UNITED STATES INTERNATIONAL TRADE COMMISSION

Washington, D.C.

In the Matter of **CERTAIN GRAPHICS SYSTEMS, COMPONENTS THEREOF, AND DIGITAL TELEVISIONS CONTAINING THE SAME**	**Inv. No. 337-TA-1318**

ORDER NO. 31: CONSTRUING TERMS OF THE ASSERTED CLAIMS

(October 18, 2022)

TABLE OF CONTENTS

I. INTRODUCTION 1

II. RELEVANT LAW 2

III. TECHNICAL BACKGROUND AND LEVEL OF ORDINARY SKILL 8

IV. THE ASSERTED PATENTS 9

A. 053 Patent 10

B. 547 Patent 11

C. 381 Patent 13

D. 628 Patent 14

V. CLAIM CONSTRUCTION 15

A. 053 Patent - "arbiter" 15

053 Patent - "select a command thread from either of the plurality of pixel command threads and the plurality of vertex command threads" 15

053 Patent - "relative priorities" 15

B. 053 Patent - "command processing engine" 17

C. 053 Patent - "command threads" 19

D. 053 Patent - "memory device" 20

E. 053 Patent - "coupled to" 21

F. 547 Patent - "coupled to" 22

G. 547 Patent - "wavefront" 24

H. 547 Patent - "event" 25

I. 547 Patent - "event synchronizer" 26

J. 547 Patent - "switch" 28

K. 547 Patent - "wave event generator" 29

L. 547 Patent - "event timestamp module" 29

547 Patent - "client notification module" 30

M. 547 Patent - “shader export module” 32

N. 547 Patent - “shader core” 33

381 Patent - “shader core” 33

O. 381 Patent - “task” 36

P. 381 Patent - “engine”/“plurality of engines” 38

Q. 381 Patent - “load” 41

R. 628 Patent - “unpacked” 42

S. 628 Patent - “the original texture block” 43

T. 628 Patent - “the one or more disjoint subsets” 45

U. 628 Patent - “the approximation” 46

Realtek Ex. 1011, Page 3
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

I. INTRODUCTION

This investigation was instituted by the Commission on June 7, 2022 to determine whether certain graphics systems, components thereof, and digital televisions containing the same infringe U.S. Patent Nos. 7,742,053 ("the 053 patent"), 8,468,547 ("the 547 patent"), 8,760,454 ("the 454 patent"), 8,854,381 ("the 381 patent"), and 11,184,628 ("the 628 patent"). *See* 87 Fed. Reg. 34718-9 (June 7, 2022). The complainants are Advanced Micro Devices, Inc. and ATI Technologies ULC. The respondents are TCL Industries Holdings Co. Ltd., TCL Industries Holdings (H.K.) Limited, TCL Electronics Holdings Limited f/k/a TCL Multimedia Holdings, Ltd., TCL Technology Group Corporation, TTE Corporation, TCL Holdings (BVI) Limited, TCL King electrical Appliances (Huizhou) Co. Ltd., Shenzhen TCL New Technologies Co., Ltd., TCL MOKA International Limited, TCL Smart Device (Vietnam) Co., Ltd., Manufacturas Avanzadas SA de CV, TCL Electronics Mexico, TCL Overseas Marketing Ltd., and Realtek Semiconductor Corp. (collectively "Respondents"). The 454 patent has been terminated from the investigation, so the parties have only presented terms from the remaining patents (together, "the Asserted Patents") for construction. *See* Order No. 10 (July 14, 2022).

Pursuant to the procedural schedule and Order No. 9, the parties filed a joint claim construction chart setting forth certain terms to be construed, and initial and rebuttal claim construction briefs. On September 6, 2022, a videoconference *Markman* hearing took place. On October 5, 2022, a final updated joint claim chart was filed.[1] EDIS Doc. ID 781678.

---

[1] The briefs and amended chart submitted by the parties are hereafter referred to as:

CIMB	BBiTV's Initial *Markman* Brief
CRMB	BBiTV's Rebuttal *Markman* Brief
RIMB	Respondents' Initial *Markman* Brief
RRMB	Respondents' Rebuttal *Markman* Brief
Hr'g Tr.	*Markman* Hearing Transcript
JC	Updated Joint Claim Construction Chart

Realtek Ex. 1011, Page 4
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

II. RELEVANT LAW

"An infringement analysis entails two steps. The first step is determining the meaning and scope of the patent claims asserted to be infringed. The second step is comparing the properly construed claims to the device accused of infringing." *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 976 (Fed. Cir. 1995) (en banc) (internal citations omitted), *aff'd*, 517 U.S. 370 (1996). Claim construction is a "matter of law exclusively for the court." *Id.* at 970-71. "The construction of claims is simply a way of elaborating the normally terse claim language in order to understand and explain, but not to change, the scope of the claims." *Embrex, Inc. v. Serv. Eng'g Corp.*, 216 F.3d 1343, 1347 (Fed. Cir. 2000).

Claim construction focuses on the intrinsic evidence, which consists of the claims themselves, the specification, and the prosecution history. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1314 (Fed. Cir. 2005) (en banc); *see also Markman*, 52 F.3d at 979. As the Federal Circuit in *Phillips* explained, courts must analyze each of these components to determine the "ordinary and customary meaning of a claim term" as understood by a person of ordinary skill in the art at the time of the invention. 415 F.3d at 1313. "Such intrinsic evidence is the most significant source of the legally operative meaning of disputed claim language." *Bell Atl. Network Servs., Inc. v. Covad Commc'ns Grp., Inc.*, 262 F.3d 1258, 1267 (Fed. Cir. 2001).

"It is a 'bedrock principle' of patent law that 'the claims of a patent define the invention to which the patentee is entitled the right to exclude.'" *Phillips*, 415 F.3d at 1312 (quoting *Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). "Quite apart from the written description and the prosecution history, the claims themselves provide substantial guidance as to the meaning of particular claims terms." *Id.* at 1314; *see also Interactive Gift*

Express, Inc. v. Compuserve Inc., 256 F.3d 1323, 1331 (Fed. Cir. 2001) ("In construing claims, the analytical focus must begin and remain centered on the language of the claims themselves, for it is that language that the patentee chose to use to 'particularly point [ ] out and distinctly claim [ ] the subject matter which the patentee regards as his invention."). The context in which a term is used in an asserted claim can be "highly instructive." *Phillips*, 415 F.3d at 1314. Additionally, other claims in the same patent, asserted or unasserted, may also provide guidance as to the meaning of a claim term. *Id*. "Courts do not rewrite claims; instead, we give effect to the terms chosen by the patentee." *K-2 Corp. v. Salomon S.A.*, 191 F.3d 1356, 1364 (Fed. Cir. 1999).

The specification "is always highly relevant to the claim construction analysis. Usually it is dispositive; it is the single best guide to the meaning of a disputed term." *Phillips*, 415 F.3d at 1315 (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)). "[T]he specification may reveal a special definition given to a claim term by the patentee that differs from the meaning it would otherwise possess. In such cases, the inventor's lexicography governs." *Id.* at 1316. "In other cases, the specification may reveal an intentional disclaimer, or disavowal, of claim scope by the inventor." *Id.* As a general rule, however, the particular examples or embodiments discussed in the specification are not to be read into the claims as limitations. *Id.* at 1323. In the end, "[t]he construction that stays true to the claim language and most naturally aligns with the patent's description of the invention will be . . . the correct construction." *Id.* at 1316 (quoting *Renishaw PLC v. Marposs Societa' per Azioni*, 158 F.3d 1243, 1250 (Fed. Cir. 1998)).

In addition to the claims and the specification, the prosecution history should be examined, if in evidence. *Phillips* at 1317; *see Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 913 (Fed. Cir. 2004). The prosecution history can "often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the

invention in the course of prosecution, making the claim scope narrower than it would otherwise be." *Phillips*, 415 F.3d at 1317; *see Chimie v. PPG Indus. Inc.*, 402 F.3d 1371, 1384 (Fed. Cir. 2005) ("The purpose of consulting the prosecution history in construing a claim is to exclude any interpretation that was disclaimed during prosecution.").

When the intrinsic evidence does not establish the meaning of a claim, then extrinsic evidence (*i.e.*, all evidence external to the patent and the prosecution history, including dictionaries, inventor testimony, expert testimony, and learned treatises) may be considered. *Phillips*, 415 F.3d at 1317. Extrinsic evidence is generally viewed as less reliable than the patent itself and its prosecution history in determining how to define claim terms. *Id*. "The court may receive extrinsic evidence to educate itself about the invention and the relevant technology, but the court may not use extrinsic evidence to arrive at a claim construction that is clearly at odds with the construction mandated by the intrinsic evidence." *Elkay Mfg. Co. v. Ebco Mfg. Co.*, 192 F.3d 973, 977 (Fed. Cir. 1999).

