

Michael T. Renaud
617 348 1870
mtrenaud@mintz.com



One Financial Center
Boston, MA 02111
617 542 6000
mintz.com

April 5, 2024

VIA EMAIL

Dax Terrill, Esq.
Chief, Exclusion Order Enforcement Branch
Regulations and Rulings, Office of Trade
U.S. Customs and Border Protection
90 K Street, NE, 10th Floor
Washington, DC 20229-1177

eoebbranch.itc337.rulings@cbp.dhs.gov
dax.terrill@cbp.dhs.gov
aaron.c.cook@cbp.dhs.gov

Re: Ruling Request Pursuant to 19 C.F.R. Part 177 by Realtek Semiconductor Corp.

Dear Mr. Terrill:

In response to the 19 CFR Part 177 Ruling Request on Behalf of Realtek Semiconductor Corp., please find a submission on behalf of Advanced Micro Devices, Inc. and ATI Technologies ULC.

Please note the parties are working together to provide bracketed copies of the confidential exhibits provided hereto.

Sincerely,

Michael T. Renaud

*Counsel for Complainants
Advanced Micro Devices, Inc.
and ATI Technologies ULC*

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. PROCEDURAL BACKGROUND.....	8
III. CATEGORIES OF PRODUCTS SUBJECT TO THE LEO	9
1. Standalone Realtek Integrated Circuits (ICs)	9
2. Realtek Development Boards	10
IV. TECHNOLOGY OVERVIEW	13
V. PATENT BACKGROUND	16
VI. LEGAL PRINCIPLES	22
1. Customs’ Scope of Review/Enforcement	22
2. Realtek’s Burden of Proof	24
3. Infringement of Apparatus Claims.....	26
VII. IDENTIFICATION OF THE ARTICLES SUBJECT TO REALTEK’S RULING REQUEST LETTER.....	28
VIII. REALTEK’S PATCHES TO THE DRIVER SOFTWARE DO NOT ALTER THE HARDWARE CONFIGURATION UNDERLYING THE COMMISSION’S FINDINGS OF INFRINGEMENT	29
1. The Commission Found that the Mali Driver is not Part of the Infringing Devices but Instead Serves as the Scheduling Module in the Environment in which the Infringing Devices Operate	30
2. The Commission Found that the Claimed Plurality of Engines in the Infringing Devices Were Implemented in Hardware— <u>not</u> Software	36
3. The Commission Found that the Claimed Shader Core in the Infringing Devices Was Implemented in Hardware— <u>not</u> Software	39
4. The Commission Found that Hardware— <u>not</u> Software—in the Infringing Devices Processed Compute Tasks and Graphics Tasks as Required by Claim 19.....	45
5. The Commission’s Decision in Connection with Prioritization for Claim 20 was Found in the Infringing Hardware Code and Not Just in the Software Driver.....	46
6. Realtek Admits that it does not make any changes to its Hardware Source Code that was Found to Infringe at the ITC.	48
IX. THE PATCHED MALI DRIVER DOES NOT OPERATE AS REALTEK ADVERTISED IN ITS RULING REQUEST LETTER	48

