Trials@uspto.gov
571-272-7822

Paper No. 21
Entered: April 29, 2019

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MEDIATEK INC. and MEDIATEK USA INC., and
ARM LTD. and ARM, INC.,
Petitioners,

v.

ADVANCED MICRO DEVICES, INC. and ATI TECHNOLOGIES ULC,
Patent Owner.

Case IPR2018-00102
Case IPR2018-01149[1]
Patent 7,633,506 B1

Before JONI Y. CHANG, BRIAN J. McNAMARA, and
PAUL J. KORNICZKY, *Administrative Patent Judges*.

McNAMARA, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and
37 C.F.R. § 42.73

[1] IPR2018-00102 and IPR2018-01149 have been joined. All references in this Decision are to Papers and Exhibits in IPR2018-00102, unless otherwise noted.

BACKGROUND

On December 12, 2018, IPR2018-01149 was joined to this IPR2018-00102 in which a trial already had been instituted. *See ARM Ltd. and ARM Inc. v. Advanced Micro Devices, Inc. and ATI Techs., Inc.*, Case IPR2018-01140, slip op. (PTAB Dec. 12, 2018) (Paper 16, Decision to Institute and Grant of Motion For Joinder). "Petitioner" refers to MediaTek Inc., MediaTek USA Inc., ARM Ltd., and ARM, Inc., collectively. "Patent Owner" refers to Advanced Micro Devices, Inc. and ATI Technologies ULC collectively.

Inter Partes Review: On October 20, 2018, Petitioner filed a Petition ("Pet.") for *inter* partes review of claims 1–9 of U. S. Patent No. 7,633,506 B1 ("the '506 patent"). On April 27, 2018, in IPR2018-00102, we instituted an *inter partes* review of claims 1–9 of the '506 patent on all grounds asserted in the Petition. Paper 14 ("Dec. to Inst."). Patent Owner filed a Patent Owner Response. Paper 17 ("PO Resp."). Petitioner filed a Petitioner Reply. Paper 28 ("Pet. Reply"). Patent Owner filed a Sur-Reply. Paper 35 ("PO Sur-reply").

Motion to Amend: Patent Owner filed a Contingent Motion to Amend. Paper 18 ("Mot. To Amend"). Petitioner filed an Opposition to Patent Owner's Motion to Amend. Paper 29 ("Opp. To Mot. To Amend"). Patent Owner filed a Reply to Petitioner's Opposition to its Motion to Amend. Paper 34 ("Reply to Opp. To Mot. To Amend"). Petitioner filed a Sur-reply. Paper 41 ("Pet. Sur-Reply to Mot. To Amend").

Motion to Exclude Evidence: Patent Owner filed a Motion to Exclude Evidence. Paper 40 ("Mot. To Exclude"). Petitioner filed a Response. Paper 42 ("Resp. To Mot. To Exclude").

We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a). We base our decision on the preponderance of the evidence. 35 U.S.C. § 316(e); 37 C.F.R. § 42.1(d). A transcript of an oral hearing held on January 22, 2019, (Paper 47, "H'rg. Tr.") has been entered into the record.

Having reviewed the arguments of the parties and the supporting evidence, we conclude that Petitioner has demonstrated by a preponderance of the evidence that the challenged claims are unpatentable. We also deny Patent Owner's Motion to Amend.

THE '506 PATENT (EXHIBIT 1001)

According to the '506 patent, a three dimensional (3D) image typically is displayed on a two dimensional array of pixels using a plurality of graphical objects, such as points, lines, and polygons, known as primitives that are the basis of most rendering instructions. Ex. 1001, 1:45–50. Visible primitives that are part of a scene are drawn individually by determining those pixels falling within the edges of the primitives and obtaining attributes that correspond to each of the pixels to determine the displayed color value of the applicable pixels. *Id.* at 1:52–60. The final displayed color of an individual pixel may be a blend of colors from multiple surfaces or layers. *Id.* at 1:65–67. A blending function based on an opacity value associated with each pixel of each primitive can be used to blend colors of overlapping surfaces or layers when the top surface is not completely opaque. *Id*. at 1:61–65. 3D image data represents attributes such as color, opacity, texture, depth, and perspective information. *Id*. at 2:5–6. Graphics processing is the execution of draw commands that may include X and Y coordinates for the vertices of the primitive, as well as attribute parameters

for the primitive (color and depth or "Z" data) to generate a display image. *Id.* at 2:6–11.

According to the '506 patent, a graphics chip designed to carry out instruction processing to render graphics on a screen typically has a front-end and a back-end. Ex. 1001, 2:19–20. The front-end receives graphics instructions and generates geometry defining primitives or combinations of primitives that are processed by the back-end, where they might be textured, shaded, colored, or otherwise prepared for final output. *Id.* at 2:20–26. When back-end processing of primitives is complete, each pixel in the screen has a specific number value that defines a unique color attribute the pixel will have when drawn. *Id.* at 2:26–29. That final value is kept in a frame buffer for use at an appropriate time. *Id.* at 2:29–31. Systems have become more complex to accommodate three-dimensional (3D) data, requiring a 256-bit system that processes 512 bits in a single logic cycle. *Id.* at 2:32–40. The use of data words with a 256-bit frame buffer is a challenge for the input/output (I/O) system used by the graphics processing back-end because granularity may be too coarse. *Id.* at 2:40–45.

In the '506 patent, geometry representative data presented to the back-end of the graphics chip is divided into data words and provided to one or more parallel pipelines. *Id.* at 2:55–60. The display screen is divided into tiles and a portion of the display screen defined by one or more tiles is serviced by a pipeline. *Id.* at 2:60–64, Fig. 3. Work is allocated to pipelines based on a repeating square pixel tile pattern. *Id.* 5:23–25. The tiling pattern is based on the number of active pipelines. *Id.* at 5:50–51. As shown in Figure 5, logic 520 in set-up unit 515 intersects graphics primitives with the repeating tile pattern, such that a primitive is sent to a pipeline only

if it is likely to result in generation of covered pixels. *Id.* at 5:25–28. Set-up unit 515 creates a bounding box based on X, Y coordinates for each vertex of a polygon and the bounding box is compared to the tile pattern and mapped to one or more pipelines. *Id.* at 5:51–67.

Each parallel pipeline comprises a raster back-end having (i) a scan converter to step through the geometric patterns passed to the back end, (ii) a "hierarchical-Z" component to define the borders of the geometry more precisely, (iii) a "Z-buffer" for performing three dimensional operations of the data, (iv) a rasterizer for computing texture addresses and color components for a pixel, (v) a unified shader for combining multiple characteristics for a pixel and outputting a single value, and (vi) a color buffer logic unit for taking the incoming shader color and blending it into the frame buffer using the current frame buffer blend operations. *Id*. at 2:65–3:8, Fig. 5.

ILLUSTRATIVE CLAIM

Claim 1, with claim element designations used in the Petition shown in brackets, is reproduced below:

1. [preamble] A graphics chip comprising:
   [a] a front-end in the graphics chip configured to receive one or more graphics instructions and to output a geometry;
   [b] a back-end in the graphics chip configured to receive said geometry and to process said geometry into one or more final pixels to be placed in a frame buffer;
   [c] wherein said back-end in the graphics chip comprises multiple parallel pipelines;
   [d] wherein said geometry is determined to locate in a portion of an output screen defined by a tile; and
   [e] wherein each of said parallel pipelines further comprises a unified shader that is programmable to perform both color shading and texture shading.

GROUNDS OF INSTITUTION

In our Decision to Institute, we instituted trial on the following challenges to patentability (which were all of the grounds asserted):

Claims 1–9 as unpatentable under 35 U.S.C. § 103 over Rubinstein[2] in view of Collodi[3], and

Claims 8–9 as unpatentable under 35 U.S.C. § 103 over Rubinstein, in view of Collodi, and in further view of Zatz.[4]

CLAIM CONSTRUCTION

In our Decision to Institute, we applied the ordinary and customary meaning to the terms not construed. We applied the broadest reasonable interpretation to the following terms that required construction:

Z-buffer logic unit

In our Decision to Institute, we construed "Z-buffer logic unit" to mean "a logic unit that facilitates visibility testing by comparing depth values." Dec. to Inst. 8. The parties do not dispute the substance of our construction of this term.

Hiererchical Z-interface

In our Decision to Institute, we construed "Hierarchical Z-interface" to mean "an interface with a z-buffer logic unit that provides for visibility testing at a coarse level, including, for example, for an entire tile or primitive." Dec. to Inst. 8–9. The parties do not dispute the substance of our construction of this term.

---

[2] U.S. Patent No. 7,102,646 B1, issued September 5, 2006 (Ex. 1004).
[3] U.S. Patent Appl. Publ. No. 2003/0076320 A1, published April 24, 2003 (Ex. 1007).
[4] U.S. Patent No. 6,809,732 B2, issued October 26, 2004.

Early Z-interface and late Z-interface

In our Decisions to Institute, we construed "early Z-interface" to mean "an interface with a z-buffer logic unit that provides for visibility prior to shading and texturing." We construed the term "late Z-interface" to mean "an interface with a z-buffer logic unit that provides for visibility testing after shading and texturing. Dec. to Inst. 9–10. The parties do not dispute the substance of our construction of these terms.

Graphics Chip

Neither party proposed a construction of "graphics chip" prior to our Decision to Institute. The Petition states that the preamble of claim 1 is not limiting because the term "graphics chip" does not describe an essential structure or step. Pet. 34. Petitioner argues that "to the extent the Board finds the preamble is limiting," the "graphics chip" requires only "a system with one or more chips for graphics processing." *Id.* at 35. Patent Owner argues that whether or not the preamble is limiting, "the body of the claim requires that the claimed elements be on the same graphics chip." PO Resp. 24. Therefore, Patent Owner proposes that the broadest reasonable interpretation of "graphics chip" in the context of the front-end is the same graphics chip in the context of the back-end, necessitating that the invention in the challenged claims of the '506 patent exists on a single graphics chip. *Id.* at 25. Petitioner responds that the '506 patent refers to a "graphics system" or "graphics chips" in the plural, that the invention is never described in the Specification as consisting of a single graphics chip, and that Patent Owner cites only to specific embodiments of the purported invention. Pet. Reply 10 (citing Ex. 1001, Abstract, 1:16, 2:49–55, 2:65–3:6, 3:54, 3:62–63, 4:27–32).

The preamble of claim 1 provides antecedent basis for the claim limitations "a front-end in the graphics chip" and a "back-end in the graphics chip." Noting that when complex graphics are desired in computers additional, components or chips are added to assist with complex instructions to render graphics on a screen, the Specification states "[g]raphics chips may be considered as having a front-end and a back-end," and that "[t]he front-end typically receives graphics instructions and generates the primitives or combinations of primitives that define geometric patterns," and "primitives are processed in the back-end where they might be textured, shaded, colored, or otherwise prepared for final output." Ex. 1001, 1:30–31, 2:19–26. The Specification further states that "[t]he present invention relates to a parallel array graphics system" that "includes a back-end configured to receive primitives and combinations of primitives (i.e., geometry) and process the geometry to produce values to pace in a frame buffer" (*id.* at 2:49–54) and relative to Figure 5 describes "An Embodiment of a Back-End Graphics Chip" (*id.* at 4:67–7:14).

Although the antecedents in claim 1 appear to indicate the claim is drawn to a single graphics chip, the description in the '506 patent of a back-end graphics chip as a separate entity indicates that the distinction between a single graphics chip and the implementation of a graphics system using multiple chips is not material for purposes of deciding obviousness in this Decision.

Unified Shader

(a) The ITC construction

After the Petition was filed, in a parallel proceeding at the International Trade Commissions (ITC), the Administrative Law Judge

("ALJ") construed a "unified shader" to be one that performs both color shading and "texture coordinate shading." PO Resp. 13–14. Patent Owner notes that the Commission's Notice of Review proposed the following, even more limited, construction for "unified shader:"

> a single shader circuit capable of performing color shading **and texture coordinate shading**, wherein the single shader circuit may not include separate dedicated hardware blocks that perform separate color and texture operations, and **wherein texture coordinate shading** may include texture address operations, indirect texturing, and bump mapping performed by the unified shader **to modify texture coordinates**.