If, after a review of the intrinsic and extrinsic evidence, a claim term remains ambiguous, the claim should be construed so as to maintain its validity. *Phillips*, 415 F.3d at 1327. Claims, however, cannot be judicially rewritten in order to fulfill the axiom of preserving their validity. *See Rhine v. Casio, Inc.*, 183 F.3d 1342, 1345 (Fed. Cir. 1999). Thus, "if the only claim construction that is consistent with the claim's language and the written description renders the claim invalid, then the axiom does not apply and the claim is simply invalid." *Id.*

The construction of a claim term is generally guided by its ordinary meaning. However, courts may deviate from the ordinary meaning when: (1) "the intrinsic evidence shows that the patentee distinguished that term from prior art on the basis of a particular embodiment, expressly disclaimed subject matter, or described a particular embodiment as important to the invention"; or

(2) "the patentee acted as his own lexicographer and clearly set forth a definition of the disputed claim term in either the specification or prosecution history." *Edwards Lifesciences LLC v. Cook Inc.*, 582 F.3d 1322, 1329 (Fed. Cir. 2009); *see also GE Lighting Sols., LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1309 (Fed. Cir. 2014) ("the specification and prosecution history only compel departure from the plain meaning in two instances: lexicography and disavowal."); *Omega Eng'g, Inc, v. Raytek Corp.*, 334 F.3d 1314, 1324 (Fed. Cir. 2003) ("[W]here the patentee has unequivocally disavowed a certain meaning to obtain his patent, the doctrine of prosecution disclaimer attaches and narrows the ordinary meaning of the claim congruent with the scope of the surrender."); *Rheox, Inc. v. Entact, Inc.*, 276 F.3d 1319, 1325 (Fed. Cir. 2002) ("The prosecution history limits the interpretation of claim terms so as to exclude any interpretation that was disclaimed during prosecution."). Nevertheless, there is a "heavy presumption that a claim term carries its ordinary and customary meaning." *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002) (citations omitted). The standard for deviating from the plain and ordinary meaning is "exacting" and requires "a clear and unmistakable disclaimer." *Thorner v. Sony Computer Entm't Am. LLC*, 669 F.3d 1362, 1366-67 (Fed. Cir. 2012); *see Epistar Corp. v. Int'l Trade Comm'n*, 566 F.3d 1321, 1334 (Fed. Cir. 2009) (requiring "expressions of manifest exclusion or restriction, representing a clear disavowal of claim scope" to deviate from the ordinary meaning) (citation omitted). As the Federal Circuit has explained, "[w]e do not read limitations from the specification into claims; we do not redefine words. Only the patentee can do that." *Thorner*, 669 F.3d at 1366; *see Continental Circuits LLC v. Intel Corp.*, 915 F.3d 788, 797 (Fed. Cir. 2019); *Unwired Planet, LLC v. Apple Inc.*, 829 F.3d 1353, 1359 (Fed. Cir. 2016) ("[W]e have repeatedly held that it is 'not enough that the only embodiments, or all of the embodiments, contain a particular limitation' to limit claims beyond their plain meaning.").

Courts are not required to construe every claim limitation of an asserted patent. *See O2 Micro Intern. Ltd. v. Beyond Innovation Technology Co.*, 521 F.3d 1351, 1362 (Fed. Cir. 2008) (citations omitted); *Vanderlande Indus. Nederland BV v. Int'l Trade Comm'n*, 366 F.3d 1311, 1323 (Fed. Cir. 2004) (noting that the administrative law judge need only construe disputed claim terms). Rather, "claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims." *Id.* at 1362 (quoting *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997)); *see also Embrex*, 216 F.3d at 1347 ("The construction of claims is simply a way of elaborating the normally terse claim language in order to understand and explain, but not to change, the scope of the claims.") (citation omitted). In addition, "[a] determination that a claim term 'needs no construction' or has the 'plain and ordinary meaning' may be inadequate when a term has more than one 'ordinary' meaning or when reliance on a term's 'ordinary' meaning does not resolve the parties' dispute." *O2 Micro*, 521 F.3d at 1361. Claim construction, however, is not an "obligatory exercise in redundancy." *U.S. Surgical Corp.*, 103 F.3d at 1568. "[M]erely rephrasing or paraphrasing the plain language of a claim by substituting synonyms does not represent genuine claim construction." *C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 863 (Fed. Cir. 2004). However, "there are limits to the court's duties at the claim construction stage. For example, courts should not resolve questions that do not go to claim scope, but instead go to infringement . . . or improper attorney argument." *Eon Corp. IP Holdings v. Silver Spring Networks*, 815 F.3d 1314, 1319 (Fed. Cir. 2016).

A claim must also be definite. Pursuant to pre-AIA 35 U.S.C. § 112(2): "The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention." 35 U.S.C. § 112(2). In *Nautilus, Inc. v. Biosig Instruments, Inc.*, 134 S. Ct. 2120, 2129 (2014), the Supreme Court held that § 112(2) requires "that

a patent's claims, viewed in light of the specification and prosecution history inform those skilled in the art about the scope of the invention with reasonable certainty." A claim is required to "provide objective boundaries for those of skill in the art," and a claim term is indefinite if it "might mean several different things and no informed and confident choice is among the contending definitions." *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1371 (Fed. Cir. 2014). A patent claim that is indefinite is invalid. 35 U.S.C. § 282(b)(3)(A).

The determination as to whether a claim limitation invokes 35 U.S.C. § 112 ¶ 6 or § 112(f) is a two-step process. *Robert Bosch, LLC v. Snap-On Inc.*, 769 F.3d 1094, 1097 (Fed. Cir. 2014). First, one must determine if the claim limitation is drafted in the means-plus-function format. *Id.* In particular the use of the term "means" creates a rebuttable presumption that § 112 ¶ 6 is invoked. *Id.* If "means" is not used, the converse rebuttable presumption arises, but that presumption can be overcome if "the claim term fails to recite sufficiently definite structure or else recites function without reciting sufficient structure for performing that function." *Id.* (internal citation omitted). "The standard is whether the words of the claim are understood by persons of ordinary skill in the art to have a sufficiently definite meaning as the name for structure." *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1349 (Fed. Cir. 2015). "When evaluating whether a claim limitation invokes § 112, ¶ 6, the essential inquiry remains 'whether the words of the claim are understood by persons of ordinary skill in the art to have a sufficiently definite meaning as the name for structure.'" *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1007 (citing *Williamson*, 792 F.3d at 1348).

If it is determined that § 112 ¶ 6 applies, then the "corresponding structure, material, or acts described in the specification" must be identified and the claim term will be given a scope commensurate to the corresponding structures, materials, or acts described, and equivalents thereof. *See Williamson*, 792 F.3d at 1347; *Blackboard, Inc. v. Desire2Learn, Inc.*, 574 F.3d 1371, 1382

(Fed. Cir. 2009). If one is unable to identify any "corresponding structure, material, or acts" in the specification, then the claim term is indefinite. *Id.*; *see Noah Sys., Inc. v. Intuit Inc.*, 675 F.3d 1302, 1312 (Fed. Cir. 2012). "For means-plus-function limitations where the disclosed structure is a computer programmed to implement an algorithm, the patent must disclose enough of an algorithm to provide the necessary structure under 35 U.S.C. § 112 ¶ 6." *In re Aoyama*, 656 F.3d 1293, 1297 (Fed. Cir. 2011) (internal citation omitted). The algorithm may be presented in any "understandable manner, including as a flowchart, so long as sufficient structure is disclosed." *Id.*

III. TECHNICAL BACKGROUND AND LEVEL OF ORDINARY SKILL

The Asserted Patents cover apparatuses, methods, and systems pertaining to graphics processing and parallel computing. The claims of the 053 patent encompass "graphics processing system[s]" comprising hardware components that perform operations involving "pixel command threads" and "vertex command threads." *See* 053 patent at cls. 1, 5. The claims of the 547 patent encompass methods, systems, and computer memory for synchronizing the processing of data in parallel computing. *See* 547 patent at 1:11-22, cls. 1, 16, 22. The claims of the 381 patent encompass apparatuses, systems, and methods "for enabling asynchronous task dispatch," particularly in the processing of graphics data. *See* 381 patent at 1:20, 2:66. The claims of the 628 patent encompass methods and systems for graphics data "decompression" involving "color endpoints." *See* 628 patent at cl. 1.

According to the parties, a skilled artisan would have a degree in electrical engineering, computer engineering, computer science, or a related field, with Respondents specifically identifying a four-year degree in one of these disciplines. *See* CIMB at 1; RIMB at 2. The parties also agree that a skilled artisan would have a minimum of two years of experience, although they disagree on precisely what that experience should be. *See* CIMB at 1; RIMB at 2. The accused products and domestic industry products are seemingly graphics processing units, but the claims of

the 547 and 381 patents are not limited to graphics processing and instead cover parallel computing more generally. Moreover, all the claims, and especially the method claims, recite features that require computer programming, so a skilled artisan would need a familiarity with appropriate software; AMD's proposal of experience in "computer graphics hardware [and] computer architecture" is therefore too narrow. CIMB at 1.

Respondents' experience proposal, by contrast, is more appropriate to the claimed inventions, and is therefore adopted. *See* RIMB at 2. Accordingly, a POSITA would have a bachelor degree in electrical engineering, computer engineering, computer science, or a related field, and at least two years of experience in "parallel computing or graphics processing including developing, designing, or programming software or hardware for graphics processing or parallel processing units, graphics accelerators[,] or other graphics processing systems," with additional education substituting for experience and vice versa. *Id.*

IV. THE ASSERTED PATENTS

The 053 patent, entitled "Multi-Thread Graphics Processing System," issued on June 22, 2010 to Laurent LeFebvre, Andrew Gruber, and Stephen Morein. The 053 patent reports an assignment on its face to ATI Technologies ULC and claims priority to an application filed September 29, 2003.

The 547 patent, entitled "Method and System for Synchronizing Thread Wavefront Data and Events," issued on June 18, 2013 to Laurent LeFebvre, Michael Mantor, and Deborah Lynne Szasz, based on an application filed November 23, 2010. The 547 patent reports an assignment on its face to Advanced Micro Devices, Inc. and ATI Technologies ULC.

The 381 patent, entitled "Processing Unit that Enables Asynchronous Task Dispatch," issued on October 7, 2014 to Michael Mantor and Rex McCrary. The 381 patent reports an assignment on

Realtek Ex. 1011, Page 12
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

its face to Advanced Micro Devices, Inc. and claims priority to a provisional application filed September 3, 2009.

The 628 patent, entitled "Texture Decompression Techniques," issued on November 23, 2021 to Konstantine Iourcha and Andrew Pomianowski. The 628 patent reports an assignment on its face to ATI Technologies ULC and claims priority to an application filed August 31, 2006.

