilford’s routing device is as a router in a communications network (e.g., the Internet, a LAN, a WAN, etc.) as Wilford describes an improved linecard, and routers used in such networks in the 2001 timeframe routinely had linecards. Pet. 37–38. We are persuaded by this contention in the Petition and adopt it as our own.

e. Claims 6, 11, and 18

Claims 6, 11, and 18 of the ’378 patent include a limitation that the “queues are assigned priorities and the CPU always processes the highest level of priority regardless of CPU load.”

Petitioner asserts Wilford’s MDRR module has a low-delay mode in which, when processing a set of CoS queues (e.g., COS0-7 queues) for a particular port (PORT0-15) or the CPU, the high-priority queue (e.g., COS7)

is processed whenever it is not empty, and the lower priority queues (e.g., COS0-6) are only processed when there is no data in the high priority queue, thereby guaranteeing “the lowest possible latency for high-priority traffic, but at the expense of starvation of the lower priority traffic.” Pet. 52 (citing Ex. 1108, 34:63–35:7, 35:3–6, 35:65–36:6; Ex. 1105 ¶ 231).

Thus, according to Petitioner, in low-delay mode, Wilford’s MDRR module does not limit the CPU resources available to the highest priority queue (e.g., COS7) and instead processes the highest priority queue until it is empty, so that the CPU always processes this highest level of priority (e.g., from queue COS7) regardless of CPU load as required by claims 6, 11 and 18. We address Patent Owners arguments regarding these limitations below. We agree with and adopt Petitioner’s contentions in the Petition as to claims 6, 11, and 18.

Patent Owner argues that

[i]f the processing of the high-priority and low-priority pointer queues is the processing of the queued packets according to category (as required by the limitation in the independent claims), then the high priority pointer queue and low priority pointer queue are not assigned priorities. However, the Petitioners argue not that the processing of the pointer queues is processing by category, but that the processing of the CoS software queues is the processing according to category and the use of the pointer queues shows the assignment of priorities.

In other words, Patent Owner argues that if the high and low priority are the claimed categories they cannot also be the claimed queue priorities. We disagree. Wilford queues the packets according to category where COS1 is assigned to one queue and the remaining COS2–7 to the remaining queues. It follows that if there are only two queues then the only priorities that can be assigned those categories are a binary choice of high and low. Thus,

Patent Owner has not explained sufficiently why having two categories and then assigning one to a high priority queue and one to a low priority queue would not meet the claim language.

Therefore, we are not persuaded by this argument.

f. Claims 7, 12, and 19

Claims 7, 12, and 19 of the '378 patent require that “each queue can be assigned a rate limit and the CPU observes this limit.” The MCC maintains “a programmable value called the ‘quantum’ and a variable called the ‘deficit’ for each queue.” *Id.* at 35:21–22, 38:51. Wilford uses the quantum and deficit values to control the amount of data that can be read from each queue for a particular DRR round. *Id.* at 35:23–33; 36:24–28, 36:57–37:20. The quantum is written by the CPU at configuration time. *Id.* at 35:29–30; Ex. 1105 ¶¶ 233–34.

For any DRR round, the deficit establishes the maximum number of bytes that can be read for the queue for that round. Ex. 1108, 36:27–28; Ex. 1105 ¶ 235 (citing Ex. 1113). The deficit value is initially set to equal the quantum, which represents the quantum of service credited to the queue for each round of DRR. Ex. 1108, 35:23–25; 36:50–53; Ex. 1105 ¶ 235 (citing Ex. 1113). As bytes are read from a queue during a DRR round, they are subtracted from the queue’s deficit. Ex. 1108, 35:24–28; 35:44–48; 35:61–62. If the deficit becomes less than or equal to zero, the queue drops out of the round. *Id.* at 35:61–62. At the end of the round (e.g., when all queues are either empty or have a nonpositive current deficit), any remaining deficit (which may be negative) is added to the quantum to establish a new deficit value for the next round. *Id.* at 35:30–32; 36:24–28; Ex. 1105 ¶ 235 (citing Ex. 1113). Thus, the new deficit for the next round equals the leftover

deficit from the previous round plus the quantum value. Ex. 1108, 35:30–32; 36:24–28. We address Patent Owners arguments regarding these limitations below. We agree with and adopt Petitioner’s contentions in the Petition as to claims 7, 12, and 19.