Id. at 14 (citing Ex. 2008, 3) (emphasis in PO Resp.). According to Patent Owner, our Decision to Institute did not consider the ALJ's construction adequately and Patent Owner urges that we now also consider the Commission's construction. *Id.*

Our Decision to Institute included an extensive discussion of the ITC ALJ's construction. Dec. to Inst. 11–15. For example, we noted that in addition to "unified shader," the ALJ construed the function actually recited in claim 1, i.e., "texture shading," to have its plain and ordinary meaning, which the ALJ stated is "texture shading operations including coordinate *texture mapping* and *texture address operations*." *Id.* at 12 (citing Ex. 1009, 5–6 (17:12–18:3)). Noting that the ALJ's construction of "texture shading" includes two terms that are not otherwise defined, i.e. "coordinate texture mapping" and "texture address operations," we were not persuaded that we could apply the ALJ's construction in this proceeding. We also noted that the ITC staff and Patent Owner proposed "unified shader" be construed differently in the context of other patents in the ITC proceeding (U.S. Patent No. 8,760,454 ("the '454 patent") and U.S. Patent No. 9,582,846 ("the '846

patent"), i.e., to mean "a single shader circuit configured to perform both vertex and pixel operations." *Id*. at 14 (citing Ex. 2003, 10). In view of the undefined terms and the ITC staff's different constructions of unified shader in the context of different patents, we could not determine an ordinary and customary meaning of the terms "texture shading" or "unified shader." *Id.* 14–15.

The construction in the Commission's Notice of Review cited by Patent Owner also does not provide guidance necessary for this proceeding. The following terms in the construction articulated in the Commission's Notice of Review are not used or defined anywhere in the text of the '506 patent: texture coordinate shading, texture address operations, indirect texturing, bump mapping, and modify texture coordinates. *See* Ex. 2008. These terms appear in the U.S. Patent No. 7,796,133 B1 ("the '133 patent"), the application that purportedly was incorporated by reference into the '506 patent. As much of the construction applied by the ITC and proposed by Patent Owner comes from disclosure in the related '133 patent, we analyze the incorporation by reference below.

Furthermore, although we have considered the findings of fact and conclusions reached by the ITC, we are not bound by them. *See Nobel Biocare Servs. AG v. Instradent USA, Inc.*, 903 F.3d 1365, 1375 (Fed. Cir. 2018) (noting the Federal Circuit is not bound by its prior affirmance of an ITC decision when reviewing a final written decision of the Board because "[a]s the Board correctly observed, the evidentiary standard in its proceedings, preponderance of the evidence, is different from the higher standard applicable in ITC proceedings, clear and convincing evidence"). Here, in the instant Final Written Decision, we have made an independent

determination of claim construction and patentability of the challenged claims based on the parties' contentions, the specific evidence presented in this proceeding, and the standards applicable to *inter partes* review proceedings.

Incorporation by reference of the '133 patent

Patent Owner argues that the '506 patent includes a definition of a unified shader, i.e., "the claimed unified shader must be capable of performing both color shading and texture coordinate/address shading," as "already identified by the ALJ, OUII, and the ITC Commission in the corresponding ITC investigation." PO Resp. 1. Patent Owner's reliance on the ITC construction reflects an argument that incrementally incorporates into the construction of "unified shader" features that are not found in the literal text of the '506 patent. Claim 1 recites that the unified shader performs "texture shading." Patent Owner cites the '506 patent Specification to assert that the '506 patent specifically defines the unified shader to require a single shader circuit that can perform both color shading and texture *address* shading. PO Resp. 15 (citing Ex. 2010 (Pfister Dep. 8:10–20)); *see also*, Ex. 1001, 6:45–54 ("A unified shader is so named because the functions of a traditional color shader and a traditional texture address shader are combined into a single unified shader. The unified shader performs both color shading and texture address shading."). Although the '506 patent does not use the term "texture coordinate shading," Patent Owner next relies on the disclosure of texture addresses and texture coordinates in the '506 patent and a purported incorporation by reference of the application that led to issuance of the '133 patent to argue that the '506 patent equates texture address shading with texture coordinate shading. Ex.

1001, 6:60–63, 11:15–16. Patent Owner then relies on the ’133 patent and the ITC’s construction to further limit the construction of “unified shader” in the ’506 patent to a shader that performs texture coordinate shading that may include texture address operations, indirect texturing, and bump mapping performed by the unified shader to modify texture coordinates. PO Resp. 14. None of these functions are described in the ’506 patent. Even the Commission’s construction of a “texture coordinate shader” states that texture coordinate shading *may* include these functions, but it does not require them. *See* Ex. 2008, 3.

The issue of whether the ’506 patent properly incorporated the ’133 patent by reference came to the Board’s attention in the context of Patent Owner’s Motion to Amend. H’rg. Tr. 14:18–15:6. We address Patent Owner’s Motion to Amend later in this Decision. For purposes of this discussion, we note that although Patent Owner’s Motion to Amend proposes to amend claim 1 to recite “texture coordinate shading” explicitly (*see* Opp. To Mot. To Amend 29, 1), Patent Owner urges us to construe “texture shading” in the context of the unified shader in unamended claim 1 to mean “texture coordinate shading.” PO Resp. 15–19.

U.S. Patent Application 10/724,384 (“the ’384 application”) that matured into the ’506 patent was filed in November 26, 2003. Ex. 1001, cover page. As to the “unified shader,” the ’384 application stated:

> A unified shader is so named because the functions of a traditional color shader and a traditional texture address shader are combined into a single, unified shader. The unified shader performs both color shading and texture address shading. The conventional distinction between shading operations (i.e., color texture map and coordinate texture map or color shading operation and texture address operation) is not handled by the use

> of separate shaders. In this way, any operation, be it for color shading or texture shading, may loop back into the shader and be combined with any other operation.
>
> The functionality of a unified shader is further described in commonly owned co-pending U.S. Patent Application entitled "Unified Shader", with serial number 10/xxx,xxx, filed December XX, 2003, and is hereby fully incorporated by reference.

Ex. 1017, 307. Thus, as originally filed, the application that matured into the '506 patent did not describe the functionality of the unified shader (other than to state it performs color shading and texture address shading), but instead sought to incorporate that description by reference to an application that had not been filed and could not be identified.[5] Patent Owner took no action to address this issue until filing an Office Action Response on July 10, 2007, when Patent Owner amended the Specification's description of the unified shader to identify U.S. Patent Application 10/730,965 filed December 8, 2003 ("the '965 application") as the referenced application. Ex. 1017, 252.

The '965 application, which matured into the '133 patent, is identified as a continuation of abandoned U.S. Patent Application No. 10/716,946 filed

[5] Similarly, in its description of set-up unit 515, the scan converter and hierarchical Z interface, and the scan converter and Early Z interface, the originally filed '384 application that matured into the '506 patent stated that "the functionality" of each device "is further described in commonly owned co-pending U.S. Patent Application entitled 'Scalable Rasterizer Interpolator', with serial number 10/xxx,xxx, filed December XX, 2003, and is hereby fully incorporated by reference." Ex. 1017, 305, 306. In an amendment filed on July 7, 2007, Patent Owner amended the Specification to change this incorporation by reference to identify Application No. 10/730,864 ("the '864 application"), filed December 8, 2003. *Id*. at 251–252.

on November 18, 2003 ("the '946 application"), i.e., several days before the November 26, 2003 filing date of the '384 application that matured into the '506 patent. *See* Ex. 2003. Although the '946 application has not been made of record in this proceeding, as a continuation application, the '965 application that matured into the '133 patent would have been the same as the abandoned '946 application. A review of Patent Office records indicates that before the Office took any action concerning it, the '946 application was abandoned on February 4, 2004, several weeks after the '965 application was filed. We are not aware of any explanation in the record concerning why the '946 application was abandoned and the '965 application was filed in its place.

Figures 9–15 and the corresponding subject matter that appears in the '506 patent beginning at column 9, line 27 through column 14, line 20 were not included in the '384 application as filed on November 26, 2003. *See* Ex. 1017, 312. Thus, the '384 application as originally filed did not include the '506 patent text under the headings "Unified Shader," "Unified (Pixel) Shader Architecture," "Shader Code Partitioning," "Control Logic," "Register Subsystem," "Multiple Shaders," and "ALU I/O Description." *See* Ex. 1001, 9:27–14:20.

On March 31, 2008, Patent Owner filed "a Substitute Specification that includes subject matter of a co-pending application (serial no. 10/730,965) that was incorporated by reference in its entirety in the originally filed application." Ex. 1017, 126. As discussed above, the subject matter was not incorporated by reference in the originally-filed application because Patent Owner did not identify a specific application to be incorporated by reference. *See* 37 C.F.R. § 1.57(c). Indeed, the '384

application that matured into the ’506 patent included three separate incorporation by reference assertions in the form of application “10/xxx,xxx, filed on Dec. XX, 2003” and Patent Owner’s July 10, 2007 amendment incorporated two different applications by reference, i.e., the ’965 application concerning the unified shader and the ’864 application concerning the set-up unit, scan converter, and the Early and Hierarchical Z interfaces. The existence of the subject matter to be incorporated by reference is irrelevant until Patent Owner actually identifies the subject matter to be incorporated. *See id.*, Pet. Reply to Opp. to Mot. to Amend 4. Identification of the application to be incorporated by reference occurred years after the original filing date, i.e., on July 10, 2007 at the earliest.

The Substitute Specification filed on March 31, 2008, added Figures 9–15 and the corresponding description, but did not import into the ’384 application (now the ’506 patent) the literal text using the term “texture coordinate shading” or include other material from the ’965 application (now the ’133 patent), e.g., discussions of “indirect texturing,” and “bump mapping.” There is no evidence that this issue was brought to the attention of the ITC.

(b) The description of “unified shader” in the ’506 patent

In our Decision to Institute, we declined to construe the term “unified shader” by itself. Dec. to Inst. 10–25. Rather than construe this term out of the context of the claims, we preliminarily construed the entire phrase “unified shader that is programmable to perform both color shading and texture shading” to mean “a processing mechanism that through an interface receives packets from a rasterizer and has at least one ALU/memory pair that can be programmed to adjust the color of a pixel, and issue a texture

request to a texture unit or write received texture values to a memory, and outputs resultant values to a frame buffer." *Id.* at 25 (emphasis omitted). Our construction recognized that the recited limitation includes several elements: (i) unified shader; (ii) color shading, and (iii) texture shading.

We turn first to the "unified shader" portion of the limitation. Our construction of the "unified shader" portion of this term is based, in part, on the following description of a "unified shader" in the '506 patent Specification as amended on March 31, 2008:

> One embodiment of a unified shader is shown in the block diagram of FIG. 9. Unified shader **1100** performs per-pixel shading calculations on rasterized values that are passed from a rasterizer unit **1110.** The results of the calculations are sent to frame buffer **1120.** As part of the calculation performed by unified shader **1100,** a texture unit **1130** may receive texture lookup requests from the shader **1100.** The actual shading algorithm used may vary and may be defined by a set of instructions, such as microcode instructions.

Ex. 1001, 9:36–44.

Patent Owner proposes that we construe "unified shader" to "include 'a single shader circuit capable of performing color shading and texture coordinate shading.'" PO Resp. 18. Patent Owner's proposed construction is highly functional and introduces at least the following uncertainties: (i) it leaves open to question what makes up the shader circuit, rendering the term "shader circuit" a nonce term; (ii) it changes the claim term "texture shading" to "texture coordinate shading"—thus, on its face, Patent Owner proposes further limiting the claim to a particular type of texture shading, i.e., texture *coordinate* shading; and (iii) it does not define either "texture shading" or "texture coordinate shading," thereby providing no insight into the functional being performed. In view of these uncertainties, Patent

Owner's proposed construction provides no basis upon which we can interpret the metes and bounds of the claim.