A. 053 Patent

The 053 patent has 9 claims. As of the date of this order, all of claims 1-9 are asserted in this investigation. The asserted claims read as follows (with the first instance of the disputed terms highlighted in bold):

> 1. A graphics processing system comprising:
>
> at least one **memory device** comprising a first portion operative to store a plurality of pixel **command threads** and a second portion operative to store a plurality of vertex command threads; and
>
> an **arbiter**, **coupled to** the at least one memory device, operable to select a command thread from either of the plurality of pixel command threads and the plurality of vertex command threads based on **relative priorities** of the plurality of pixel command threads and the plurality of vertex command threads.
>
> 2. The graphics processing system of claim 1, further comprising:
>
> a **command processing engine**, **coupled to** the **arbiter**, wherein the arbiter is further operable to provide the command thread to the command processing engine.
>
> 3. The graphics processing system of claim 1, wherein the first portion of the at least one **memory device** comprises a first memory device and the second portion of the at least one memory device comprises a second memory device.
>
> 4. The graphics processing system of claim 1, wherein a single **memory device** comprises the first portion and the second portion of the at least one memory device.
>
> 5. A graphics processing system comprising:
>
> at least one **memory device** comprising a first portion operative to store a plurality of pixel **command threads** and a second portion operative to store a plurality of vertex command threads;

Realtek Ex. 1011, Page 13
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

an **arbiter**, **coupled to** the at least one memory device, operable to **select a command thread from either of the plurality of pixel command threads and the plurality of vertex command threads**; and

a plurality of **command processing engines**, coupled to the arbiter, each operable to receive and process the command thread.

6. The graphics processing system of claim 5, wherein the plurality of **command processing engines** comprises at least one arithmetic logic unit.

7. The graphics processing system of claim 5, wherein the plurality of **command processing engines** comprises at least one texture processing engine.

8. The graphics processing system of claim 5, wherein the first portion of the at least one **memory device** comprises a first memory device and the second portion of the at least one memory device comprises a second memory device.

9. The graphics processing system of claim 5, wherein a single **memory device** comprises the first portion and the second portion of the at least one memory device.

B. 547 Patent

The 547 patent has 22 claims. As of the date of this order, claims 16-21 are asserted in this investigation. The asserted claims read as follows (with the first instance of the disputed terms highlighted in bold):

16. A system for synchronizing one or more thread wavefronts and associated events, comprising:

a **shader core** comprising a plurality of processing units configured to execute a first **wavefront** outputting a first data;

a **wave event generator coupled to** the shader core and configured to generate a first **event** associated with the outputting of the first data;

an **event synchronizer** configured to release the first event before releasing events inserted subsequent to the first event;

a **switch** coupled to the shader core and a memory, and configured to:

insert the first event into the event synchronizer; and

release the first event from the event synchronizer after the first data is stored in the memory.

17. The system of claim 16, wherein the **switch** is a crossbar switch, comprising:

Realtek Ex. 1011, Page 14
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

a plurality of ring buffers, each corresponding to a respective memory controller;

a plurality of interfaces to the **shader core**, wherein each respective interface to shader core is configured to write to any one of the plurality of ring buffers; and

a plurality of interfaces to the respective memory controllers.

18. The system of claim 17, wherein the crossbar **switch** further comprises:

an **event timestamp module** configured to:

maintain an event timestamp counter associated with **events** inserted into the **event synchronizer**;

attach an event timestamp from the event timestamp counter to the first event before inserting into the event synchronizer;

attaching the event timestamp to respective ones of the first data before writing to switch buffers;

comparing timestamp values of the first event and the first data in the switch buffers; and

determining, based on the comparison, completion of storing the first data to the memory.

19. The system of claim 17, wherein the crossbar **switch** further comprises:

a **client notification module** configured to:

determine one or more clients to be notified of at least one of the first **event** or one or more processing events; and

notify the one or more clients of the at least one of the first event or the one or more processing events.

20. The system of claim 17, wherein the crossbar **switch** is further configured to:

request one or more memory controllers **coupled to** outputs of the switch for acknowledgement messages when the one or more memory controllers have completed storing of the first data to the memory; and

determine completion of storing the first data to the memory based on the acknowledgment messages received from the one or more memory controllers.

21. The system of claim 16, further comprising:

one or more **shader export modules** coupled between the **shader core** and the **switch**, including:

Realtek Ex. 1011, Page 15
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

a plurality of shader export data buffers configured to store the first data received from the shader core;

a shader export **event synchronizer** configured to:

insert a shader export first **event** associated with the first data into a shader export event synchronizer, wherein the shader export event synchronizer is configured to release the shader export first event before releasing events inserted into the shader export event synchronizer after the first event; and

release the shader export first event from the shader export event synchronizer after the first data is sent to the switch.

C. 381 Patent

The 381 patent has 27 claims. As of the date of this order, claims 15-20 are asserted in this investigation. The asserted claims read as follows (with the first instance of the disputed terms highlighted in bold):

15. A computing system comprising:

a memory;

a first processing unit; and

a bus coupled to the memory and the first processing unit, wherein the first processing unit comprises:

a **plurality of engines** configured to receive, from a scheduling module associated with a second processing unit, a plurality of **tasks** and to **load** state data associated with each of the plurality of tasks; and

a **shader core** configured to receive the plurality of tasks from at least one of the plurality of engines and to execute a first task from the plurality of tasks while executing a second task from the plurality of tasks based on respective state data associated with each of the first and second tasks.

16. The computing system of claim 15, wherein the **shader core** comprises:

a first plurality of processing elements configured to execute the first **task** based on a first set of state data associated with the first task; and

a second plurality of processing elements configured to execute the second task based on a second set of state data associated with the second task.

Realtek Ex. 1011, Page 16
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

17. The computing system of claim 15, wherein the **plurality of engines** comprises a first queue configured to receive the first **task** and a second queue configured to receive the second task.

18. The computing system of claim 15, wherein the first **task** comprises a low-latency task and the second task comprises a regular-latency task.

19. The computing system of claim 15, wherein the first **task** comprises a graphics-processing task and the second task comprises a general-compute task.

20. The computing system of claim 15, wherein the **shader core** is configured to receive the plurality of **tasks** based on a priority associated with each of the plurality of tasks.

D. 628 Patent

The 628 patent has 12 claims. As of the date of this order, claims 7-12 are asserted in this investigation. The asserted claims read as follows (with the first instance of the disputed terms highlighted in bold):

7. A computer processing system comprising:

a processor and a memory, wherein the processor is configured to:

receive a compressed texture block, wherein the compressed texture block includes two or more disjoint subsets into which data in the compressed texture block is to be **unpacked**, wherein the two or more disjoint subsets include a first subset having a first set of color endpoints and a second subset having a second set of color endpoints; and

individually decompress each of the two or more disjoint subsets to form texels, wherein the decompressing includes:

determining a first texel color for a first texel that is part of the first disjoint subset by interpolating between color endpoints of the first set of color endpoints based on a first index value for the first texel; and

determining a second texel color for a second texel that is part of the second disjoint subset by interpolating between color endpoints of the second set of color endpoints based on a second index value for the second texel.

8. The computer processing system of claim 7, wherein the compressed texture block represents **the original texture block**.

9. The computer processing system of claim 7, wherein **the one or more disjoint subsets** have different sizes.

Realtek Ex. 1011, Page 17
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

10. The computer processing system of claim 7, wherein the processor is further configured to independently decompress each of **the one or more disjoint subsets**.

11. The computer processing system of claim 7, wherein the processor is further configured to apply **the approximation** to a rendered surface for display.

12. The computer processing system of claim 7, wherein each of **the one or more disjoint subsets** is associated with a quality parameter that met a pre-selected threshold.

V. CLAIM CONSTRUCTION

The disputed claim terms are summarized in the parties' Joint Claim Construction Chart. *See* JC. They are discussed individually below.

A. 053 Patent - "arbiter"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Component that actively selects one of a plurality of command threads based on state bits associated with each command thread.

053 Patent - "select a command thread from either of the plurality of pixel command threads and the plurality of vertex command threads"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Actively choosing between a pixel command thread and a vertex command thread based on state bits associated with each command thread.

053 Patent - "relative priorities"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Priority based on whether a command thread is a pixel command thread versus a vertex command thread.

JC at 1.

Independent claim 5 of the 053 patent recites a memory device for storing a plurality of "pixel command threads" and a plurality of "vertex command threads," as well as an "arbiter" that is "operable to select a command thread from either of the plurality of pixel command threads and the plurality of vertex command threads." 053 patent at 8:5-12. Independent claim 1 recites the same elements, and adds the limitation that the arbiter must make the selection "based on relative priorities of the plurality of pixel command threads and the plurality of vertex command threads." *Id*. at 7:11-21.

Respondents contend that the arbiter's selection must be based on "state bits associated with each command thread." JC at 1. The term "state bits" is not found in any claim of the 053 patent, and the intrinsic evidence otherwise does not support Respondents' position. One disclosed embodiment teaches that "bits [are] required for the thread arbitration," but the specification explains that such bits may be either "state bits," which are stored in a "one read port device," or "status bits," which are stored in a "multi-read port device." 053 patent at 4:49-56. And a different disclosed embodiment teaches that the arbiter "may always retrieve the oldest available thread," but it is not clear that the age of any particular thread is indicated solely by a state bit. *Id*. at 3:35-36. So Respondents have not shown that the arbiter necessarily "requires 'state bits' in order to perform its arbitration scheme" in all disclosed embodiments. RIMB at 8.