Patent Owner asserts “the argument that Wilford implicitly teaches the use of a rate-limit is belied by the fact that the DEFICIT value utilized by the CPU can be set as a negative value, as opposed to a positive value.” PO Resp. 21 (citing Ex. 2001 ¶¶ 58–59). Patent Owner argues further that the quantum value can be updated to take into account the negative deficit amounts from a previous round. *Id.* Thus, according to Patent Owner, “this changes the QUANTUM away from a limiting over time to an adjusted parameter disconnected from the rate leaving the queue.” *Id.*

Petitioner responds:

The reason Wilford’s deficit can go negative is because when reading packets from the queues, the MDDR cannot determine their “exact size” and thus cannot “calculate MDRR deficits in advance of reading the packet” and therefore might read a packet that is larger than expected resulting in the deficit being exceeded for the round. Wilford, 35:35-48. . . . It is undisputed that Wilford’s quantum “results in rates being constrained” (Melendez-Decl., ¶ 59) and limits the average number of bytes read from each CoS queue over time. The claims require nothing more.

Reply 24–25. In other words, Petitioner asserts that the rate limit is broad enough to cover limiting “the average number of bytes read from each CoS queue over time.”

Neither party provides a construction for the term rate limit.

Petitioner’s counsel explained at the hearing “the 378 Patent’s discussion of rate limit is completely agnostic to any particular implementation. It does

not define any special type of rate limit, nor require that the rate limit be implemented in any specific way.” Tr. 29:20–23.

The Specification states, “The load increase then reaches a threshold designed to cause CPU **207** to begin ignoring or discarding some or all of those packets. In this way, CPU **207** of data router **103**, and by implication, the router itself is protected from overload. It is able to continue servicing other categories of packets at reasonable rates.” Ex. 1101, 7:16–22. Thus, the Specification allows for “some or all” packets to be ignored or discarded when reaching a threshold. This is consistent with Wilford, in which the quantum sets a threshold at which some or all of the packets will not be read based on the deficit value. Thus, we are not persuaded by Patent Owner’s argument.

g. Claim 16

Claim 16 of the ‘378 Patent requires an additional step of “configuring the queues and setting a selection protocol for processing by a software component executing on the network node.” Petitioner asserts the protocol implemented by Wilford’s MCC MDRR module to select from the CoS queues is configurable. Petitioner relies on a “code snippet,” in Wilford’s specification, run by the MDRR module, which uses Boolean values to configure processing of the CoS queues in either low-delay-mode or fair-mode. Pet. 55 (citing Ex. 1108, 35:51–55, 36:4–8; Ex. 1105 ¶ 241). We address Patent Owners arguments regarding these limitations below. We agree with and adopt Petitioner’s contentions in the Petition as to claim 16.

Patent Owner argues that “while Wilford does disclose the setting of a selection protocol from these software queues, the selection protocol

disclosed by Wilford is explicitly implemented in hardware.” PO Resp. 22. Further, Patent Owner asserts “in Claim 16, the selection protocol must be running in software, which Wilford does not disclose.” *Id.* at 24.

Neither party offers a construction the claim language in claim 16. We determine that the claim language refers to software “configuring” and “setting a protocol for processing” but does not require the protocol to be running in software or require the software to do the processing of packets which is done by the CPU. In other words, we determine that “setting a protocol for processing” via software does not mean that the processing of the protocol must be by software – only that the “protocol for processing” be set by the software.

This is consistent with the Specification. The Specification allows for an embodiment in which software “SW 104” configures the hardware queues. The Specification recites: “SW 104 in a preferred embodiment is enabled to configure the hardware implemented in the unique NAC to sort and provide packets to the CPU in a variety of ways, described in more detail below.” Ex. 1101, 5:66–68. Also, “in a preferred embodiment, SW 104 as an enhancement is enabled to configure the new queue hardware in NAC 202, providing an interface for configuring the queuing system, such as which and how many queues will be used, what the priority is to be, and the method by which packets are selected from each queue for the CPU.” *Id.* at 6:47–53. In both examples, the Specification states that the software configures the queue, such as setting the method for selection, but does not state that the actual selection or running of the selection protocol is done by the software.