Petitioner agrees with Patent Owner that we should not import into the definition of unified shader the structural requirements that the processing mechanism "through an interface receives packets from a rasterizer and has at least one ALU/memory pair," and "outputs resultant values to a frame buffer." Pet. Reply 2, 7–10. The only structural limitation on the unified shader itself in this preliminary construction is that it includes at least one ALU/memory pair. *See* Ex. 1001, 9:45–10:40; Fig. 10. The remaining portions of this preliminary construction concern inputs (i.e., packets from a rasterizer) to and outputs (resultant values to a frame buffer) from the unified shader. Petitioner also proposes that we simplify the "texture shading" element of claim 1 to mean "issue a texture request or receive texture information in response to that request." Pet. Reply 9. Thus, Petitioner proposes we construe "unified shader" as a "processing mechanism that can be programmed to adjust the color of a pixel and issue a texture request or receive texture information in response to that request." *Id.* at 10 (emphasis omitted).

Like Patent Owner's proposed construction, Petitioner's proposed construction is highly functional and proposes no structure for the unified shader, other than it is a "processing mechanism" programmed to perform certain functions, i.e. issue a texture request and receive texture information in response to that request. As compared to the '506 patent's description of operations 5 and 6 performed by the unified shader architecture shown in Figure 10 discussed below, Petitioner's proposed construction substitutes "texture request" for the term "texture address" and "texture information"

for "texture value." *See* Ex. 1001, 10:15–19. Petitioner's proposal is consistent with the '506 patent's description of operations issuing a texture address as "texture requests" and receiving in the SRAMs returned "texture data." *Id.* at 10:15–20.

Petitioner notes that defining a particular claim term by its function is not improper. Pet. Reply. 8 (citing *Hill-Rom Svcs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1374-75 (Fed. Cir. 2014)). In this case, however, the claim itself recites the particular function, i.e., "unified shader that is programmable to perform both color shading and texture shading." Further construing "unified shader" by the function it performs is either redundant or contradictory to the recited function. That the unified shader is programmable to perform both color shading and texture shading begs the question as to what constitutes the unified shader, other than a "circuit" as Patent Owner proposes, or a "processing mechanism" as Petitioner proposes.

The term "unified shader" is not used consistently in the Specification. The Specification describes Figure 9 as "a block diagram of a unified shader according to an embodiment of the present invention." Ex. 1001, 3:37–38, 9:36–37. The implication from this description of Figure 9 is that the elements of Figure 9 taken together make up at least one embodiment of the claimed unified shader. Figure 9 includes four elements: (i) rasterizer block 1110, shown providing an input to; (ii) unified shader block 1100 that provides inputs to and receives outputs from; (iii) texture unit block 1130; and (iv) frame buffer block 1120 that receives inputs from unified shader block 1100. The Specification states that the interface receives packets from a rasterizer and that the outputs of the unified shader are provided to a frame buffer. Ex. 1001, 9:37–40 ("Unified shader 1100

performs per-pixel shading calculations on rasterized values that are passed from a rasterizer unit 1110. The results of the calculations are sent to the frame buffer 1120."); *id*. at 11:14–17 ("Rasterizer 1400 generates packets of data containing information for a block of 16 pixels (4 quads). Each pixel contains one or more sets of texture coordinates (texture addresses) and one or more color values.").

The presence of block 1100 labelled "unified shader" in Figure 9, which itself is described as being a block diagram of a "unified shader," is only one inconsistency in the use of the term "unified shader." In the context of describing the "Unified (Pixel) Shader Architecture," the Specification also states that "Fig. 10 is a block diagram of a Unified Shader according to an embodiment of the present invention." Ex. 1001, 9:45–47. Figure 10 is also described as "a unified shader architecture according to an embodiment of the present invention." *Id.* at 3:37–38. Figure 10 is reproduced below:

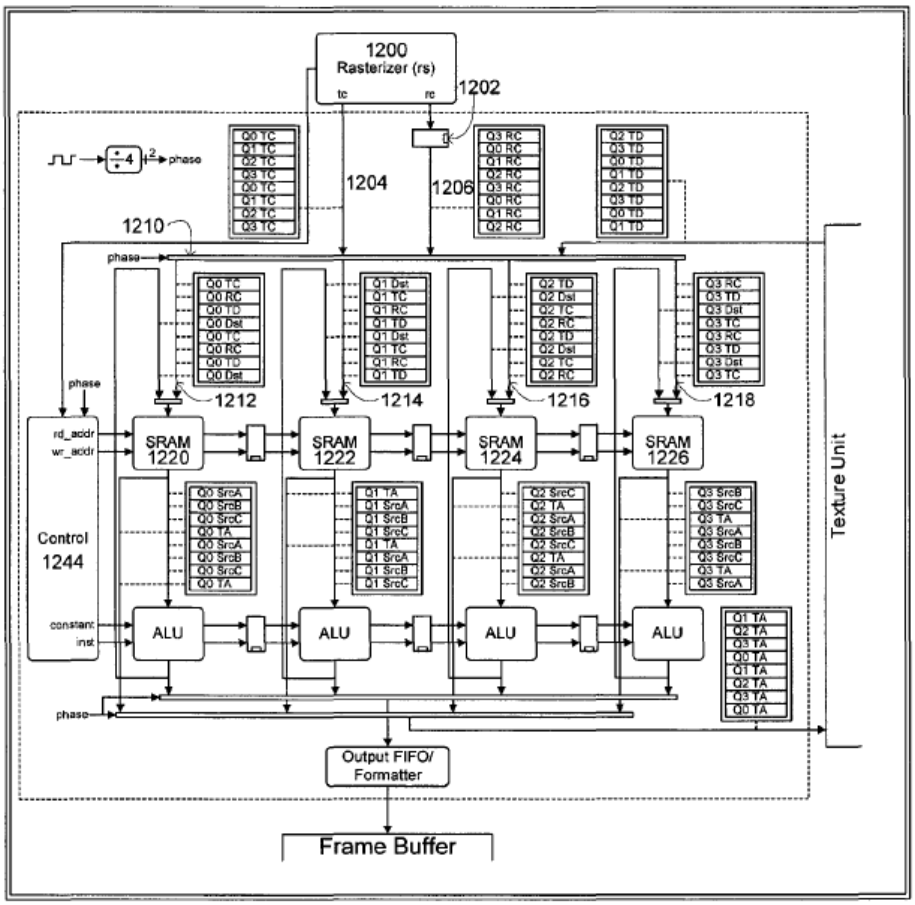

FIGURE 10

The structure of the Unified Shader in Figure 10 includes rasterizer 1200, control 1244, four ALU/SRAM pairs (1220, 1222, 1224, 1226), an un-numbered block labelled output FIFO/formatter, and an unnumbered block labelled Frame Buffer. The Output FIFO/Formatter, busses 1204, 1206, clock delay circuit 1202 connected to bus 1206 (with the clock delay further indicated by the order of Q(0-3)TC and Q(0-3)RC), a divide by four circuit, control block 1244, and the four ALU/SRAM pairs and corresponding busses 1212 (Q0 data), 1214 (Q1data), 1216 (Q2 data), 1218 (Q3 data) are shown inside an un-labeled box of dashed lines. Four way crossbar 1210 labelled “phase” and a similar line labelled “phase” indicate

bi-directional communication with an unnumbered "Texture Unit." Although the Specification states that Figures 9 and 10 both show a unified shader (according an embodiment of the invention), Figure 10 does not include a block labeled "unified shader," as shown in Figure 9.

The Specification describes rasterizer 1200 of the Unified Shader illustrated in Figure 10 generating a texture address (tc) and rasterization color (rc) (in any suitable format) at a rate of one "pixel quad" (a 2x2 tile of pixels) per clock. *Id.* at 9:48–51. Rasterization color rc, delayed by one clock to provide correct interleaving, and un-delayed texture address tc are provided to respective busses 1206, 1204 that pass the packet through four way crossbar 1210 that rotates through four clocks to provide outputs 0–3 to corresponding quads 0–3. *Id.* at 9:51–59. As a result, output 0 of crossbar 1210 contains only quad 0 data, output 1 (1214) contains only quad 1 data, output 2 (1216) contains only quad 2 data, and output 3 (1218) contains only quad 3 data. Each of four identical ALU/SRAM pairs[6] processes the rc and tc data from the rasterizer, performing the following operations in one four clock cycle:

1. Writes one rasterizer texture address to the SRAM.
2. Writes one rasterizer color value to the SRAM.
3. Reads up to three source operands from the SRAM and executes one shader instruction.
4. Writes the result from the (2nd previous) shader instruction back to the SRAM.
5. Reads one texture address from the SRAM and issues it to the texture unit.
6. Writes one return texture value to the SRAM.

[6] Memory units other than SRAMs, with space allocated to store input values and intermediate variables needed by the shader program for multiple quads can also be used. Ex. 1001, 9:67–10:3.

Id. at 9:60–10:14. A single mux multiplexes texture requests for the four ALU/SRAM pairs into a single stream containing one texture request every clock, so that the resulting texture data is de-multiplexed and written back into the SRAMs. *Id.* at 10:15–20. Control block 1244 generates SRAM read/write addresses and generates ALU instructions for the first SRAM and ALU 1220—each SRAM/ALU pair receives the same instructions delayed by one clock cycle from the previous one. *Id*. at 10:23–26.

Claim 1 recites a graphic chip with a back-end that comprises multiple parallel pipelines. Claim 1 further recites that the parallel pipelines comprise a unified shader that is programmable to perform both color shading and texture shading. The Specification discusses the parallel pipelines in the backend in the context of Figures 1 and 3, in each case showing the geometry from the front end being routed to the pipelines and the pipelines routing the information to the frame buffer. Ex. 1001, Figs. 1, 3. Figures 1 and 3 do not mention a unified shader. Figure 5 shows a back-end chip embodiment having rasterization pipeline 520 including scan converter 540 with early and hierarchical Z interfaces providing data to rasterizer 560, where it is routed to unified shader 570. *See id.* at 4:67–7:14.

Consistent with Figures 9 and 10, Figure 5 shows unified shader 570 sending and receiving information to texture unit 585. As discussed above, Figure 9 (Ex. 1001, 9:36–37) purports to be a block diagram of a unified shader embodiment in which the unified shader includes the rasterizer, texture unit, frame buffer and a unit called "unified shader." Figure 10, which also purports to be a unified shader embodiment (*id*. at 9:46–47), illustrates the parallel pipelines (ALU/SRAMs) receiving data from the

rasterizer and indicates that these elements communicate with the texture unit.

Claim 7 recites the parallel pipeline as further comprising a scan converter, a rasterizer, and a texture unit. Applying the doctrine of claim differentiation, we cannot construe the "unified shader" in claim 1 to include the scan converter, the rasterizer, or the texture unit, notwithstanding the descriptions of Figures 9 and 10. As the only purported description of the unified shader itself is in Figures 9 and 10 and these include the rasterizer and texture unit (and Figure 10 shows the ALU/SRAM pairs that make up the parallel pipelines as communicating with the texture unit), we understand one description of a structure that corresponds to the unified shader, i.e., block 1100 in Figure 9 to be the structure within the dashed lines in Figure 10.

Referring to Figure 5, the '506 patent Specification states that:

> [U]nified shader 570 works in conjunction with texture unit 585 and applies a programmed sequence of instructions to the rasterized values. These instructions may involve simple mathematical functions (add, multiply, etc.) and may also involve requests to the texture unit. A unified shader reads in rasterized texture addresses and colors, and applies a programmed sequence of instructions. A unified shader is so named because the functions of a traditional color shader and a traditional texture address shader are combined into a single, unified shader. The unified shader performs both color shading and texture address shading. The conventional distinction between shading operations (i.e., color texture map and coordinate texture map or color shading operation and texture address operation) is not handled by the use of separate shaders, In this way, any operation, be it for color shading or texture shading, may loop back into the shader and be combined with any other operation.

Ex. 1001 6:43–59.

Although the above description of Figure 5 articulates that the unified shader is programmed to perform both color shading and texture address shading, it does not describe the program that performs those functions. The discussion of the architecture embodied in Figure 10 describes the operations that are performed by control logic block 1244 and the ALU/SRAM pairs in the unified shader. As to "texture address shading," the description of Figure 10 states that operation 5 reads one texture address from the SRAM and issues it to the texture address unit; operation 6 writes one return texture value to the SRAM. *Id.* at 10:12–14.