Respondents also contend that for claim 1, the "relative priorities" of the "plurality of pixel command threads and the plurality of vertex command threads" refers to whether the command thread is "either a 'vertex' priority or a 'pixel' priority." RIMB at 11. But the claim itself requires selection of "a [single] command thread" from among at least four such threads, two (or more) pixel threads and two (or more) vertex threads, so the selection must be based at least in part on considerations other than merely the "command thread type." *Id*. at 11; 053 patent at 7:16. And as

noted, in one embodiment the selection is based on thread age, without regard to thread type. 053 patent at 3:35-36. So this contention, too, is not supported by the intrinsic evidence.

Respondents further contend that the arbiter must "actively select[]," rather than merely "select," the command thread. JC at 1. To be sure, during prosecution the applicant traversed a prior art rejection over a reference called Airey by arguing that "'select' means 'to take as a choice from among several; pick out,'" and that Airey did not disclose "active 'selection.'" RIMB, Ex. B at 7-8. But the plain and ordinary meaning of "select" presupposes some degree of data processing or computational activity; an arbiter that simply stores the command threads "in a display list," without routing them, as in Airey, would presumably not "select" anything and so would not satisfy the claim language. *Id*. The 053 patent's subsequent *inter partes* review proceeding has little bearing on this issue, because there Complainant again argued that the definition of "select" was "pick out," while the Patent Trial and Appeals Board adopted a construction of "arbiter" – "any computer hardware, software, or combination thereof that receives and provides a command thread" – that does not limit how the selection is made. *LG Electronics, Inc. v. ATI Technologies ULC*, IPR2015-00325, Final Written Determination at 6-8 (Apr. 14, 2016) ("053 FWD"), *reversed on other grounds*, *ATI Technologies ULC v. Iancu*, 920 F.3d 1362 (Fed. Cir. 2019). Respondent's contention that "arbiter" has no plain and ordinary meaning is similarly irrelevant, because the dispute is not over what an arbiter is, but over what it does, which is "select a command thread." *See* RIMB at 7-8. And there appears to be no dispute that that language has a plain and ordinary meaning.

B. 053 Patent - "command processing engine"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: engine for processing each command thread selected by and received	Software or hardware module that processes commands.

from the arbiter.	

JC at 1.

In claims 2, 5, 6, and 7, once the arbiter selects a command thread, the command thread is transmitted to a "command processing engine," or a plurality of such engines, and in claim 5 the plurality of engines "process the [selected] command thread." AMD proposes that this term be given its plain and ordinary meaning, and alternatively that it be construed as an "engine for processing each command thread selected by and received from the arbiter." JC at 1. This construction is too narrow for claim 2, however, because that claim requires only that the arbiter "provide the command thread to the command processing engine," with no requirement that the command processing engine do anything with it. 053 patent at 7:24-26. This construction is also too narrow for claim 5 and its dependent claims, because claim 5 only requires that the command processing engine "receive and process the command thread," not that it receive the thread "from the arbiter." *Id*. at 8:13-15.

So AMD's alternative construction is rejected. Beyond that, the parties' dispute appears to be over whether the command processing engine may be hardware or software implemented on hardware (or presumably firmware, as well) (Respondents' position) or is limited to hardware (AMD's position). *See* RRMB at 4; CIMB at 8-9. The intrinsic evidence does not conclusively resolve this dispute. The specification states that the command processing engine "may be any suitable engine," but does not disclose whether or not an engine must be hardware. 053 patent at 2: 2:59-62. AMD did argue that a command processing engine could be software in the 053 patent's IPR proceeding, but that argument was not germane to the dispute, which was over whether all claimed command processing engines had to be capable of processing all types of command threads, and in any event Respondents offer no basis to find disavowal, such that AMD cannot advance a

more precisely tailored construction in the present investigation. *See* RRMB at 4; RIMB, Ex. C at 10-11; 053 FWD at 8-12.

So the construction of this term requires extrinsic evidence. AMD has offered such extrinsic evidence, including an opinion from an expert, but Respondents have not. *See* CIMB at 9. Resolution of this dispute would be aided by additional evidence, particularly expert evidence from Respondents, and construction of this term is therefore deferred.

C. 053 Patent - "command threads"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: (1) streams of instructions or processes that are parts of a larger process or program or (2) a sequence of commands applicable to the corresponding element.	A sequence of commands, instructions, and state bits required to process a vertex or pixel element.

JC at 1.

Respondents urge a particular construction for "command thread(s)," while AMD urges that it be accorded its plain and ordinary meaning, and in the alternative one of two particular constructions. *See* JC at 1. AMD's second alternative construction is the definition of the term in the specification, and so is well supported by the intrinsic evidence. *See* 053 patent at 2:41-42 ("a command thread is a sequence of commands applicable to the corresponding element").

The parties' dispute, however, is not over the meaning of the term as such, but is instead over whether "state bits" are a necessary component of a command thread. *See* RIMB at 13; RRMB at 5; CIMB at 10; CRMB at 4. There is a suggestion in the specification that status bits, at least, may be part of a command thread in one embodiment. *See* 053 patent at 5:9-10 ("all of the above status bit fields from the command threads go to the arbitration circuitry"). Nonetheless, the intrinsic evidence shows that state bits are not necessarily part of a command thread, for two reasons. First, the definition of "command thread" in the specification only mentions commands, not state bits,

status bits, or instructions. *See* 053 patent at 2:41-42. Second, as explained above, it is not clear from the specification that state bits are needed for arbitrating all embodiments, and so it cannot be said they are necessary components of command threads in all embodiments, either.

Lastly, although AMD's second alternative construction was not timely proposed, as Respondents correctly observe, Respondents have been aware of the specification's definition of the term for some time, and even quoted it in their initial *Markman* brief. *See* JC at 1 n.1; RIMB at 13. Adopting it would therefore not unduly prejudice Respondents.

Accordingly, "command threads" is construed as "sequence of commands applicable to the corresponding element," with no requirement that they comprise state bits.

D. 053 Patent - "memory device"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Memory with dedicated storage areas for pixel and vertex command threads.

JC at 1-2.

Independent claims 1 and 5 recite "at least one memory device" comprising a first portion operative to store the pixel command threads and a second portion operative to store the vertex command threads. 053 patent at 7:11-14, 8:5-9. Some dependent claims require that the first and second portions be separate memory devices, and other dependent claims require that the first and second portions be part of a single memory device. *Id*. at cls. 3, 4, 8, 9.

The plain and ordinary meaning of "memory device," as well as the associated "portions," would seem to be clear, even to a layperson. Respondents nonetheless propose a construction, "memory with dedicated storage areas for pixel and vertex command threads," that is inappropriate for two reasons. First, it would mean that the embodiment described in Figure 2 is not within the scope of the claims because the command storage device (reservation station 202) stores generic

"graphic commands" which presumably could be either pixel or vertex command threads. *See* RIMB at 14-15; RRMB at 6. But as AMD points out, the specification describes "reservation station 202 operat[ing] similar to a [FIFO] memory device," and otherwise does not disclose any particular memory storage structure. *See* CRMB at 5 (citing 053 patent at 3:4-6). So nothing in the specification prohibits the embodiment of Figure 2 from falling within the scope of the claims. Second, the term "dedicated" implies either that a particular memory portion only stores a particular type of command thread, or that the memory portion cannot be programmed to store other data – as AMD puts it, that the memory in the device cannot be "dynamically allocated." CRMB at 5. But nothing in the claims or the specification requires this, so there is no basis in the intrinsic evidence to limit the memory device to possessing "dedicated" storage areas.

Therefore, Respondents' proposed construction of "memory device" is rejected, which resolves the parties' dispute, and the term is otherwise not construed.

E. 053 Patent - "coupled to"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Directly connected.

JC at 2.

The claims require an arbiter "coupled to" the memory device(s), and for some claims a processing engine or engines "coupled to" the arbiter. *See* 053 patent at cls. 1, 2, 5. The nature of the connection differs from claim to claim: for claim 2, the arbiter must "provide the command thread" to the processing engine; for claim 5, the processing engines must "receive and process" the command thread, but not necessarily directly from the arbiter; and for claim 1, there is no processing engine required, so there is necessarily no connection required, either. *See id*. Moreover, no other language in any claim suggests a "direct" connection between the memory device(s) and the arbiter is present or required.

"Coupled to" must therefore be construed broadly, so that it has the same meaning in each claim. *See Phillips*, 415 F.3d at 1315 ("claim terms are normally used consistently throughout the patent"); *In re Varma*, 816 F.3d 1352, 1363 (Fed. Cir. 2016) ("But the principle that the same phrase in different claims of the same patent should have the same meaning is a strong one, overcome only if 'it is clear' that the same phrase has different meanings in different claims."). Respondents nevertheless argue that "coupled to" requires a direct connection. *See* JC at 2. But there would seem to be no barrier to, say, inserting a second memory device or buffer between the claimed memory device and the arbiter, such that the arbiter "select[s] a command thread" from the data stored in the second memory device, rather than directly from the claimed memory device, because what is important is simply that there is communication between the elements 053 patent at cls. 1, 5; *see* RIMB at 16 ("the arbiter is coupled to the memory so that it can select a thread from the memory"). This is consistent with the breadth of claim 5 which, as noted, explicitly states that the processing engines receive the command thread, but leaves open from where (*e.g.*, arbiter or other component). Furthermore, it is irrelevant that every embodiment described in the specification teaches a direct connection because, again, the claim language itself is not so limiting. *See* RIMB at 16-17; *Kamstrup A/S v. Axioma Metering UAB*, 43 F.4th 1374, 1383-4 (Fed. Cir. 2022) (refusing to read several limitations from embodiments into claims); *Sunoco Partners Marketing & Terminals L.P. v. U.S. Venture, Inc.*, 32 F.4th 1161, 1176 (Fed. Cir. 2022) (holding plain language of claim was nonetheless broader than specification's "preferably" statement).