The only other reference to software SW 104 is that: a web server analogous to server 101 of FIG. 1 may be enhanced with a version of SW 104 (FIG. 1) and queues equivalent to queues 209.” *Id.* at 7:54–56. This quotation simply states that a web server may be “enhanced” by SW 104 and explicitly refers to hardware queues 209. Thus, as Petitioner suggests, the Specification is specific that the software is configuring and setting of the method of selection but the Specification does not state that the software runs the protocol. See Reply 26 (The “specification nowhere describes a software component that performs a selection protocol.”)

Patent Owner suggests this is inconsistent with a determination that the algorithm doing the selection is running in hardware. Patent Owner states

the selection protocol disclosed by Wilford is explicitly implemented in hardware. See Petition for Inter Partes Review at 55 (‘For each set of 8 COS queues the hardware performs the following algorithm’), Declaration of Dr. Lavian at ¶240, Declaration of Dr. Melendez at ¶62. The Petitioners rely upon the hardware nature of the selection algorithm to attempt to convince the Board that the queues are hardware queues.

PO Resp. 22–23. Thus, Patent Owner suggests an inconsistency between our analysis of hardware and software for purpose of claim 1 versus claim 16. We do not find such an inconsistency.

As noted above, Petitioner points to a software algorithm in Wilford’s MDRR module that sets a value which instructs the hardware queue to operate in low delay or fair mode. Pet. 55 (citing Ex. 1108, 28:12–13, 35:21-30; Ex. 1105 ¶ 241). We find that this software instruction meets the claim language. According to Petitioner the hardware queue runs in low delay or fair mode. *Id.* This is a selection protocol because low delay places

greater priority on the high priority queue while fair mode does not favor the high priority as much. Pet. 26. Therefore, the hardware queues are running the selection protocol. Thus, we are not persuaded by Patent Owner's argument.

h. Summary

Claims 2–4, 6–7, 9, 11–12, 14–16, and 18–19 depend ultimately from claims 1, 8 and 13. On the complete record before us, we agree that Petitioner has adequately shown Wilford's disclosure of the additional limitations of claims 2–4, 6–7, 9, 11–12, 14–16, and 18–19. Pet. 29–37, 51–56. Accordingly, for the reasons stated above, Petitioner has shown by a preponderance of the evidence that claims 1–4, 6–9, 11–16, and 18–19 would have been obvious over Wilford.

D. Asserted Obviousness over Wilford and Ayres

Petitioner contends claims 5, 10, and 17 are unpatentable under 35 U.S.C. § 103(a) as obvious over Wilford and Ayres. Pet. 56–66. Petitioner explains how Wilford and Ayres allegedly describe all of the claim limitations. *Id.* Patent Owner does not argue claims 5, 10, and 17 separately. PO Resp. 23–24. Ayres is directed to techniques for regulating the flow of data packets in a packet data router that stores received data packets in ingress queues. Ex. 1104, Abstract, 1:20–23.

Claims 5, 10, and 17 recite “a software component for monitoring CPU load, and for configuring the system to select from the queues for processing according to a scheme considering the CPU load.” Petitioner contends that Ayres teaches *software components* “that manage the processing of different priority packet queues in a routing device. Pet. 64–65 (citing Ex. 1104, 2:41–45 (“An IP router is typically controlled with a

real time operating system ('RTOS'), which allows multiple processes of different priorities to co-exist ... (e.g., within a single central processing unit)"); 6:18–49 (flow manager and memory manager are part of the RTOS); Ex. 1105 ¶ 256). Given that the claims broadly recite “considering CPU load,” Petitioner relies on Ayres’ teaching of monitoring router processing resources and other “current operating conditions” and adjusting the rate of processing packets when heavy packet loads occur. Pet. 67 (citing Ex. 1104, 2:64–3:2, 3:17–46, 7:43–8:6). Petitioner also contends that “Ayres discloses monitoring CPU utilization as one non-limiting example of how the load on the CPU may be monitored.” *Id.* (citing Ex. 1104, 3:39–42 (current operating conditions of the router, such as, e.g., such as (sic) processor or memory utilization”); 2:64–3:3; 3:17–23).