In order to "appl[y] a programmed sequence of instructions to the rasterized values" and to "read[] in rasterized addresses and colors" (Ex. 1001, 6:44–47), the unified shader must receive packets from a rasterizer. The Specification explains that rasterizer 560 receives its data through an early and hierarchical interface (scan converter 540). *Id.* at 6:1–37. During the hearing, Petitioner conceded that a proper construction of "unified shader" requires some structure in the form of computational units, e.g., ALUs, and access to memory, and that the unified shader receives rasterized values. H'rg. Tr. 10:3–11:5. Consistent with the recitation of a parallel pipeline comprising a unified shader and the description of Figure 10 in the Specification, there must be at least one pair of ALU/SRAMs (or other memory) to carry out the instructions (one ALU/SRAM and another ALU/SRAM working in parallel). Thus, we construe the structure of the claimed unified shader to include at least one computational unit that receives rasterized values and has access to memory.

(c) The meaning of "texture shading"

We now turn our attention to the meaning of the term "texture shading" in claim 1. Petitioner acknowledges that the '506 patent states "[t]he unified shader performs both color shading and texture address shading." Pet. Reply 2–3 (citing Ex. 1001, 6:49–54). Petitioner argues that "the only texture operations described in the '506 patent that are performed by the unified shader involve issuing texture requests and receiving texture information back." Pet. Reply 4 (citing Ex. 1001, 6:47, 9:40–43, 10:12–20; Ex. 1012 (Pfister Suppl. Decl.) ¶ 10; Ex. 1018 (Wolfe Tr.) 26:4–9).

Patent Owner contends that the '506 patent specifically defines "unified shader" as requiring texture address shading and equates texture address shading with texture coordinate shading. PO Resp. 18 (citing Ex. 1001, 11:14–19). Patent Owner does not claim to have invented either texture address shading or texture coordinate shading. H'rg. Tr. 32:8–10, 14–15. According to Patent Owner, "[w]hat wasn't conventional was having a unified shader, a single shader circuit that could do that type of specific texture coordinate shading and one that could do color shading as well. That was the invention." *Id*. at 32:10–13. Without more, however, Patent Owner's description of the invention merely articulates an idea, i.e., providing a single device that performs both functions—but Patent Owner does not describe the structure or operation of such a circuit.

Patent Owner also argues that "[s]hading, everybody agrees, is a modification of something . . . in color shading, it's the modification of a color of a pixel. Texture coordinate shading is the modification of a texture coordinate. Texture address shading is the modification of a texture address." Hr'g Tr. 31:11–15. As Petitioner notes, however, the '506 patent

does not define texture shading, texture address shading, or texture coordinate shading. *Id*. at 18:7–17.

The text of the ’506 patent cited by Patent Owner refers to “texture coordinates/texture addresses” but does not mention “texture coordinate shading.” Indeed, the term “texture coordinate shading” is not used at all in the ’506 patent. Although Patent Owner criticizes the Decision to Institute as focusing on the disclosure of “texture coordinate shading” in related U.S. Patent No. 7,796,133 B1 (Ex. 2003, “the ’133 patent”) (*see* PO Resp. 17–18), in its Preliminary Response, Patent Owner urged exactly that approach (Prelim. Resp. 12–14). In the Patent Owner Response, Patent Owner again cites the ’133 patent “which the ’506 patent explicitly incorporates by reference” to argue that the ’506 patent equates texture coordinate shading and texture address shading. PO Resp. 18 (citing Ex. 2003, 3:15–29). As previously discussed, Patent Owner also urges that we adopt the ITC’s construction that includes terms from the ’133 patent, but there is no indication that the late incorporation by reference or its implications for written description support relative to the ’506 patent was ever disclosed to the ITC.

More importantly, in the Decision to Institute, we compared the disclosures in the ’506 patent (referring to a “traditional texture address shader”) and the ’133 patent (referring to a “texture coordinate shader”). Dec. to Inst. 15–17. A further comparison of the ’506 patent’s description of Figure 10 and the ’133 patent’s description of Figure 2 indicates that the unified shader illustrated in the respective figures is programmed to perform the same six operations. Ex. 1001, 9:60–10:14; Ex. 2003, 5:24–6:9. In view of Patent Owner’s acknowledgement that texture address shading and

texture coordinate shading are equivalent, to determine the meaning of "programmed to perform . . . texture shading," we focus our analysis on the disclosure in the '506 patent (and the '133 patent) concerning the steps carried out by the unified shader.

As discussed above, Figure 10 of the '506 patent shows the parallel pipelines as part of the unified shader and the Specification describes the processing that takes place within the unified shader as a series of six operations performed in one four clock cycle. *Id.* at 9:60–10:14, Fig. 10. The code is partitioned so that texture operations at the same level of indirection are executed in parallel, where indirection level refers to the number of passes through the texture system. *Id.* at 10:42–67, Fig. 11. The '506 patent provides additional disclosure of the control logic for the Unified Pixel Shader in the context of a state machine embodiment shown in Figure 12 in which a "control token" is generated as data for each block of pixels is received from the rasterizer. *Id.* at 11:1–13. Rasterizer 1400 generates packets of data containing information for a block of 16 pixels (4 quads), where each pixel "contains one or more sets of texture coordinates (texture addresses), and one or more colors." *Id.* at 11:14–17.

> For each instruction in this sequence, the level 0 Texture machine instructs the SRAM's to read a set of texture coordinates, and then issues commands to the texture unit to perform a lookup on these texture coordinates. As data is returned from the texture unit, it gets written into the SRAM's at the appropriate location. Upon receipt of the return data for the last instruction in the level 0 texture sequence, the level 0 texture machine passes the control token to the level 0 ALU machine **1450.**

Ex. 1001, 11:34–42. Patent Owner urges that the '506 patent equates "texture coordinates" with "texture addresses." PO Resp. 17–18 (citing Ex. 1001, 11:14–19). At this state, machine control implements

substantially the same operations discussed in the context of the unified shader shown in Figure 10, we consider the use of the term "texture coordinates" in the context of instructing the texture unit to perform a look-up to be synonymous with the term "texture addresses," i.e., the state machine of the unified shader could perform the texture look-up using either texture coordinates or texture addresses. *See also,* Ex. 1001, 11:16 (referring to "texture coordinates (texture addresses)"). For this reason, we agree with Petitioner that a proper construction encompasses identifying pixels by both texture coordinates and texture addresses and refers more generally to issuing a "texture request" and receiving "texture information" or "texture data." The '506 patent consistently describes the operations associated with Figures 10 and 12 as commands to the texture unit to perform a lookup. *See* Ex. 1001, 10:12–14 (operations 5 and 6, read one texture address from SRAM and issue it to the texture unit, then write one value to the SRAM); 12:34–39 ("For each instruction in this sequence, the level 0 Texture machine instructs the SRAM's to read a set of texture coordinates, and then issues commands to the texture unit to *perform a lookup* on these texture coordinates. As data is returned from the texture unit, it gets written into the SRAM's at the appropriate location." (emphasis added)). *See also*, Ex. 2003, 7:1–6 (the exact same language in the '133 patent). The texture unit performs the lookup. As the texture unit is a separate element (*see* claim 7), the function performed by the unified shader is to issue texture commands (or requests) and receive responses and store them. Applying the broadest reasonable interpretation, whether or not texture address shading is equivalent to texture coordinate shading, the unified shader carries out at least those operations in the disclosed embodiment, i.e., issue a texture

request (to the texture unit), receive a response (from the texture unit), and store the response (in SRAM).

(d) Related issues — "packets" and "programmable to"

In our construction of "unified shader" in the Decision to Institute, we included a reference to "a processing mechanism that through an interface receives packets." Dec. to Inst. 25 (emphasis omitted). Neither party proposes that the construction be limited to receiving packets. The description of unified shader architecture in Figure 10 states that "rasterizer 1200 generates a texture address (tc) and a rasterization color (rc) in any suitable format and order." Ex. 1001, 9:47–48. In view of this disclosure, we do not limit the unified shader to receiving information in the form of packets.

Claim 1 recites "a unified shader is *programmable to* perform both color shading and texture shading." Emphasis added. Although neither party proposes that we construe the term "programmable to," in related case *MediaTek v. Advanced Micro Devices*, Case IPR2018-00101 ("the -00101 IPR"), Patent Owner asserts that "[w]hen a POSA refers to a programmable shader, s/he understands this to refer to newer technologies that began to replace fixed-function pipelines in order to be compatible with DirectX 8, its descendants, and other similarly capable graphics standards." -00101 IPR PO Resp. 56. Noting that Patent Owner's expert, Dr. Wolfe, despite a lengthy discussion of programmable shaders in the graphics industry, does not reference the '506 patent on this issue, Petitioner argues that nothing in the '506 patent supports such a narrow construction. -00101 IPR Pet. Reply 18 (citing Ex. 2009, Declaration of Dr. Andrew Wolfe in Support of Patent Owner's Response ("Wolfe Resp. Decl.") ¶¶ 83–93). We agree with

Petitioner. Neither the ’506 patent Specification nor the ’133 patent Specification ascribes any special meaning to the term “programmable to.” Claim 17 of the ’506 patent recites the same language as claim 1. Claim 10 of the ’506 patent recites “the unified shader performs both color shading and texture shading based on programmable instructions.” The term “programmable to” is not used elsewhere in the ’506 or the ’133 patents. The ’506 patent states:

> A unified shader 570 works in conjunction with texture unit 585 and applies a programmed sequence of instructions to the rasterized values. These instructions may involve simple mathematical functions (add, multiply, etc.) and may also involve requests to the texture unit.

Ex. 1001, 6:43–47.

(e) Summary and construction of “unified shader”

The ’506 patent claims are drawn to a graphics chip (claim 1) and a method for processing computer graphics (unchallenged claim 10) with a “unified shader” that performs both color shading and “texture shading.” As originally filed, the application that led to issuance of the ’506 patent referred to “texture address shading” but did not include a description of the unified shader. The application entitled “unified shader” that led to the ’133 patent and was incorporated by reference into the ’506 patent application in 2007 refers to “texture coordinate shading” and includes a description of the architecture of an embodiment of the unified shader. The text describing that architecture and the corresponding figures from the application that led to the ’133 patent were explicitly added to the ’506 patent application in 2008. The description of the architecture common to the ’506 patent and the ’133 patent describes an embodiment of a unified shader performing six operations. Patent Owner contends that “texture shading” should be

construed to mean "texture address shading" and that "texture address shading" is equivalent to "texture coordinate shading." Whether one refers to the terms "texture shading," "texture address shading," or "texture coordinate shading," our construction of unified shader must cover the embodiment in the common description in the '133 patent and the '506 patent. Therefore, a unified shader must be construed to perform those six operations described as executed by an embodiment of the architecture of the unified shader. Applying the broadest reasonable construction, other features, such as indirect shading, bump mapping, and modifying texture coordinates are not addressed by the disclosed embodiment, and therefore, are not part of the construction. This approach is consistent with the ITC construction cited by Patent Owner that states "wherein texture coordinate shading *may* include texture address operations, indirect texturing, and bump mapping performed by the unified shader to modify texture coordinates." Ex. 2042, 15–20 (emphasis added).[7]

In consideration of the above, for purposes of this Decision, we construe the term "unified shader programmed to perform color shading and texture shading" to mean: *at least one computational unit with access to memory that receives rasterized values and executes instructions to adjust the color of a pixel, issue a texture request, and store received values.*

[7] Notwithstanding that the term "texture coordinate shading" does not appear in the literal text of the '506 patent, the Commission states that texture address shading is "a term used interchangeably with 'texture coordinate shading' in the '506 patent." Ex. 2042, 18–19.

ANALYSIS OF PRIOR ART CHALLENGES

Claims 1–9 as Unpatentable Over Rubinstein In View of Collodi

Rubinstein

Rubinstein discloses a demand-based memory system for graphics applications using a tiled architecture that dynamically allocates varying amounts of memory per tile depending on demand. Ex. 1006, Abstract. The computer system for processing graphics data in Rubinstein has a binning engine that receives graphic data and bins the graphic data into screen tiles. *Id*. at 2:27–29. A memory coupled to the binning engine is divided into a plurality of pages and is configured to allocate one or more pages per tile depending on tile requirements. *Id*. at 2:30–34. A page-based memory system removes the limitation on the memory size of a single tile by allocating more pages when necessary. *Id.* at 2:1–2. When all the geometry data stored in a page of memory has been rendered, the page is freed and made available for reallocation. *Id.* at 2:5–7.