Therefore, Respondents' proposed construction of "coupled to" is rejected, which resolves the parties' dispute, and the term is otherwise not construed.

F. 547 Patent - "coupled to"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Directly connected.

JC at 2.

Claim 16 of the 547 patent requires a "wave event generator" that is "coupled to" a "shader core," and a switch "coupled to" the shader core and to a memory. 547 patent at 17:8-15. Dependent claim 21 further requires "one or more shader export modules coupled between the shader core and the switch." *Id*. at 18:18-19. Respondents contend that in every instance, "coupled to" means "directly connected." JC at 2.

Respondents' contention is refuted by the descriptions of multiple embodiments in the specification. Figure 8 shows a set of shader cores 810 (called a "shader complex" in the Figure) and a switch 802, connected only by two shader export modules 804a and 804b. *See* 547 patent at Figure 8, 10:53-59. Similarly, Figure 9 shows a block diagram of a shader export module, with an upstream shader core interface 902 and downstream "crossbar switch" interface 912; there is no description of any other connections between the shader core and the switch. *See id*. at Figure 9, 13:15-19. Lastly, Figure 10 shows a block diagram of a switch, with upstream shader export module interfaces 1002a and 1002b, and other interfaces, including a memory controller interface 1010 and event interface 1020, but no interface directly to any shader core. *See id.* at Figure 10, 13:61-64, 14:51, 14:64.

In short, at least three disclosed embodiments, at least of claim 21, show only an indirect connection between the shader core and the switch, and "coupled to" thus cannot be limited to direct connections. Therefore, Respondents' proposed construction of "coupled to" is rejected, which resolves the parties' dispute, and the term is otherwise not construed.

G. 547 Patent - "wavefront"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: a plurality of identical concurrent threads that are executed on separate processors.	A plurality of identical concurrent threads that are executed on separate processors.

JC at 2.

Claim 16 covers a system that includes a "shader core comprising a plurality of processing units configured to execute a first wavefront outputting a first data." 547 patent at 17:3-7. The specification states that a "thread wavefront" is a "plurality of identical concurrent threads that are executed on separate processors." *Id*. at 1:20-22. AMD has offered expert evidence that the term "wavefront" may have a plain and ordinary meaning that is both broader and narrower than this, specifically, that a wavefront involves threads executed "together," "in lockstep," or "in parallel," as opposed to threads that are "concurrent" but not necessarily executed concurrently, and that the threads are not necessarily "identical." CIMB, Ex. 5 at ¶ 52. AMD's expert also opined that the 547 patent's use of "wavefront" is in any event consistent with its plain meaning, and AMD has offered as an alternative to adopt Respondents' proposed construction. *See* CIMB, Ex. 5 at ¶ 53; JC at 2.

Although the terms "wavefront" and "thread wavefront" may have the same meaning, AMD expressly declined to concede that point at the *Markman* hearing, and its expert did not opine on the subject, so even assuming that there has been lexicography for the latter term, it is unclear if that lexicography also applies to the former term, as Respondents contend. *See* Hr'g Tr. at 59:23-61:17; CIMB, Ex. 5 at ¶¶ 51-53. Indeed, given such a record, it is not clear that the parties actually disagree about the term's meaning. If there is such a disagreement, it will have to be clarified and resolved later; in the meantime, the term is not construed.

H. 547 Patent - "event"

AMD's Construction	Respondents' Construction
token or other message that can be sent between modules, or communicated to other threads, wavefronts, or processing modules, to enforce synchronization, flush caches, report status to host applications, indicate the completion of the execution of a particular wavefront, report status of the thread wavefront to the command processor, and the like, or for other predetermined purposes. Compromise construction: "event" a token or other message that can be sent between modules to enforce synchronization, flush caches, report status to host applications, or for other predetermined purposes.	A token or other message that is sent between modules to indicate the completion of the execution of a wavefront. The token or other message may also be used to enforce synchronization, flush caches, report status to host applications, or for other predetermined purposes. Compromise construction: "First event" – a token or other message that is sent between modules to indicate the completion of the execution of a wavefront (i.e., a wave event). Other "events" – a token or other message used to enforce synchronization, flush caches, report status to host applications, or for other predetermined purposes.

JC at 2-3.

During the Markman hearing it became clear that the parties had not reached a meeting of the minds regarding what needs to be construed in connection with "event." *See* Hr'g Tr. at 73:16-23, 76:20-77:25. The dispute appears to be over whether the claimed "first event" is merely a species of the larger genus of "event," or is instead entirely distinct from "event," and if so, how "first event" should be construed. *See id*. But that dispute was not adequately briefed by the parties, so it cannot be resolved at present.

Therefore, resolution of the parties' dispute over the construction of "event" and "first event" is deferred.

Realtek Ex. 1011, Page 28
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

I. 547 Patent - "event synchronizer"

AMD's Construction	Respondents' Construction
Means-plus-function; Function: release the first event before releasing events inserted subsequent to the first event; Structure: logic hardware, comprising one or more FIFO queues, and equivalents thereof; See also event.	Means-plus-function; Function: release the first event before releasing events inserted subsequent to the first event; Structure: one or more first in first out FIFO buffers (4:58-63).

JC at 3.

Claim 16 recites an "event synchronizer configured to release the first event before releasing events inserted subsequent to the first event." 547 patent at 17:11-13. Respondents have offered expert evidence that the term "event synchronizer" connotes a function, but not any particular structure, and the parties agree that such language recites a means-plus-function term under 35 U.S.C. § 112 (6) (pre-AIA). *See* CIMB at 17; RIMB at 20-23; *see also Personalized Media Comms., LLC v. Int'l Trade Comm'n*, 161 F.3d 696, 704 (Fed. Cir. 1998) (presumption that Section 112(6) does not apply to a term that does not use "means" may be rebutted when the "evidence so warrant[s]"). The parties further agree that the function is "release the first event before releasing events inserted subsequent to the first event." JC at 3.

The parties offer similar descriptions of the corresponding structure as well, with both referring to first-in-first-out (FIFO) activity. This is entirely reasonable given that "release the first event before releasing events inserted subsequent to the first event" is essentially the definition of FIFO.

Respondents define the structure as a FIFO buffer, as opposed to a FIFO queue, "because a FIFO 'queue' would not necessarily be able to accomplish the claimed function of releasing a first event *after* data is stored in memory." RIMB at 22 (citing RIMB, Ex. A at ¶ 28). This is immaterial, however. The agreed function, stated above, does not refer to releasing events after data is stored

in memory—only that the structure "release the first event before releasing events inserted subsequent to the first event." JC at 3. Respondents also argue "logic hardware," as in AMD's proposal, has no meaning and "has never been found to be corresponding structure for a means-plus-function limitation." RRMB at 9-10. While this may be true, this particular patent strongly implies "FIFO event queues" are themselves structure. It states, "[t]he event synchronizer can comprise one or more event synchronizing structures. According to an embodiment, event synchronizer comprises FIFO event queues in one or more shader export modules and a FIFO event queue in a switch, such as a crossbar switch." 547 patent at 4:41-45. The patent continues, "[a]ccording to an embodiment, an ordering of events is enforced such that other events that occurred later than the wave event corresponding to the first thread wavefront are not released until after the wave event is released. For example, the event synchronizer can be implemented as one or more first-in-first-out (FIFO) queues." *Id.* at 4:54-59. These are clear links in the specification between FIFO queues and the claimed function. *Noah Systems, Inc. v. Intuit Inc.*, 675 F.3d 1302, 1311 (Fed. Cir. 2012) ("[T]he patent's specification must provide 'an adequate disclosure showing what is meant by that [claim] language.'").

Overall, then, the intrinsic evidence strongly associates "FIFO queue" with structure and the activity of an event synchronizer, and this is done without any reference to "logic hardware" which Respondents find objectionable. And AMD, for its part, does not dispute the accuracy of Respondents' choice of "FIFO buffer," only calling it "overly restrictive" without elaboration. *See* CIMB at 17; CRMB at 7-8; *see generally* RIMB at 21-22 (citing 547 patent at 6:62-67, 7:36-5, 8:53-9:6, 11:60-12:4, 13:38-64); 547 patent at 14:40-42 ("The released events can be temporarily held in a temporary event buffer 1012. The temporary event buffer 1012 can be a FIFO buffer or can enforce another queue discipline.").

Accordingly, the term "event synchronizer" is interpreted as a means-plus-function element, with "release the first event before releasing events inserted subsequent to the first event" as the function, and with "FIFO queue" or "FIFO buffer" as the corresponding structure and equivalents thereof. *See* 35 U.S.C. § 112 (6) (such terms shall be "construed to cover the corresponding structure . . . and equivalents thereof").

J. 547 Patent - "switch"

AMD's Construction	Respondents' Construction
Plain and Ordinary Meaning.	Means-plus-function; Function: insert the first event into the event synchronizer; and release the first event from the event synchronizer after the first data is stored in the memory; Structure: none; indefinite

JC at 3.

The term "switch" in claims 16 through 21 is presumptively not subject to means-plus-function claiming because it does not use the word "means." *Personalized Media*, 161 F.3d at 704. And because the term "switch" very clearly connotes structure even to a layperson, Respondents' extrinsic evidence on this point fails to rebut the presumption. *See* RIMB, Ex. A at ¶¶ 29-32. To be sure, the term may present invalidity issues under Section 112, including for enablement and written description, because the role the "switch" plays is wide-ranging. *See* RIMB at 23-25. It must, depending on the claim: (1) insert the first event into the event synchronizer (claim 16); (2) release the first event from the event synchronizer (claim 16); (3) be a crossbar switch comprising a plurality of ring buffers (claim 17); (4) comprise an event timestamp module (claim 18); (5) comprise a client notification module (claim 19); and (6) be configured to "request . . . acknowledgement messages" (claim 20). Exactly how a single "switch" might perform all these functions is unclear.