Petitioner demonstrates reasons supported by a rational underpinning for combining Wilford and Ayres, including at least that the combination uses known methods to obtain predictable results. Pet. 61–66. For example, Petitioner provides evidence and argument that:

A [person of ordinary skill in the art at the time of the invention] would have understood that monitoring CPU utilization as taught in Ayres is a conventional technique for monitoring CPU load in relation to the CPU’s capacity, and that known technology could be employed in Wilford to monitor utilization of the linecard’s CPU (as Ayres does) and provide that information to the added dynamic packet flow rate control module. (Lavian, ¶259 (discussing Ayres and citing Ex. 1112 at 4:67, 9:62-65, 10:18-20, and 10:26)).

Pet. 66. We agree with and adopt Petitioner’s contentions in the Petition as to claims 5, 10, and 17.

Accordingly, for the reasons stated above, Petitioner has shown by a preponderance of the evidence that claims 5, 10, and 17 would have been obvious over Wilford and Ayers.

III. MOTION TO STRIKE

Petitioner request that Patent Owner’s entire Sur-Reply be struck because it contains allegedly new arguments. Paper 24. Specifically, Petitioner states Patent Owner argued for the first time “a hardware queue” requires some unspecified “structure” “define[d]” by hardware “such as shift registers.” Paper 42, 2 (citing Sur-Reply at 2–3). Petitioner also asserts that Patent Owner, for the first time in “[t]he Sur-Reply[,] offers unsupported attorney arguments about alleged differences between the ‘hardware queues’ in Exs. 1110 and 1111 and Wilford’s queues based on alleged differences between SRAM and DRAM.” Paper 42, 3 (citing Sur-Reply at 2–3). Petitioner asserts Patent Owner’s arguments could have been made in the Patent Owner Response and thus are waived. Paper 8, 3 (arguments not in POR “will be deemed waived”); Update to the Trial Practice Guide (83 Fed. Reg. 38,989 (Aug. 13, 2018) (notifying the public of the updated “Trial Practice Guide” and its accessibility through the USPTO website: <https://go.usa.gov/xU7GP>)) at 15 (a “sur-reply that raises a new issue or belatedly presents evidence may not be considered”; a sur-reply may “respond” to arguments “raised” in the reply but “respond ... does not mean embark in a new direction with a new approach as compared to positions taken in a prior filing.”).

Petitioner’s motion is denied. Patent Owner’s reference to “shift registers” is in response to arguments in the Reply regarding the definition of hardware queue. Additionally, Patent Owner’s reference to Exhibits 1110

and 1111 are in response to Petitioner's reference to those exhibits for the first time in its Reply.

IV. CONCLUSION

For the foregoing reasons, we determine Petitioner has demonstrated, by a preponderance of the evidence, the unpatentability of claims 1–19 of the '378 patent.

V. ORDER

Accordingly, it is

ORDERED that claims 1–19 of U.S. Patent No. 6,738,378 B2 are unpatentable;

FURTHER ORDERED Petitioner's Motion to Strike is *denied*; and

FURTHER ORDERED that parties to the proceeding seeking judicial review of this Final Written Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

IPR2018-00100
Patent 6,738,378 B2

For PETITIONER:

Richard Giunta
Randy Pritzker
Zachary Piccolomini
Anant Saraswat
WOLF, GREENFIELD & SACHS, P.C.
rgiunta-ptab@wolfgreenfield.com
rpritzker-ptab@wolfgreenfield.com
zpiccolomini-ptab@wolfgreenfield.com
asaraswat-ptab@wolfgreenfield.com

For PATENT OWNER:

Douglas Bridges
CAPITAL LEGAL GROUP, PLLC
bridges@capitallegalgroup.com

Gregory S. Donahue
DINOVO PRICE ELLWANGER & HARDY LLP
gdonahue@dpelaw.com