Collodi

Collodi discloses a "Programmable Per-Pixel Shader With Lighting Support" in which one or more Vector Shading Units operates in parallel with a Programmable Shading Unit to accelerate lighting and color calculations and make results available for use in the Programmable Shading Unit. Ex. 1007, Abstract. Figure 1 of Collodi shows Programmable Shader Unit 2 operatively connected to texture memory 3, rasterizer 6, and Lighting Sequence Unit 1. *Id.* ¶ 15. Collodi states that a standard programmable shading unit contains hardware for execution of a user-defined sequence of texture lookup and color blend operations. *Id.* at Abstract, ¶ 15. Shading

values produced by the various hardware units provide the Programmable Shading Unit more flexibility to enhance user customization of shading routines. *Id.* ¶ 16. Collodi's Programmable Shading Unit may be programmed to generate a bump map vector procedurally or to provide the bump using one or more texture lookups. *Id.* ¶ 18. Collodi also states that it is known to use massively parallel pixel architectures consisting of a plurality of pixel processors in which per-pixel processing units determine pixel color from a combination of texture map lookups, lighting calculations, and interpolated vector color values by executing user-programmable sequences of pixel commands that operate on pixel data to produce a final written color value. *Id*. ¶¶ 3, 12.

Claim 1

The preamble of claim 1 recites a graphics chip. Petitioner contends that the preamble is not limiting because it does not describe an essential structure or step. Pet. 34. Petitioner further argues that to the extent the preamble is limiting, the recitation of a graphics chip requires only a system with one or more chips for graphics processing. *Id.* at 33–34. We discuss this issue above in the context of our construction of "graphics chip." In any case, Petitioner further argues that to the extent the preamble is limiting and requires a single graphics chip, Rubinstein in view of Collodi discloses this element because (i) Rubinstein discloses a graphics processor with functional blocks that may be implemented as an integrated solution, and (ii) Collodi discloses a programmable hardware per-pixel shading unit that can be implemented with graphics hardware, such as a graphics chip. *Id.* at 36–37.

Petitioner identifies, as claim element 1-a, the limitation reciting "a front-end in the graphics chip configured to receive one or more graphics instructions and to output a geometry." Pet. 37. As evidence that Rubinstein discloses the claimed front-end, Petitioner cites Rubinstein's disclosure of a geometry processor that receives geometry mode data defining or specifying textures, blending effects, and translucence that may be applied to a rendered geometry and providing graphics data to a binning engine. *Id.* at 37–38. Patent Owner does not respond specifically to Petitioner's contentions concerning claim element 1-a. We are persuaded by the arguments and evidence that Petitioner has demonstrated Rubinstein discloses the features recited in claim element 1-a.

Petitioner identifies as claim element 1-b the limitation reciting "a back-end in the graphics chip configured to receive said geometry and to process said geometry into one or more final pixels to be placed in a frame buffer." Pet. 38. Petitioner cites the binning engine, memory controller, and rendering engine with associated memory disclosed in Rubinstein is corresponding to the claimed back-end. *Id.* at 37. Rubinstein discloses rendering engine 118 is coupled to memory controller 114 and accesses the binned graphics data contained in graphics memory 116 to render an image for display. Ex. 1004, 4:2–5. Petitioner further asserts that a person of ordinary skill would have understood that a final image for display includes multiple final pixels to be placed in a frame buffer. Pet. 39–40 (citing Ex. 1004, Fig. 1A; Ex. 1003, Pfister Decl. ¶ 81). Patent Owner does not respond explicitly to Petitioner's contentions concerning claim element 1-b. We are persuaded by the arguments and evidence that Petitioner has

demonstrated Rubinstein in view of Collodi discloses the features recited in claim element 1-b.

Petitioner identifies as claim element 1-c the limitation reciting "wherein said back-end in the graphics chip comprises multiple parallel pipelines." Pet. 40. Petitioner acknowledges that, although Rubinstein discloses a back-end pipeline for pixel processing, it does not expressly disclose using multiple parallel pipelines. *Id*. Petitioner cites Collodi's disclosure that it is desirable to improve rendering efficiency with multiple per-pixel processors running in parallel as teaching this feature. *Id.* (citing Ex. 1007 ¶¶ 3, 14). Petitioner's expert, Dr. Pfister, cites to various graphics chips released in 2000 and 2001 as evidence that at the time the application for the '506 patent was filed, graphics chip manufacturers were selling graphics chips with parallel pipelines. *Id.* at 42 (citing Ex. 1003, Pfister Decl., ¶ 85, identifying Nvidia GeForce 2 chip offered in 2000 (*see* Ex. 1009) and GeForce 3 Chip offered in 2001 (*see* Ex. 1010)). In view of these teachings, Petitioner contends it would have been obvious to implement Collodi's parallel pipelines in Rubinstein. *Id.* at 42 (citing Ex. 1003, Pfister Decl., ¶¶ 84–85). Patent Owner does not respond specifically to Petitioner's contentions concerning claim element 1-c. We are persuaded that Petitioner has demonstrated Rubinstein in view of Collodi discloses the features recited in claim element 1-c.

Petitioner identifies as claim element 1-d the limitation reciting "wherein said geometry is determined to locate in a portion of an output screen defined by a tile." Pet. 42. As evidence that this feature is disclosed in Rubinstein, Petitioner cites Rubinstein's binning engine that divides a screen into sections or tiles and identifies which geometry resides in each

tile. *Id.* at 43 (citing Ex. 1004, 3:32–34). Rubinstein discloses binning engine 112 determines which tiles are touched by each geometry. Ex. 1004, 3:55–63. Patent Owner does not respond explicitly to Petitioner's contentions concerning claim element 1-d. We are persuaded by the arguments and evidence that Petitioner has demonstrated Rubinstein in view of Collodi discloses the features recited in claim element 1-d.

Petitioner identifies as claim element 1-e the limitation reciting "wherein each of said parallel pipelines further comprises a unified shader that is programmable to perform both color shading and texture shading." Petitioner cites Collodi's disclosure of a "programmable shading unit" as evidence that Collodi teaches this feature. Pet. 44. Petitioner notes that Collodi discloses a programmable shading unit that produces a pixel color consisting of an RGB color value or an RGBA color-alpha value (Ex. 1007 ¶ 15) and hardware for execution of the user-defined sequence of texture lookup and color blend operations (Ex. 1007 ¶ 15). Pet. 45. Petitioner further notes that Collodi discloses the programmable shading unit can be configured to generate a bump map vector procedurally or provide the bump map vector as a result of a texture map lookup or a combined result of multiple texture map lookups. *Id*. at 45–46 (citing Ex. 1007 ¶ 18; Ex. 1003, Pfister Decl., ¶ 92). As we addressed in our claim construction discussion, the "unified shader" claimed in the '506 patent requires only that the texture operations comprise at least one of issuing a texture request to a texture unit and writing received texture values to the memory. Petitioner notes that Collodi includes a texture memory operatively connected to the programmable shading unit (Ex. 1007 ¶¶ 14–15) and employs a texture map

lookup or multiple texture map lookups to arrive at a bump vector (*id*. ¶ 18). Pet. 45–46.

Patent Owner argues that neither Rubinstein nor Collodi discloses a programmable unified shader or a front-end and a back-end on a single graphics chip. PO Resp. 25–26. Patent Owner states that Rubinstein is a fixed function rendering pipeline that concerns tiling aspects of graphics, but lacks programmable shaders and provides no modification of texture address or texture coordinates and no mechanism for providing programming information to a shader or shading process. *Id*. at 26–27. As Petitioner points out and as we discuss further below, Collodi discloses a programmable shader. Pet. Reply 11.

Patent Owner also argues that the ’506 patent claims a back-end with a pipeline that is replicated, each of which is attached to the front-end. PO Resp. 28. Patent Owner argues Collodi teaches away from the claimed structure because Collodi describes duplicating the entire “per-pixel processor,” i.e., the front-end and back-end as a whole, rather than the back-end pipelines. *Id*. Petitioner relies not on Collodi, but on Rubinstein, as disclosing that various function blocks in a graphics system, including the claimed front-end, back-end, tiling, and z-buffering, may be implemented by a combination of hardware and software, and that specific implementations of some or all of the functionality of various blocks may be combined into integrated solutions. Pet. 36; Pet. Reply 20. Rubinstein states: “implementation may combine interface controller and graphics processor, and combine graphics and system memory resources into an <u>integrated chipset</u>.” Ex. 1004, 4:20–23 (cited with emphasis at Pet. 36).

Patent Owner further argues that Petitioner's challenge fails because Collodi's Programmable Shader Unit is not capable of performing "texture shading," i.e., texture coordinate shading or texture address shading. PO Resp. 29. We disagree with Patent Owner's analysis because it is based on an improper construction of a unified shader programmed to perform texture shading. According to Patent Owner, none of Petitioner's references to Collodi establish modification of a texture coordinate/address, but refer instead only to texture lookup and texture mapping. *Id*. at 31–32. As discussed above, however, the only architecture and operations disclosed in the '506 patent as performed by the unified shader involve texture lookup. Ex. 1001, 10:4–14. Thus, Collodi discloses a shader programmed to perform texture shading and color shading.

We would reach the same result even if we were to adopt Patent Owner's proposed claim construction, because Collodi discloses modifying texture coordinates. Collodi discloses a programmable shading unit that "may be configured to generate a bump map vector . . . where the texture map contains 2 dimensional values or the texture map contains light values which are subsequently converted to two dimensional values." Ex. 1007 ¶ 18.

Although Patent Owner cites the ITC's Notice of Review as indicating that bump mapping is a form of texture coordinate shading (PO Resp. 14 (citing Ex. 2008, 3)), Patent Owner argues Petitioner's expert in the ITC proceeding acknowledged that "bump mapping can be performed without performing texture coordinate shading" (*id.* at 32–33 (citing Ex. 2001, 1229:8–14)). Even if "bump mapping" can be performed without texture coordinate shading, there is no evidence to exclude texture coordinate

shading as a procedure for performing "bump mapping." *See* Pet. Sur-reply To Mot. To Amend, 5 (citing Ex. 1013, 75:1–7, 302:9–13, 303:15–24 (ITC testimony of Dr. Edwards that certain types of bump mapping do constitute texture coordinate shading)). Collodi explicitly discloses that, in a preferred embodiment, the generation of a two dimensional bump map vector to be passed from the Programmable Shading Unit to a Vector Shading Unit is not limited to texture look-up, stating "[t]he Programmable Shading Unit may be programmed to generate the bump map procedurally or provide the bump map vector as a result of a texture map lookup (or a combined result of multiple texture map lookups)." Ex. 1007, ¶ 18.

As Petitioner points out, Collodi's environmental bump mapping modifies texture coordinates, thereby conforming to Patent Owner's definition of texture coordinate shading. Pet. Sur-Reply To Mot. To Amend, 5. Petitioner notes that in Collodi's disclosure of bump mapping, texture coordinates (u, v) obtained from a texture map lookup are modified by a height value "h," and that the bump map vector can be modified multiple times by additional texture map lookups. Opp. To Mot. To Amend, 20 (citing Ex. 1007 ¶¶ 18–19; Ex. 1014 (Declaration of Dr. Hanspeter Pfister In Support of Petitioner's Reply ("Supp. Pfister Decl.") ¶¶ 86–99). In addition, bump map vector b can be combined with an interpolated normal vector n to produce composite surface angle vector c that is used to modify the final pixel value when the vector is used in another texture map lookup. *Id.* Thus, even under Patent Owner's definition of "texture shading" as "texture coordinate shading," which requires modification of texture coordinates, that feature is disclosed in Collodi.

Collodi also discloses that diffuse and spectral color vectors combined through vector addition to produce combined vectors in the Accumulation Unit can be generated in the Programmable Shading Unit. Ex. 1007 ¶ 35. In any case, these vectors are used to modify the pixel color in the Programmable Shading Unit. *Id*. ¶ 36. Thus, even if we limit "texture shading" to "texture coordinate shading" requiring the modification of texture coordinates or addresses as proposed by Patent Owner, Collodi discloses such modification in the Programmable Shader Unit that performs both color shading and texture shading.