But the threshold issue presented by Respondents is whether the claimed "switch" connotes structure, regardless of the number and variety of functions it performs, and Respondents have not rebutted the presumption that it does. So the term is not subject to Section 112(6). This finding resolves the parties' dispute, so the term is otherwise not construed.

K. 547 Patent - "wave event generator"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: logic hardware configured to generate events.; If subject to 35 U.S.C. 112 ¶ 6: Function: generate a first event associated with the outputting of the first data; Structure: logic hardware and equivalents thereof; See also event.	Means-plus-function; Function: generate a first event associated with the outputting of the first data; Structure: none; indefinite.

JC at 3.

Respondents' expert opines that the claimed "wave event generator" does not have a "sufficiently definite meaning as the name for structure." RIMB, Ex. A at ¶ 33. But the lack of "means" establishes a presumption that the term is not subject to Section 112(6). *See Personalized Media*, 161 F.3d at 704. And although AMD's expert does not squarely assert that the term is structural, he does opine that a "generator" is a "module or device that initiates a bus request," and that a "wave event generator" is "logic hardware configured to generate events," and AMD's position is that such a plain and ordinary meaning connotes structure. CIMB, Ex. 5 at ¶¶ 65-71; *see* CIMB at 19. This dispute presents a classic "battle of the experts," and so cannot be resolved without additional evidence. Construction of this term is therefore deferred.

L. 547 Patent - "event timestamp module"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning. If subject to 35 U.S.C. 112 ¶ 6: Function: maintain an event timestamp counter associated with events	Means-plus-function; Function: maintain an event timestamp counter associated with events

inserted into the event synchronizer; attach an event timestamp from the event timestamp counter to the first event before inserting into the event synchronizer; attaching the event timestamp to respective ones of the first data before writing to switch buffers; comparing timestamp values of the first event and the first data in the switch buffers; and determining, based on the comparison, completion of storing the first data to the memory Structure: logic hardware of the crossbar switch, including a counter, and equivalents thereof; See also event.	inserted into the event synchronizer; attach an event timestamp from the event timestamp counter to the first event before inserting into the event synchronizer; attaching the event timestamp to respective ones of the first data before writing to switch buffers; comparing timestamp values of the first event and the first data in the switch buffers; and determining, based on the comparison, completion of storing the first data to the memory. Structure: none; indefinite.

547 Patent - "client notification module"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning. If subject to 35 U.S.C. 112 ¶ 6: Function: determine one or more clients to be notified of at least one of the first event or one or more processing events; and notify the one or more clients of the at least one of the first event or the one or more processing events; Structure: logic hardware of the crossbar switch and equivalents thereof; *see also* event.	Means-plus-function; Function: determine one or more clients to be notified of at least one of the first event or one or more processing events; and notify the one or more clients of the at least one of the first event or the one or more processing events.; Structure: none; indefinite.

JC at 3-4.

Dependent claim 17 specifies that the "switch" of claim 16 is a crossbar switch possessing certain components. Claim 18 depends from claim 17 and further specifies that the crossbar switch comprises an "event timestamp module" configured to perform five distinct functions. 547 patent at 17:28-41. Claim 19 also depends from claim 17 and specifies that the crossbar switch comprises a "client notification module" configured to perform two functions. *Id*. at 17:42-18:6.

As with the term "switch," there may well be Section 112 issues with these two "module" terms. It is unclear how a crossbar switch, even with added "modules," could perform the functions

of, say, "attaching the event timestamp to respective ones of the first data before writing to switch buffers" (claim 18), or "notify[ing] the one or more clients of the at least one of the first event or the one or more processing events" (claim 19), while also performing the functions of the switch recited in independent claim 16 and possessing the specific structure recited in claim 17.

But the threshold question presented is whether these module terms should be construed as subject to 35 U.S.C. § 112(6). *See* RIMB at 28-31. Respondents are correct that the term "module" has often been construed as a synonym for "means," and their expert opines that neither "event timestamp module" nor "client notification module" connote structure. *See id*. (citing *Williamson*, 792 F.3d at 1350); RIMB, Ex. A at ¶¶ 35, 38. Again, though, in the absence of the word "means" Section 112(6) presumptively does not apply. *Personalized Media*, 161 F.3d at 704. And as AMD observes, what is claimed is not merely a "module" in the abstract but a crossbar switch (a hardware component having a very specific structural design) possessing ring buffers and two different sets of interfaces (also hardware structures), as required by claim 17, to which are added the modules of claims 18 or 19. *See* CIMB at 24, 26. That is, the modules must be hardware within the larger "switch" structure, and not, as Respondents' expert opines, implemented via software. *See* RIMB, Ex. A at ¶¶ 35, 38. This conclusion is bolstered by the specification, which describes a counter (a hardware structure) as part of the event timestamp module. 547 patent at 14:22.

On balance, therefore, Respondents have not rebutted the presumption that Section 112(6) does not apply to the terms "event timestamp module" and "client notification module," and the terms are not subject to Section 112(6). This finding resolves the parties' dispute, so these terms are otherwise not construed.

M. 547 Patent - "shader export module"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: logic hardware configured to store data output from the shader core and transfer data to the switch.	Means-plus-function; Function: insert a shader export first event associated with the first data into a shader export event synchronizer, wherein the shader export event synchronizer is configured to release the shader export first event before releasing events inserted into the shader export event synchronizer after the first event; and release the shader export first event from the shader export event synchronizer after the first data is sent to the switch.; Structure: none; indefinite.

JC at 4.

In contrast to the two preceding "module" terms, the "shader export module" of claim 21 is a distinct component "coupled between the shader core and the switch," and so cannot be analyzed the same way as the modules of claims 18 and 19. 547 patent at 18:18-19. And because it contains two recited elements, one of which, "a plurality of shader export data buffers," connotes structure, the overall shader export module cannot be construed under mean-plus-function rules. *Id*. at 18:20.

The other element, however, is a "shader export event synchronizer," which appears to perform the same functions as the more general "event synchronizer" of claim 16, but to do so prior to sending any data to the claimed switch. *Compare* 547 patent at 17:11-18 ("event synchronizer" functions) *with id*. at 18:22-31 ("shader export event synchronizer" functions). The "event synchronizer" was construed as a means-plus-function term, in part because AMD agreed to it, so it is unclear why AMD has not agreed to the same treatment for "shader export event synchronizer." *See* CIMB at 27-29. As for Respondents' arguments, it is unclear why they overlook the clearly structural "shader export data buffer" element, and instead treat the entirety of the "shader export module" as a mean-plus-function term, with the recited function limited to just the function of the

"shader export event synchronizer." *See* JC at 4. Similarly unclear is why they do not accept the "shader export event FIFO"/"event FIFO" (which may well be a buffer) as the corresponding structure, as a parallelism to their identifying the "FIFO buffer" as the structure corresponding to the "event synchronizer" of claim 16. *See* 547 patent at 12:15-19, 13:34-37.

In view of the disconnect between the claim language and the parties' positions, and the lack of explanation for why the "shader export event synchronizer" should be construed so differently from the "event synchronizer," further briefing is required. Construction of this term is therefore deferred.

N. 547 Patent - "shader core"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: logic hardware configured to execute graphics shader programs to process data to provide a graphic output. Compromise proposal of the plain and ordinary meaning as stated in AMD's Rebuttal Brief: An array of processing elements and associated components in a GPU, capable of executing various threads (e.g., shader programs) of a single graphics-processing task.	A plurality of processing units and associated components for concurrently executing threads of a sequence of wavefronts.

JC at 5.

381 Patent - "shader core"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: graphics processing portion of the first processing unit circuitry comprising a plurality of processing elements for execution of various threads (e.g., shader programs).	An array of processing elements and associated components for executing threads.

Compromise proposal of the plain and ordinary meaning as stated in AMD's Rebuttal Brief: An array of processing elements and associated components in a GPU, capable of executing various threads (e.g., shader programs) of a single graphics-processing task.	

JC at 4-5.

The term "shader core" appears in the asserted claims of both the 547 and 381 patents. Although the two patents do not share a specification, they are in the same art and the definition of a skilled artisan is the same for both, so the term presumptively has the same plain and ordinary meaning in both patents. According to the 547 patent, a shader core comprises "a plurality of processing units," with claim 16 specifying a certain configuration (547 patent at 2:12-14, 17:5-7), and according to the 381 patent a shader core is "an array of processing elements" (381 patent at 1:60-61). These descriptions are not precisely the same, so the plain and ordinary meaning differs slightly between the two patents, but they are nonetheless similar enough that there is no clear lexicographic intent.

Notwithstanding this straightforward analysis, the parties have proposed five different constructions for this one term. *See* JC at 4-5. As an initial matter, the parties add unspecified "associated components" to three of the five proposed constructions, but they offer no reason why components "associated" with the processing units/elements would be part of the shader core, nor do they identify the components. *Id*. Otherwise, Respondents' proposals enjoy more support from the intrinsic evidence. For the 547 patent, Respondents propose a "plurality of processing units," but add "concurrent[]" execution and executing threads of "a sequence of wavefronts." *Id*. But the claim itself speaks of a single "wavefront," rather than a sequence, and nothing in the claim language

or the specification requires concurrent or parallel processing. 547 patent at cl. 16. Respondents also contend that the "shader core" may be purely software, but it clearly cannot be, because, again, it "compris[es] a plurality of processing units." *See* RRMB at 16. For the 381 patent, Respondents propose "an array of processing elements," but add "and associated components for executing threads." *Id*. at 4. But the execution requirement is already incorporated into the claim language (albeit for a "task," rather than a "thread"), so there is no need for it in construing "shader core." *See* 381 patent at 14:8-16.