Motivation to Combine the Teachings of Rubinstein and Collodi

Citing Figure 1, Petitioner notes that Rubinstein discloses graphics processor 102 that converts 3D graphics data into color pixels to be displayed on its screen and includes geometry processor 110 that receives graphics data from a CPU, i.e. a front-end. Pet. 23–24 (citing Ex. 1004, 3:10–12, 3:16–18, Fig. 1A). As Petitioner notes, graphics processor 102 includes binning engine 112 that determines which tiles are touched by each geometry. *Id*. at 24 (citing Ex. 1004, 3:55–63, Fig. 1A). Petitioner also states that Rubinstein discloses a pipeline containing a raster unit for rasterization, a pixel shading unit for pixel shading and pixel texture operations, and a color buffer for blending operations. *Id*. at 25. According to Petitioner, the graphics processor in Rubinstein is "nearly identical" to that of the '506 patent. *Id*.

Petitioner argues that a person of ordinary skill would have been motivated to combine the teachings of Rubinstein and Collodi for three reasons. First, Rubinstein and Collodi disclose complementary pieces of a whole graphics pipeline. Pet. 27. Petitioner notes that Rubinstein performs

per-pixel shading operations but does not limit how that shading is accomplished, thus motivating a person of ordinary skill to look for references disclosing programmable shaders that could implement the Rubinstein pipeline. *Id.* at 27–28. According to Petitioner, Collodi discloses using a programmable per-pixel shading unit with improved functionality in 3D graphics hardware like that disclosed by Rubinstein. *Id.* at 28.

Second, Petitioner notes that Collodi discloses a per-pixel shading unit intended to improve existing 3D graphics pipelines, like that disclosed in Rubinstein, by offering parallel processing support for per-pixel lighting computations. *Id*. (citing Ex. 1007 ¶¶ 4, 5, 13).

Third, according to Petitioner, incorporating Collodi in Rubinstein constitutes a simple substitution of Rubinstein's shader with Collodi's programmable shader that would have yielded predictable results, i.e. an improve graphics processor with multiple parallel pipelines. *Id*. (citing Ex. 1003, Pfister Decl., ¶¶ 71–72). Petitioner emphasizes that Collodi and Rubinstein use similar pipelines that include a rasterizer, shading module, and texture memory for texture lookups. *Id.*

Patent Owner argues that the architectures of Rubinstein and Collodi are incompatible (PO Resp. 35–37) because Collodi does not teach how to replicate pipelines and because Rubinstein's Z-buffers, incorporated by reference from U.S. Patent 6,697,063 B1 ("Zhu '063") (Ex. 1005), prevent the hypothetical design from being implemented. PO Resp. 35–36. As to pipeline replication, Petitioner points out that Patent Owner's expert, Dr. Wolfe, was an inventor on a prior art patent and an author of a related article that included multiple pipelines in a graphics processor, and Dr. Wolfe

testified that parallelism could include replication of units or replication of entire processors. Pet. Reply 23 (citing Ex. 1023 (U.S. Patent 7,079,133 B2), Abstract, 6:61–65; Ex. 1022 (A. Wolfe and D.B. Noonburg, *A Superscalar 3D Graphics Architecture*, Proceedings of 32nd Annual Symposium on Microrarchitecture. Nov. 16–18, 1999); Ex. 1018, Wolfe Depo. Tr. 26:14–23).

Patent Owner also states that Petitioner relies on Zhu "which is incorporated by reference in Rubinstein, and therefore is tenuous at best to describe Rubinstein's functionality." PO Resp. 36. Patent Owner's comments concerning Zhu's "tenuous" link to Rubinstein's functionality completely contradict Patent Owner's arguments that the functionality of "texture shading" in claim 1 of the '506 should be construed to mean "texture coordinate shading" because of the purported incorporation by reference of the application that led to the '133 patent. As to Zhu '063, Patent Owner argues that no combination of Rubinstein and Zhu deals with specific conditions and requirements that must be satisfied prior to using its z-buffering approach that includes Early–Z and/or Heirarchical Z. *Id.* at 36 (citing Ex. 1005, 3:6-36, 10:57-65; Ex. 2009, Wolfe Resp. Decl., ¶ 115). Petitioner responds that Patent Owner's citation of Rubinstein's discussion of possible problems in the prior art is not germane to requirements that must be met to implement the system of Rubinstein/Zhu and that Rubinstein resolves possible problems by its "screen space tiling" technique. Pet. Reply 21–22 (citing Ex. 1005, 10:66–11:6). The combination of Rubinstein and Collodi is suggested by the disclosure in Collodi of improving the pipeline such that "rather than having separate hardware units for textures and colors everything is done using a single superpipeline" and "[s]creen z's receive

special treatment to take advantage of shading after z-buffering." *See* Ex. 1005, 36:60–37:3.

Patent Owner also argues that combining Rubinstein and Collodi would render Rubinstein inoperable for its intended purpose. PO Resp. 37–38. Patent Owner argues that Rubinstein discloses a single design for a screen space tiler and a single design for a memory controller that support only a single pipeline processing a single tile and that there is no description of how the per-pixel processor of Collodi, which operates on an entire frame, can be replicated in a combination with Rubinstein, which operates on tiles. PO Resp. 37–38. Petitioner responds that Patent Owner's expert Dr. Wolfe fails to address the capabilities of Rubinstein. Pet. Reply 24. For example, Rubinstein's tile management unit includes a plurality of screen tiles and is designed to allow for parallel processing of frames comprised of tiles, such that additional frames could be added for additional parallel processing with the Binning Engine distributing these tiles into memory that can be accessed by processing pipelines. *Id.* (citing Ex. 1004, 3:55–67, 6:29–31, 7:8–18). We also credit the testimony of Petitioner's expert, Dr. Pfister, that distributed frame buffers, one for each pipeline, was a concept well-known to those of ordinary skill and the choice to implement one frame buffer or multiple frame buffers would be a matter of design choice. *Id.* at 25. We also agree with Petitioner that a person of ordinary skill would have recognized that the texture unit within each rasterization pipeline would have sufficient texture memory to store textures to be used by the unified shader for current texture operations and that there is no need to coordinate depth date in the buffers between pipelines because the only data stored in each depth buffer is for the tile being processed by the pipeline. *Id.* at 26.

Patent Owner further argues that Collodi teaches away from the combination with Rubinstein. PO Resp. 38–39. Noting that Collodi explains that the cost of general purpose pixel processors in massively parallel architectures outweighs their benefits, Patent Owner argues that Collodi suggests improving the hardware to accelerate lighting calculation, rather than the use of texture shading. *Id*. As Petitioner points out, however, Collodi also states that those skilled in the art would recognize that multiple copies of the processor may be used in parallel within a graphics system. Pet. Reply 27 (citing Ex. 1007 ¶ 14). Patent Owner also argues that Collodi discloses duplicating both the front and back end, as opposed to the back end only. Noting that Patent Owner's expert, Dr. Wolfe, acknowledges that Collodi's design is suitable for connection after a binning and tiling unit, Petitioner argues that such an architecture is similar to that of the '506 patent. *Id*. at 27–28 (citing Ex. 2009 ¶¶ 105, 109).

The teaching-away analysis is a flexible one, under which "some references may discourage more than others," and the court must make an "entirely factual determination[]" about "whether there exist reasons a skilled artisan would combine or reasons a skilled artisan would not combine." *Arctic Cat Inc. v. Bombardier Recreational Prods. Inc.*, 876 F.3d 1350, 1360–61 (Fed. Cir. 2017). A reference does not teach away if it merely expresses a general preference for an alternative configuration, but does not "criticize, discredit, or otherwise discourage" investigation into the invention claimed. *In re Fulton*, 391 F.3d 1195, 1201 (Fed. Cir. 2004).

Petitioner notes that, although the testimony of Patent Owner's expert, Dr. Wolfe, often states that the references do not teach how to perform a particular implementation or modification, Dr. Wolfe never testified that it

would have been outside the ability of one of ordinary skill to combine the references and to make any necessary modifications. Pet. Reply 17. The evidence of record does not persuade us that Collodi teaches away from Petitioner's proposed combination with Rubinstein.

Having considered the arguments and evidence of record, we are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the combination of Rubinstein and Collodi discloses all of the elements of claim 1, that a person of ordinary skill would have been motivated to combine the teaching of Rubinstein and Collodi, and that claim 1 is, therefore, unpatentable.

Claim 2

Claim 2 depends from claim 1 and recites that each of the parallel pipelines comprises "a FIFO unit for balancing said each of said pipelines." Noting that at column 1, lines 6–15, Rubinstein incorporates by reference U.S. Patent No. 6,697,063 B1 (Ex. 1005, "Zhu '063") drawn to a rendering pipeline, Petitioner cites the explicit disclosure in Zhu '063 that the purpose of the FIFO between the scan/z engine and the raster/blending engine is load balancing between scan/z computation and raster/shading computation in a pipeline. Pet. 47 (citing Ex. 1005, 13:4–7). In view of Collodi's disclosure of multiple pipelines, Petitioner argues that the combination of Rubinstein and Collodi renders claim 2 unpatentable as obvious. *Id*. at 47–48.

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 2. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 2 are disclosed by the combination of Rubinstein and Collodi and that claim 2 is unpatentable.

Claim 3

Claim 3 depends from claim 1 and recites that each of the parallel pipelines comprises "a z buffer logic unit" and "a color buffer logic unit." As discussed above, in this Decision, we apply Petitioner's proposed construction of "z buffer logic unit" to mean "a logic unit that facilitates visibility testing by comparing depth values." As to the claimed z buffer logic unit, Petitioner cites material incorporated by reference from Zhu '063 disclosing a double z scheme to resolve visibility and allow the rest of the rendering pipeline to perform set up for only visible primitives and shade only visible fragments. Pet. 49–50 (citing Ex. 1005, Abstract). As to the claimed color buffer logic unit, Petitioner cites Rubinstein's disclosure of sending processed fragments to a blending engine that resolves and outputs pixel colors into the frame buffer at the end-of-tile. *Id.* at 51. Petitioner also notes that Zhu '063 discloses a color buffer to accumulate colors. *Id.* at 52 (citing Ex. 1005, Fig. 14, 24:44–49, 24:50–55).

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 3. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 3 are disclosed by the combination of Rubinstein and Collodi and that claim 3 is unpatentable.

Claim 4

Claim 4 depends from claim 3 and recites that "said z buffer logic unit interfaces with said scan converter through a hierarchical Z interface and early Z interface." As discussed above, for purposes of this Decision, we apply Petitioner's proposed construction of "hierarchical Z interface" to mean "an interface with a z-buffer logic unit that provides for visibility testing at a coarse level, including, for example, for an entire tile or primitive." We also apply Petitioner's proposed construction of "early Z

interface" to mean "an interface with a z buffer logic unit that provides for visibility prior to shading and texturing."

Noting that Rubinstein discloses the visibility of all geometry is determined upfront using only screen x, y, z coordinates in a scan/z engine for each tile, Petitioner cites Rubinstein as disclosing a Z buffer logic unit that performs the same function as the scan converter in the Z buffer logic unit in the '506 patent. Pet. 53. As to the claimed early Z interface, citing Zhu '063, incorporated by reference into Rubinstein, Petitioner notes that z buffering is done early enough in the pipeline that the set up computation for rasterizing surface parameters need be performed only for visible primitives and visible fragments. *Id*. at 55 (citing Ex. 1005, 10:5–11).

As to the hierarchical Z interface, Petitioner also notes that Zhu '063 expressly discloses that "[d]ouble-z can be used in conjunction with hierarchical z-buffering. Under hierarchical z-buffer, the second pass double-z is extremely efficient because all invisible fragments are rejected in the z-testing states (possibly at very coarse levels of hierarchical z-buffer)." *Id*. at 55 (citing Ex. 1005, 12:20–24).

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 4. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 4 are disclosed by the combination of Rubinstein and Collodi and that claim 4 is unpatentable.

Claim 5

Claim 5 depends from claim 3 and recites that "said z buffer logic unite interfaces with said unified shader through a late Z interface." Petitioner notes that Rubinstein discloses a pipeline that implements a "double-z" scheme, i.e., one that performs z-buffering two times within the

same pipeline (early and late). Pet. 56. In particular, Petitioner points out that Rubinstein discloses that z-buffering operations can also occur after shading operation. *Id*. at 56−67 (citing Ex. 1004, Fig. 1A). Petitioner also explains that Zhu ’063, incorporated by reference into Rubinstein, discloses that the late z-interface is used for special cases that are not filtered out during early z buffering. *Id*. at 57 (citing Ex. 1005, 20:3−4). Petitioner further asserts that a person of ordinary skill in the art would have understood that these z-buffering operations are done by a z buffer that interfaces with the shader through a late z-interface because they occur after shading. *Id*. (citing Ex. 1003, Pfister Decl., ¶ 111).

Patent Owner does not respond explicitly to Petitioner’s contentions concerning claim 5. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 5 are disclosed by the combination of Rubinstein and Collodi and that claim 5 is unpatentable.

Claim 6

Claim 6 depends from claim 1 and recites that the “graphic[s] chip” comprises “a setup unit for directing said geometry into one of said multiple parallel pipelines wherein said geometry is determined to locate in a portion of an output screen defined by a tile.” The portion of claim 6 reciting “wherein said geometry is determined to locate in a portion of an output screen defined by a tile” is identical to claim limitation 1-d and is discussed above. According to Petitioner, a person of ordinary skill would have understood Rubinstein’s binning engine that divides screens into tiles and identifies geometry that touches each tile to constitute the claimed setup unit. Pet. 58. Petitioner argues that a person of ordinary skill would have been motivated to apply the teachings of Collodi to Rubinstein’s graphics

process or to use multiple rendering pipelines in parallel to process different portions of the screen by accessing specific bins created by the binning engine. *Id.* at 59–60.

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 6. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 6 are disclosed by the combination of Rubinstein and Collodi and that claim 6 is unpatentable.

Claim 7

Claim 7 depends from claim 1 and recites that the pipeline further comprises a scan converter, a rasterizer, and a texture unit. Petitioner notes that Rubinstein discloses a scan/z engine that performs the same functions as the claimed scan converter by stepping through the primitives/geometries and determining z and depth information for the z buffer logic unit, and a rasterizer. Pet. 63 (citing Ex. 1004, 3:34–36). As to the texture unit, petitioner cites Collodi as disclosing hardware for requesting texture information from memory using texture coordinates (performing texture lookups). *Id.* at 62–63 (citing Ex. 1007 ¶¶ 12, 15; Ex. 1003, Pfister Decl., ¶ 121).

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 7. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 7 are disclosed by the combination of Rubinstein and Collodi and that claim 7 is unpatentable.

Claim 8

Claim 8 depends from claim 1 and recites that "the unified shader is operative to apply a programmed sequence of instructions to rasterized values and is operative to loop back to process operations for color shading

and/or texture address shading." As to a unified shader operative to apply a programmed sequence of instructions to rasterized values, Petitioner cites Collodi as disclosing a programmable pixel shader that applies instructions and operations on the outputs of a rasterizer. Pet. 64 (citing Ex. 1007, Fig. 1). As to a shader operative to loop back to process operations for color shading and/or texture address shading, Petitioner cites Zhu '063, incorporated by reference into Rubinstein, as disclosing a pipeline can perform multiple passes and accumulate the results in a buffer to resolve color contribution for multiple passes for fragment. *Id.* at 65. Petitioner also argues that Collodi's disclosure of a programmable shader that can send its output through the lighting hardware and use it as feedback also teaches this feature. *Id.* at 66.

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 8. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 8 are disclosed by the combination of Rubinstein and Collodi and that claim 8 is unpatentable.

Claim 9

Claim 9 depends from claim 1 and recites that "the unified shader is operative to apply a programmed sequence of instructions such that a first set of instructions are executed to perform color shading and a second set of instructions are executed to perform texture shading." Petitioner cites Collodi as disclosing a programmable pixel shader capable of performing both color shading operations and texture shading operations, satisfying the limitations of claim 9 of a shader that is capable of applying a programmed sequence of instructions in which the first set of instructions perform color shading in the second set of instructions perform texture shading. Pet. 67–

68 (citing Ex. 1007 ¶¶ 3, 12, 14, 15, Fig. 1; Ex 1003, Pfister Decl., ¶¶ 127–130).

Patent Owner does not respond explicitly to Petitioner's contentions concerning claim 9. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 9 are disclosed by the combination of Rubinstein and Collodi and that claim 9 is unpatentable.

Obviousness of Claims 8–9 Over Rubinstein in view of Collodi and further view of Zatz

Motivation to Combine

Zatz discloses a programmable shader controllable by both state-based control information and program instructions. Ex. 1008, Abstract. Petitioner cites Zatz as disclosing a programmable shader that performs both color shading and texture shading and includes a loop back path for multi-pass operations. Pet. 68. Petitioner argues that a person of ordinary skill would have been motivated to combine the programmable shader of Zatz with the teachings of Rubinstein and Collodi to avoid the limitations associated with state-based control method and to use various control formats for controlling a fragment processor. Pet. 69–70.

Noting that the Zatz application was filed on December 13, 2002, i.e., after the '506 patent priority date of November 27, 2002, Patent Owner argues that Petitioner has not demonstrated that Zatz is prior art because the Petition fails to show Zatz is entitled to the § 102(e) date of its provisional application (July 18, 2002). PO Resp. 42. Petitioner notes that the Petitioner Reply is an appropriate vehicle for responding to a priority date challenge in the Patent Owner Response. Pet. Reply 30–31 (citing *Dynamic Drinkware, LLC v. Nat'l Graphics, Inc.*, 800 F.3d 1375, 1378–1381 (Fed.

Cir. 2015)). In its Reply, Petitioner responds by citing the Supplemental Declaration of its expert, Dr. Pfister. *Id.* (citing Ex. 1014, Pfister Suppl. Decl. ¶¶ 169–197). Dr. Pfister testifies that the relevant subject matter in the ’506 patent is not entitled to the priority Patent Owner claims because the provisional application from which the ’506 patent claims priority did not include a discussion of a unified shader that is operative to loop back to process operations. Ex. 1014, Pfister Supp. Decl. ¶¶ 176–177. We agree. As we discuss earlier in this Decision, the application that matured into the ’506 patent did not include a discussion of the unified shader. Noting that Zatz filed the provisional application on July 18, 2002, Dr. Pfister identifies subject matter from the Zatz provisional application and the Zatz patent illustrating a loopback circuit. *Id.* ¶ 182. In consideration of the above, we are persuaded that Petitioner has shown that Zatz is prior art.

Patent Owner also argues that Zatz cannot be combined with Rubinstein and Collodi because Zatz discloses a single pipeline that includes a conventional color shader controlled by a state controller as used with DirectX8 or a program-based controller as used with DirectX9. PO Resp. 43. According to Patent Owner, Petitioner’s argument that Zatz’s Return Path 276 discloses the loopback element in the ’506 patent fails to explain how to combine Return Path 276 to other pipeline structures. *Id.* (arguing “The specific exemplary Return Path 276 in Zatz is not applicable to the alleged Rubinstein and Collodi combination because the same structure are not within the loop and the same inputs and outputs to the feedback loop are not present in the alleged Rubinstein and Collodi combination.” (citing Ex. 2009, Wolfe Resp. Decl. ¶ 117)). Dr. Wolfe testifies that Return Path 276 of Zatz connects Combiner Unit 270, a programmable block that is

separate and distinct from the shader, to the input of Gate Keeper 220, which regulates the entry of data to the Programmable Shader Pipeline Data Path. Ex. 2009 ¶ 117. Dr. Wolfe further testifies that neither a gatekeeper, nor a combiner unit, are present in Petitioner's proposed combination. *Id.* Dr. Wolfe also states that Petitioner's proposed combination indicates the rasterizer is inside the loop back path, as opposed to outside the loop back path in Zatz's teachings. *Id.*

Petitioner responds to this argument by citing the Supplemental Declaration of its expert, Dr. Pfister. Pet. Reply 30–31 (citing Ex. 1014, Pfister Suppl. Decl., ¶¶ 168). Dr. Pfister notes that the absence of a combiner and gatekeeper in Rubinstein and Collodi is not determinative because a person of ordinary skill would have recognized that the loopback technique could have been implements with different architectures not having the same components with no more that routine experimentation. Ex. 1014, Pfister Supp. Decl., ¶ 168. As Zatz is cited for teaching the use a loopback approach, rather than for its specific implementation, we agree with Petitioner that a person of ordinary skill would have been motivated to combine the teachings of Zatz with those of Rubinstein and Collodi.

Patent Owner also argues that Zatz Texture 240 only performs texture fetching and does not perform texture shading. PO Resp. 43–44. Patent Owner's argument is not consistent with our construction of the claimed "texture shading." In addition, Petitioner responds to this argument by citing the Supplemental Declaration of its expert, Dr. Pfister to demonstrate that texture coordinate shading, as defined by Petitioner, is disclosed by Zatz. Pet. Reply 30–31 (citing Ex. 1014, Pfister Suppl. Decl., ¶¶ 198–206).

In view of the similarity of the subject matter, we are persuaded that Petitioner has cited sufficient evidence that a person of ordinary skill would have been motivated to combine the teachings of Rubinstein, Collodi, and Zatz.

Claim 8

Claim 8 depends from claim 1 and recites that "the unified shader is operative to apply a programmed sequence of instructions to rasterized values and is operative to loop back to process operations for color shading and/or texture address shading." As discussed above, we are persuaded that Petitioner has cited sufficient evidence to demonstrate that the combination of Rubinstein and Collodi disclose a unified shader operative to apply a programmed sequence of instructions to rasterized values. Noting that Zatz discloses rasterizer 144 performs scan conversion and outputs fragment, pixel, or sample data to fragment processing pipeline 147, which operates on the data, Petitioner contends that, like Collodi, Zatz discloses a programmable pixel shader that applies instructions to rasterized values. Pet.72 (citing Ex. 1008, 9:34–36).

To the extent that Rubinstein is found not to disclose the "loop back" limitation of claim 8, Petitioner cites Figure 2 of Zatz and its accompanying description as teaching a programmable shader that processes data in multiple passes through its computation units. *Id.* at 74.

Except as discussed above concerning the motivation to combine Zatz with Rubinstein and Collodi, Patent Owner does not respond explicitly to Petitioner's contentions concerning the combination of the references, as applied to claim 8. We are persuaded that Petitioner has demonstrated by a

preponderance of the evidence that the elements of claim 8 are disclosed by the combination of Rubinstein and Collodi and that claim 8 is unpatentable.

Claim 9

Claim 9 depends from claim 1 and recites that "the unified shader is operative to apply a programmed sequence of instructions such that a first set of instructions are executed to perform color shading and a second set of instructions are executed to perform texture shading." In the context of claim 9, Petitioner cites Zatz as disclosing "[t]ypically, the one or more programs are part of a shader program written to accomplish a certain shading effect." Pet. 76 (citing Ex. 1008, 6:64–67). Petitioner further notes that Zatz also discloses that color shading instructions are typically executed prior to texture shading instructions. *Id.* (citing Ex. 1008, 11:6–14).

Except as discussed above concerning the motivation to combine Zatz with Rubinstein and Collodi, Patent Owner does not respond explicitly to Petitioner's contentions concerning the combination of the references, as applied to claim 9. We are persuaded that Petitioner has demonstrated by a preponderance of the evidence that the elements of claim 9 are disclosed by the combination of Rubinstein and Collodi and that claim 9 is unpatentable.

MOTION TO AMEND

Introduction

Patent Owner proposes to substitute amended claim 22 for independent claim 1 and to substitute dependent claims 23–30, which depend directly or indirectly from claim 22 for claims 2–9, which depend directly or indirectly from claim 1. The substance of Patent Owner's Motion to Amend is to limit further the "unified shader" as follows:

> wherein the unified shader comprises a single shader circuit capable of performing both color shading and texture coordinate shading, wherein texture coordinate shading modifies a texture coordinate.

Mot. To Amend 5–6.