As for AMD's proposals, first, although processing units and elements are forms of "logic hardware" (*see* CIMB at 29), that term is not found in the specification and encompasses a very large number of components, including integrated circuits of many different types and various logic circuits that may be inappropriate for the sort of processing the patent contemplates. Next, AMD's proposed limitation to processing "threads" is either already incorporated into the claim language (in the 547 patent, at least through the preamble) or expressly focused on "tasks" rather than "threads" (as in the 381 patent).

Lastly, AMD's proposals seek to limit the shader core to graphics applications. *See* JC at 4-5; CIMB at 30. For the 547 patent, AMD has offered expert evidence that the plain and ordinary meaning of "shader core" requires that it be used for graphics processing, rather than being "simply a general-purpose processor." CIMB, Ex. 5 at ¶¶ 85-90. Moreover, the term "shader" implies that the invention has some visual aspect, such as application to graphics processing, which is plainly the preferred embodiment. But the claim language otherwise omits any reference to graphics, and the specification states that the method of processing threads to output a result may be used for "protein folding" analysis, "encryption/decryption," and "video encoding/decoding." 547 patent at 1:11-12. The specification also uses the term "shader core" multiple times without explicit reference

to graphics processing, or otherwise suggests that graphics processing is just one possible embodiment. *E.g.*, *id.* at 2:11-13 ("[t]he shader core comprises a plurality of processing units configured to execute a first wavefront outputting a first data"); 4:1-2 ("a shader core having a plurality of shader processors"). And the descriptions of the shader export module (*id*. at Figure 9) and switch (*id*. at Figure 10), both of which receive inputs from the shader core, do not reference pixels, vertices, or other graphics terminology at all. *See id*. at 13:9-14:64. The same is true for the 381 patent: "embodiments of the present invention will be described herein in terms of a [graphics processing unit, but] embodiments of the present invention may be applied to other types of processing units." 381 patent at 4:62-66; *see id*. at 3:55-56 ("the invention is not limited to the specific embodiments described herein"). In fact, dependent claim 19 requires a configuration to execute both a "graphics-processing" task and a "general-compute" task, which rules out a limitation to graphics processing. *Id*. at 14:32-34.

On balance, therefore, the term "shader core" is construed as "a plurality of processing units" in the 547 patent and "an array of processing elements" in the 381 patent.

O. 381 Patent - "task"

AMD's Construction	Respondents' Construction
Unit of processing work, comprising one or more threads and/or shader programs, as provided by the scheduling module to an engine. See also engine.	Application, or portion thereof, sent by the CPU that includes a priority.

JC at 5.

The shader core executes tasks received from the engines, which in turn receive those tasks from a "scheduling module." 381 patent at 14:6-16. Although "task" has a plain and ordinary meaning, both sides add limitations that are unsupported by the record.

AMD's expert offers three dictionary definitions, the most reasonable of which is "[a] sequence of instructions treated as a basic unit of work by the supervisory program of an operating system." CIMB, Ex. 6 at ¶ 79. Based in part on this, the expert opines that a POSITA would understand "task" to mean a "unit of processing work, comprising one or more threads and/or shader programs, as provided by the scheduling module to the engine." *Id*. at ¶ 76. That the scheduling module provides the tasks to the engines, however, is part of the claim language and so is not part of the meaning of "task." *See* 381 patent at 14:6-8. Nor need the task necessarily comprise threads or shader programs, because the specification states that those are optional: "[t]o perform a single graphics-processing task, a GPU may execute a plurality of different threads (sequence of instructions) [which] may comprise a shader program." *Id*. at 1:49-52.

Respondents' expert agrees that a task does not comprise a thread, although a task may "later be convert[ed] . . . into one or more (GPU) threads." RIMB, Ex. A at ¶ 48. Respondents' expert further opines that a "task" is a "portion[] of an end-user application or API that is sent to a co-processor or GPU to be executed (accelerated)," and that it comes from "a processing unit (e.g., a CPU)." *Id*. at ¶ 46. Inasmuch as the scheduling module is a processing unit, limiting the source of the task is already written into the claim language, and so it too should not be part of the definition of "task." Moreover, Respondents' expert does not say that a task is the application itself; rather, the task is a "portion" of an application, which presumably means a subset of the operations needed to run the application. In short, the parts of Respondents' proposed construction involving an "application" in its entirety and limiting the task to being "sent by the CPU" are not supported by the record.

So the plain and ordinary meaning of "task" is either a "unit of processing work" or a "portion of an application to be executed," possibly including a priority. The first definition is not

expressly supported by the intrinsic evidence, but is also not inconsistent with it. The second definition is better supported by the specification, which states, for instance, that tasks are "generated by" applications. *E.g.*, 381 patent at 9:13-38.. Either one of these definitions would seem to qualify as the plain and ordinary meaning of "task."

As for priority, independent claim 15 requires that the engines "load state data associated with the plurality of tasks" and that the shader core execute tasks "based on respective state data associated with" the tasks, while dependent claim 20 requires that the shader core be configured to "receive the plurality of tasks based on a priority associated with each of the plurality of tasks." 381 patent at 14:8-16, 14:35-37. So a priority is needed for claim 20 but presumably not for claim 15 (or the other dependent claims) (*Phillips*, 415 F.3d at 1314-5), and a task need not "include a priority" in any event, because the priority must merely be "associated with" each task. Furthermore, not all disclosed embodiments require a priority, the prosecution history on which Respondents rely pertains to claim language that expressly recited that each task has an associated priority, and that same prosecution history does not limit the definition of "task" to require "includ[ing] a priority." *See* 381 patent at 3:35-37 ("the plurality of tasks are executed . . . based on the set of state data"); RIMB, Ex. D at 1. Respondents' contention that a priority must be included is accordingly rejected. *See* RIMB at 42.

Therefore, the term "task" is construed as "a unit of processing work or portion of an application to be executed," and it need not include a priority.

P. 381 Patent - "engine"/"plurality of engines"

AMD's Construction	Respondents' Construction
engine: portion of the first processing unit circuitry, configured to cause one or more tasks to be independently executed on a shader core that is shared with other engines.	plurality of engines: Two or more virtual modules that are each associated with a particular task received from the operating system.

plurality of engines: see engine. Compromise construction: hardware component of the first processing unit, which may be assisted by software to perform its functions, and is configured to dispatch one or more tasks to be independently executed on a shader core that is shared with other engines, wherein for software implementations, the software enables: (i) using and/or interfacing with the engine; or (ii) defining the design and/or fabrication of the engine (e.g., in hardware description languages).	

JC at 5-6.

The parties have proposed essentially three constructions for "engine" and "plurality of engines," which are terms in claims 15 and 17. *See* JC at 5-6. All proposed constructions are rejected either as meaningless or as inconsistent with the intrinsic evidence.

The hottest dispute between the parties is whether the engines are at least "circuitry" or "hardware" (AMD's position) or are instead purely "virtual," that is, software (Respondents' position). *See* Hr'g Tr. at 103:12-15, 108:16-22. Neither side is correct, because the specification is unequivocal on this point: "[e]mbodiments of the present invention may be implemented using hardware, software or a combination thereof." 381 patent at 10:59-60. The specification then describes hardware, software, and combination embodiments. *Id*. at 10:62-12:43. That most of the disclosed embodiments are virtual rules out AMD's proposed construction, but that at least one disclosed embodiment is not virtual similarly rules out Respondents', and Respondents cite nothing in the prosecution history that rules out a hardware implementation (*see* RRMB at 18-19).

AMD proposes other limitations that fare no better. As explained during the *Markman* hearing, the shader core must "receive" tasks from the engines, but the engines do not "cause" tasks

to be executed on the shader core. *See* 381 patent at 14:10; Hr'g Tr. at 106:2-15. In response to questioning during the *Markman* hearing, AMD suggested the engines instead "send" such tasks, and its compromise construction uses the word "dispatch." *See* Hr'g Tr. at 106:16-107:7. "Send" and "dispatch" are more clearly consistent with the claim language, but they add no meaningful content to the requirement that the shader core "receive" the tasks from the engines. Next, AMD's proposal that the shader core "is shared with other engines" is refuted by the claim's recitation of a shader core that receives tasks "from at least one of the plurality of engines"; that is, the claim requires a plurality of engines, but encompasses a shader core that is connected to only one engine. 381 patent at 14:11. As for AMD's proposal that the tasks must be "independently executed," the claim does require execution by the shader core, but the language in the specification on which AMD relies appears to refer to the independence of the engines: "[e]ach virtual engine is configured to receive [tasks] and independently execute these tasks on the single shader core." *Id*. at 4:48-51. Lastly, AMD's compromise proposal regarding what the "software enables" was not briefed, is not supported by the intrinsic evidence, and is bizarre, in that the claims cover a "computing system" including a plurality of engines, not software for "defining the design and/or fabrication of the engine" itself.

Respondents' proposed construction, in using the expression "two or more," clarifies nothing, because that is simply the definition of "a plurality." Reclassifying the engine as a "module" does an equal amount of nothing, and although the specification teaches an embodiment where "different virtual engines . . . process different types of tasks," the shader core "receive[s] [a] plurality of tasks from at least one of the plurality of engines." 381 patent at 7:2-4, 14:10-11. So again, although a plurality of engines is required, the claim is not limited to one task per engine.

In sum, to the extent the dispute over "engines"/"plurality of engines" centers on the hardware/software distinction, the claim is not limited to either implementation. Otherwise, the parties' proposed constructions are not well taken, and any further construction is deferred.

Q. 381 Patent - "load"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning.	Extract and store.

JC at 6.