Written Description Support

When filing a Motion to Amend, Patent Owner must set forth the support in the *original disclosure* of the patent for each proposed substitute claim, i.e., Patent Owner must identify clearly the written description support in the disclosure corresponding to the earliest date upon which Patent Owner seeks to rely. 37 C.F.R. §§ 42.121(b)(1), 42.121(b)(2); *see Lectrosonics, Inc. v. Zaxcom, Inc.*, Case IPR2018-01129, -01130, slip op. at 7−8 (PTAB Feb. 25, 2019) (Paper 15) (precedential). Patent Owner contends that: (1) the proposed substitute claims are supported by the original application that became the ’506 patent and (2) the substitute claims also have written description support in the ’133 patent that was incorporated by reference in its entirety into the ’506 patent and original U.S. application 10/730,965 (“the ’965 application”) from which the ’133 patent claims priority. Mot. To Amend 4. We understand Patent Owner’s assertion that the claims are supported by the original application that became the ’506 patent to be the earliest date on which Patent Owner relies for written description support of the substitute claims.

Petitioner contends that because the ’965 application was not part of the original disclosure and did not exist when the application that led to the ’506 patent was filed, it cannot provide written support for the claims. Opp. To Mot. To Amend 6–7 (citing *Ariad Pharm. Inc. v. Eli Lilly* Co., 598 F.3d 1336, 1355 (Fed. Cir. 2010) for the proposition that anything that occurs

after the filing date of the patent is irrelevant to whether the original, as-filed specification provides written description support for the claims).

Petitioner also notes that essential material can only be incorporated by reference by way of a U.S. patent or a U.S. patent application publication. Opp. To Mot. To Amend 9–10 (citing 37 C.F.R. § 1.57(c)). Petitioner contends, therefore, that even if the ’133 patent application had existed at the time of filing, because it was filed with a nonpublication request (and issued after the ’506 patent), the application that led to issuance of the ’133 patent was not published before the ’506 patent issued and could not provide written description support. Opp. To Mot. To Amend 10.

Petitioner also contends that, although the purported incorporation by reference is of the entire ’965 application that led to the ’113 patent, Patent Owner’s choice to add verbatim the material in the ’506 patent that now appears at column 9, line 28 through column 14, line 13 concerning the structure, architecture, and operations of a unified shader, indicates that other material concerning texture coordinate shading and indirect texturing is not applicable to the ’506 patent. Opp. To Mot. To Amend 11.

Earlier in this Decision, we noted that the application that matured into the ’506 patent did not describe the functionality of the unified shader (other than to state it performs color shading and texture address shading), but instead sought to incorporate that description by reference to an application that had not been filed and could not be identified. Ex. 1017, 307. The ’506 patent as originally filed does not mention “texture coordinate” or “texture coordinate shading.” *See* Ex. 1017, 295–326. Patent Owner also does not identify where the application that led to issuance of the

’506 patent as originally filed describes texture coordinate shading as modifying a texture coordinate.

We extensively addressed the purported incorporation by reference of the ’133 patent into the ’506 patent. The application for the ’506 patent filed on November 26, 2003 included several attempts to incorporate by reference one or more applications identified as having serial number 10/xxx,xxx filed on December XX, 2003. Thus, the application that issued as the ’506 patent attempted to incorporate by reference unidentified applications that had not yet been filed. We determined that the purported incorporation by reference was inadequate at least until Patent Owner filed an amendment in 2007 identifying two such applications. Ex. 1017, 252; *See* 37 C.F.R. § 1.57(c). It is immaterial that the ’965 application that led to the ’133 patent was a continuation of an application filed several days before the filing date of the application that led to the ’506 patent. The fact remains that no application was actually identified until 2007. Moreover, there is no evidence that the ’965 application or its parent were published before issuance of the ’506 patent. Thus, it is not clear that the subject matter from the applications that led to issuance of the ’133 patent was incorporated by reference before the ’506 patent issued.

We are also not persuaded by Patent Owner’s arguments that Petitioner acknowledged the incorporation by reference in the ITC proceeding. Reply to Opp. To Mot. To Amend 1–2. Whether the issue was brought to the attention of the ITC is not a matter for this proceeding, where it clearly has been made an issue.

Patent Owner argues that the mention of texture address shading in the original ’506 patent is sufficient written description support because

"texture address shading" and "texture coordinate shading" refer to the same process. *Id.* at 3. However, Patent Owner does not propose substitute claims that recite the language in the '506 patent Specification, i.e., texture address shading. Instead, Patent Owner seeks to insert the term "texture coordinate shading" from the '133 patent. We are unpersuaded by Patent Owner's argument that Petitioner's ITC expert acknowledges the equivalence of these terms. Reply to Opp. To Mot. To Amend 3, 6 (citing Ex. 2001, 1220:12–14, 17–18). Much of that testimony was offered on December 1, 2017, in the context of claim constructions already being applied in the ITC (*see* Ex. 2005 claim construction order entered November 8, 2017), where the incorporation by reference of the '965 application went unchallenged. That is not the case in this proceeding.

In view of the circumstances, we are not persuaded by Patent Owner's assertion that the substitute claims are supported by the original application that became the '506 patent (Mot. To Amend 4) and we deny the Motion to Amend as lacking written description support.

Texture Coordinate Shading

We also consider whether "texture coordinate shading" recited in the proposed substitute claims is disclosed by the references. Petitioner asserts that texture coordinates are in texture space relative to location (0, 0) in the texture image, but that texture addresses point to a specific texel in a screen and can be used to fetch a texture from memory. Opp. To Mot. To Amend 16 (citing Ex. 1014, Pfister Suppl. Decl. ¶¶ 25–32). Thus, unlike texture coordinates, texture addresses are not relative values and are not normalized to a limited range of (0, 1). *Id.* According to Petitioner, texture coordinate shading is an operation resulting in texture coordinates and not an operation

using texture addresses, as the ’506 patent describes. *Id.* at 17 (citing Ex. 1012, Pfister Suppl. Decl. ¶¶ 41–47).

To the extent that texture coordinate shading involves modifying texture coordinates, as urged by Patent Owner, Petitioner cites environment mapped bump mapping disclosed by Collodi as disclosing texture coordinate shading. Opp. To Mot. To Amend 17–20. As discussed earlier in this Decision, Collodi explicitly discloses that in a preferred embodiment the generation of a two dimensional bump map vector to be passed from the Programmable Shading Unit to a Vector Shading Unit is not limited to texture look-up, stating “[t]he Programmable Shading Unit may be programmed to generate the bump map procedurally or provide the bump map vector as a result of a texture map lookup (or a combined result of multiple texture map lookups).” Collodi’s environmental bump mapping modifies texture coordinates, thereby conforming to Patent Owner’s definition of texture coordinate shading. Sur-Reply To Mot. To Amend, 5. Petitioner notes that in Collodi’s disclosure of bump mapping texture coordinates (u, v) obtained from a texture map lookup are modified by a height value “h,” and that the bump map vector can be modified multiple times by additional texture map lookups. Opp. To Mot. To Amend, 20 (citing Ex. 1007 ¶¶ 18–19; Ex. 1014, Supp. Pfister Decl., ¶¶ 86–99), In addition, bump map vector b can be combined with an interpolated normal vector n to produce composite surface angle vector c that is used to modify the final pixel value when the vector is used in another texture map lookup. *Id.* Thus, even under Patent Owner’s definition of “texture shading” as “texture coordinate shading,” which requires modification of texture coordinates, that feature is disclosed in Collodi.

To the extent that texture coordinate shading involves modifying texture coordinates, Petitioner cites Rich's disclosure of perspective correction in which texture coordinates (u, v) are modified by the Processing Element by dividing the texture coordinates by a depth value w prior to using those coordinates to retrieve texture map information. Opp. To Mot. To Amend 20–21; *see also* -00101 IPR Opp. To Mot. To Amend 17 (citing Ex. 1012, Pfister Suppl. Decl., ¶¶ 70–75). Patent Owner rebuts Petitioner's position by arguing that in the ITC, Dr. Edwards testified that perspective correction is part of generation and cannot be texture coordinate shading. Reply to Opp. To Mot. To Amend 7; *see also*, -00101 IPR Reply to Opp. To Mot. To Amend 7 (citing Ex. 2001, 1220:22–1223:6, 1224:9–16). However, Dr. Edwards' testimony is offered in the context of different claim constructions and in a forum that requires clear and convincing evidence of invalidity, as opposed to this proceeding where the standard is preponderance of the evidence that a claim is unpatentable.

We further note that Patent Owner cites the Commission's opinion that texture coordinate shading broadly means any modification to texture coordinates, which can be accomplished by the unified shader through indirect texturing, bump mapping, or texture address operations. *Id.* at 6 (citing Ex. 2042, 18–19). Petitioner notes that it is undisputed that Rich's perspective correction modifies texture coordinates by performing division on them. Sur-reply to Mot. To Amend, 7; *see also*, -00101 IPR Pet. Sur-Reply to PO's Reply to Opp. to Mot. To Amend 7. Petitioner points out that Figure 5 of Rich shows texture coordinates (u, v) undergoing perspective correction after color shading and before texture lookups, all of which occurs after generation of coordinates during rasterization. *Id.* at 8.

Thus, in the context of the proposed Motion to Amend, Petitioner has demonstrated that, even if we adopt Patent Owner's arguments that texture coordinate shading is disclosed in the '506 patent, to the extent that such texture coordinate shading comprises modifying the coordinate as recited in the substitute claim, modification of the texture coordinates in the unified shader is disclosed at least by Rich and Collodi. Therefore, we deny the Motion to Amend on this basis as well.

MOTION TO EXCLUDE

Patent Owner moves to exclude Exhibits 1037 and 1038 as irrelevant, prejudicial, and not properly authenticated. Mot. To Exclude 1–2. Patent Owner also argues that Petitioner did not cite Exhibits 1037 and 1038 in any pleading and that these exhibits were filed two weeks after the last paper in which Petitioner was allowed to submit evidence.

Petitioner states that it filed and served Exhibits 1037 and 1038 in response to Patent Owner's objections to Exhibit 1020 and to demonstrate that Exhibit 1020 was publically accessible. Resp. to Mot. to Exclude 1. As Patent Owner has not moved to exclude Exhibit 1020, Petitioner states that Exhibits 1037 and 1038 are not necessary. *Id.* at 2. We do not refer to Exhibits 1037 and 1038 in this Decision. In view of the circumstances, we dismiss Patent Owner's Motion to Exclude as moot.

MOTION TO SEAL

Patent Owner moves to seal the transcript of an August 30, 2018 conference call (Exhibit 2039 in each case) and requests entry of a stipulated protective order (i.e., the Board's Default Protective Order) on the grounds that pages 33:2 through 35:18 contain confidential business information. Patent Owner also notes that on September 14, 2008, it filed a redacted

transcript (Exhibit 2034). Petitioner does not oppose entry of the protective order or Patent Owner's Motion to Seal. In consideration of the circumstance, Patent Owner's Motion to Seal is granted.

CONCLUSION

Having reviewed the pleadings and the evidence of record, we conclude that Petitioner has demonstrated by a preponderance of the evidence that: (i) challenged claims 1–9 of the '506 patent are unpatentable under 35 U.S.C. § 103 as obvious over the combination of Rubinstein and Collodi; and (ii) claims 8–9 are unpatentable as obvious over the combination of Rubinstein, Collodi, and Zatz.

We also deny Patent Owner's Motion to Amend as not supported by the written description of the '506 patent application as filed and as not patentable over the combination of Rubinstein and Collodi.

We dismiss Patent Owner's Motion to Exclude as moot.

We grant Patent Owner's Motion for entry of a protective order and to seal Exhibit 2039.

ORDER

In consideration of the above, it is

ORDERED that claim 1–9 of the '506 patent are unpatentable;

FURTHER ORDRED that Patent Owner's Motion to Amend is DENIED;

FURTHER ORDERED; that Patent Owner's Motion to Exclude is dismissed as moot;

FURTHER ORDERED that Patent Owner's Motion for entry of a Protective Order and to Seal Exhibit 2039 is GRANTED; and

FURTHER ORDERED, that because this is a final written decision, parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

PETITIONER

Kevin Anderson
kpanderson@duanemorris.com

Scott Felder
sfelder@wileyrein.com

PATENT OWNER

William Meunier
Michael Renaud
Adam Rizk
MINTZ LEVIN
wameunier@mintz.com
mtrenaud@mintz.com
arizk@mintz.com