The plurality of engines recited in claim 15 must be "configured to receive . . . a plurality of tasks and to load state data associated with each of the plurality of tasks," and the shader core must be configured to receive the tasks, but not necessarily the state data, from the engines. 381 patent at 14:6-16. The claim does not require that the engines load the state data into their own memories (as opposed to some other component or components), it does not require the engines (as opposed to some other component) to first extract the state data, and it does not specify how or from which component the shader core receives the state data. And although "store" may be one plain and ordinary meaning of "load," loading data does not necessarily require it to be stored. "Storing" data implies that it is placed in some sort of memory, but the data could, consistent with the claim language, be loaded into one component and then immediately transferred into another, with no "storage" except in a trivial sense. Indeed, in one disclosed embodiment the data used by the shader core's "processing elements 220" is "stor[ed]" in "local data shares (LDS) 228," which are described as part of the shader core. 381 patent at 8:23-25. There is no direct connection, however, between the engines and the shader core, so the storage in the LDS cannot be the result of a "load" by the engines.

Respondents' argument that "load" should be construed as "extract and store" is therefore not well taken. Because AMD urges only the plain and ordinary meaning of "load," rejecting

Realtek Ex. 1011, Page 44
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687

Respondents' proposed construction resolves the parties' dispute, and the term is otherwise not construed.

R. 628 Patent - "unpacked"

AMD's Construction	Respondents' Construction
Extracted.	Extracting information from the compressed texture block that is required to decompress the data stored in the subset.

JC at 6.

Claim 7 of the 628 patent covers a processor configured to receive a "compressed texture block" that includes two or more disjoint subsets "into which data in the compressed texture block is to be unpacked." 628 patent at 17:2-7. The parties agree that "unpacking" such data involves "extracting" it, but disagree on whether further limitations are required. *See* JC at 6.

Respondents are correct that the specification provides an express meaning for "unpacking." *See* RIMB at 45 (citing 628 patent at 8:53-56). Adopting that meaning, however, seemingly changes the meaning of the claim, from "two or more disjoint subsets into which data in the compressed texture block is to be unpacked," to "two or more disjoint subsets into which information from the compressed texture block that is required to decompress the data stored in the subsets is to be extracted." The actual claim language requires unpacking or extraction of "data in the compressed texture block" into the (presumably empty) subsets, while Respondents' proposed construction requires unpacking or extraction of "information from the compressed texture block" needed to decompress the data (presumably already "stored in the subsets").

Furthermore, Respondents' proposed construction is inconsistent with Figure 5: (1) in step 505, the processor receives the compressed texture block; (2) in step 510, the partition and the disjoint subsets used to compress the block of data are determined; (3) in step 515, the data "for each subset" is unpacked; and (4) in step 520, the processor decompresses the data in each subset.

See 628 patent at 8:38-62, Fig. 5. Importantly, steps 510 and 515 are omitted from the claim language, that is, claim 7 does not expressly require that the claimed processor determine the partition and disjoint subsets, and then extract the compressed data and store it in each subset for decompression at step 520. Claim 7 does, however, require that the compressed texture block "is to be unpacked," and unpacked "into" the subsets, such that prior to the claimed unpacking the subsets must be empty. Thus, even though the claimed invention need not actually perform the unpacking – presumably another processor or processors may carry out that step – unpacking the compressed data into the subsets must necessarily take place before the processor decompresses "each of the . . . subsets." *Id.* at 17:11-12.

So the specification's statement that "'[u]npacking' refers to extracting the information from the block that is required to decompress the data stored in the subset" does not qualify as lexicography, as Respondents contend. 628 patent at 8:54-56; *see* RIMB at 45. It makes little sense when applied to extracting the texture data, as opposed to other data, it uses the word "refers" rather than "means," or "is defined as," and it appears in reference to step 515, which need not even be performed by the invention of claim 7. On balance, there has been no "clear [] . . . definition of the disputed claim term" or clear "intent to redefine the term." *Luminara Worldwide, LLC v. Liown Electronics Co. Ltd.*, 814 F.3d 1343, 1353 (Fed. Cir. 2016).

Instead, AMD's proposal, that "unpacked" simply means "extracted," is consistent with the rest of the claim language and with the specification. Therefore, the term "unpacked" is construed as "extracted."

S. 628 Patent - "the original texture block"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: the texture block before being compressed into the compressed texture	Indefinite.

block.	

JC at 6.

In dependent claim 8, the compressed texture block "represents the original texture block." 628 patent at 18:4-6. Respondents contend that the term is indefinite. *See* JC at 6; RIMB at 45-47; RRMB at 23-24. To be sure, it would seem that the whole point of decompression is to recover the original texture block, such that the scope of claim 8 may well be the same as the scope of claim 7, from which it depends, which would appear to introduce an ambiguity into the claim. *See* RIMB at 46.

But the only ambiguity Respondents argue is whether "the original texture block" lacks antecedent basis. *In re Downing*, 754 F.App'x 988 (Fed. Cir. 2018), is directly on point. In *Downing*, the claim in suit recited "the end user," without an antecedent reference to "an end user." *Id*. at 990. The Federal Circuit noted that a lack of antecedent basis need not result in indefiniteness when the claim apprises a skilled artisan of its scope, observed that "claim 1's recitation of one end user could only refer to the end user using the product," and held that the claim was thereby not indefinite. *Id*. at 996. The same is applicable here: the claim covers decompression of compressed texture data, and a skilled artisan would surely recognize that "the original texture block" refers to the texture data before partition and compression. Certainly this is what the specification suggests. *See* 628 patent at 8:27-30 ("the original block of pixels is subdivided . . . [and] compressed to form a compressed texture block"), 8:57-59 ("the subsets are independently decompressed to generate an approximation of the original texture block").

Therefore, the term "the original texture block" is not indefinite for lack of antecedent basis. Because AMD principally urges only the plain and ordinary meaning of the term, rejecting

Respondents' indefiniteness argument resolves the parties' dispute, and the term is otherwise not construed.

T. 628 Patent - "the one or more disjoint subsets"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: the two or more disjoint subsets.	Indefinite.

JC at 6.

During prosecution, and in response to an obviousness rejection, what issued as claim 7 was amended to change the number of disjoint subsets from "one or more" to "two or more." CIMB, Ex. 8 at 4. The dependent claims, however, were not similarly amended, and as issued those claims still refer to "the one or more" disjoint subsets. *Id*. at 6-7. Claim 7 was also amended to add the elements involving decompressing to form first and second "texels." *Id*. at 5. The explanation offered to overcome the obviousness rejection was that the cited prior art did not disclose either "two or more" disjoint subsets or decompression to form texels. *Id*. at 9-12.

AMD argues that the discrepancy between independent claim 7 and dependent claims 9, 10, and 12 is due to typographical errors. *See* CIMB at 47-49. But even if true, this is beside the point, because unlike a District Court, the Commission lacks the statutory authority to correct even obvious typographical errors. *See Novo Industries, L.P. v. Micro Molds Corp.*, 350 1348, 1357 (Fed. Cir. 2003) (district courts may "correct obvious minor typographical and clerical errors" in patents, but not "major errors"); *see, e.g.*, *Certain Variable Speed Wind Turbines and Components Thereof*, Inv. No. 337-TA-641, Comm'n Op. at 36 (Mar. 2, 2010) ("The Commission lacks the authority to correct inventorship under Section 256 or any other statutory provision, and the Commission's authority in this regard must be conferred by statute.").

Moreover, it is not clear that the error is merely typographical. As it stands, the dependent claims both lack antecedent basis and are broader than the independent claim. Had the applicant changed "one or more" to "two or more" in the dependent claims, which is the correction AMD urges, both indefiniteness problems would have been solved. But as noted, the applicant made two amendments at the same time, either of which would seem to have been sufficient to overcome the prior art on their own. So had the applicant rewritten the dependent claims in independent form, the broader "one or more disjoint subsets" as in the original independent claim could have been retained, while simultaneously avoiding indefiniteness and overcoming the prior art due to the additional elements. *See Otsuka Pharm. Co., Ltd. v. Sandoz, Inc.*, 678 F.3d 1280, 1296 (Fed. Cir. 2012) (holding obviousness of dependent claims need not be considered if independent claim found non-obvious). Because the intrinsic evidence does not reveal which of these two corrections the applicant hypothetically would have applied, "rewrit[ing] the claim to preserve its validity" is not feasible in this instance, even assuming that the Commission has the authority to do so. *Hill-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1374 (Fed. Cir. 2014).

As written, then, a skilled artisan examining the intrinsic evidence would not be able to discern whether the claims 9, 10, and 12 require "two or more" disjoint subsets, or permit just "one or more." Claims 9, 10, and 12 of the 628 patent are accordingly indefinite.

U. 628 Patent - "the approximation"

AMD's Construction	Respondents' Construction
Plain and ordinary meaning, in the alternative: the estimation of the original block of pixels formed by decompressing the compressed texture block, including using interpolation.	Indefinite.

JC at 7.

The parties' dispute over "the approximation," a term in dependent claim 11, is essentially the same as their dispute over "the original texture block," namely, Respondents contend it is indefinite for lack of antecedent basis. *See* RIMB at 47-48. The resolution is essentially the same, as well: under *Downing*, 754 F.App'x at 996, a skilled artisan would surely recognize that "the approximation" refers to the decompressed texture data that "approximat[es] . . . the original texture block." 628 patent at 8:58-59.

Therefore, the term "the approximation" is not indefinite for lack of antecedent basis. Because AMD principally urges only the plain and ordinary meaning of the term, rejecting Respondents' indefiniteness argument resolves the parties' dispute, and the term is otherwise not construed.

SO ORDERED.

Cameron Elliot
Administrative Law Judge

Realtek Ex. 1011, Page 50
Realtek Semiconductor Corp., v. Advanced Micro Devices, Inc.
IPR2023-00687