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

A.	Claim 19: The “New Mali Driver” Patches Simply Reconfigure Vertex Jobs to Initiate the Execution of General-Compute Tasks.....	49
1.	Realtek’s Ruling Request Letter Fails to Describe the Effect of the Performance Enhancement Patch Embedded in the RRL Articles and Source Code on Realtek’s Claims.....	51
2.	Despite Realtek’s Software Patches, the Ruling Request Letter Articles Continue to Infringe Claim 19	53
3.	The EOE Branch Should Deny Any Attempt By Realtek to Seek a Ruling Based on Further Modification to the Ruling Request Letter Articles or Argument of Non-Infringement of the Performance Enhancement Patch	60
B.	Claim 20: The “New Mali Driver” Patches Simply Create New Prioritization Schemes or Shift the Priority Considered by the System Without Eliminating Consideration of Priority that Infringe Claim 20	62
1.	Application Programmers May Implement Prioritization in Application Software to Circumvent the ARM Patches	64
2.	The Patches Only Partially Address the Priorities Considered in the Hierarchy of Prioritization of the Mali Driver and Mali GPUs	65
3.	The Patched Mali Driver Has Shifted Job Slot Prioritization to Job Slot 0 Over Job Slot 1 and Job Slot 2.....	67
4.	The Patches Have Not Disabled the Consideration of Priority by the Function Identified by AMD and Cited by the Commission (jsctx_rb_peek_prio) Which Prioritizes Jobs Based on Age or Sequence Number	69
5.	The Patched Mali Driver Prioritizes Jobs Based on Their Associated Context.....	70
6.	The GPU Hardware Includes Multiple Mechanisms to Prioritize Work Associated with Jobs for Receipt by the Execution Resources	70
C.	Claim 20: The ARM Behavioral Change Patches to Avoid Claim 19 Introduces Yet an Additional Priority Scheme That Infringes Claim 20	73
1.	Graphics Jobs Are Prioritized When a Graphics Job is Executing and Jobs Marked as Compute Jobs Are Prioritized When a Job Marked as Compute is Executing	74
X.	REALTEK HAS FAILED TO SATISFY ITS BURDEN OF PROOF GIVEN ITS DEFICIENT EVIDENTIARY PRODUCTION	78
A.	The Declarations of Mr. Larri, Mr. de la Iglesia, and Dr. Pfister Fail to Substantiate Realtek’s Claims.....	80
1.	Realtek Failed to Produce a Witness That Can Authenticate How the Source Code is Actually Implemented in Realtek Products	80

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

2.	Mr. Larri’s Declaration, Release Notes, and Testimony Should be Viewed as a Self-Serving Extension of Realtek’s Attorney Arguments ...	82
3.	Realtek’s Submission Amounts to Nothing More than Attorney Arguments That Are Not Supported by a Credible Expert Opinion Regarding Non-Infringement.....	85
4.	Realtek Failed to Produce the Relevant Documentation to Describe How the Patched Mali Driver Uses the Pilot Shader Programs	88
B.	Realtek Has Failed to Present the Evidence Necessary to Substantiate its Basis for Representativeness of the R26 DDK Driver Version and its Patches	90
C.	Realtek Failed to Produce Source Code to Substantiate its Claims that the RTD2851MPJ is Representative of the RTD2951M or How Sample Boards 2 and 3 Operate	93
D.	Realtek Failed to Produce Evidence that its Downstream Consumer Manufacturers Cannot Modify Realtek’s Driver Patches as They Wish.....	94
XI.	CONCLUSION.....	98

ABBREVIATIONS TABLE

Abbreviation	Document
'381 Patent	US Patent No. 8,854,381
CIB	ITC-1318 Complainant Initial Post-Hearing Brief
CX	Complaint's Exhibit from the ITC-1318 Evidentiary Hearing
FD	Final Determination - Investigation No. 337-TA-1318
Hrg. Tr.	ITC-1318 Evidentiary Hearing Transcript (03/13-17/2023)
ID	Initial Determination - Investigation No. 337-TA-1318
Iglesia Decl.	177 Proceeding Iglesia Decl. and Exhibits
Iglesia Tr. (01/06/2023)	ITC-1318 Deposition Transcript of Erik de la Iglesia (01/6/2023)
Iglesia Tr. (03/22/2024)	177 Proceeding Deposition Transcript of Erik de la Iglesia (03/22/2024)
JX	Joint Exhibit from the ITC-1318 Evidentiary Hearing
Larri Decl.	177 Proceeding Larri Decl. and Exhibits
Larri Tr. (03/19/2024)	177 Deposition Transcript of Guy Larri (03/19/2024)
Larri Tr. (11/01/2022)	ITC-1318 Deposition Transcript of Guy Larri (11/01/2022)
LEO	Limited Exclusion Order - Investigation No. 337-TA-1318
Order No. 31	ITC-1318 Order No. 31: Construing Terms of the Asserted Claims (Oct. 18, 2022)
Pfister Decl.	177 Proceeding Pfister Decl. and Exhibits
Pfister Tr. (03/25/2024)	177 Proceeding Deposition Transcript of Hanspeter Pfister (03/25/2024)
Reinman Decl.	177 Proceeding Reinman Declaration
RIB	ITC-1318 Respondent Initial Post-Hearing Brief
RRL	Realtek 177 Ruling Request Letter

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

Exhibit No	Document Title	Doc ID Trial Exhibit No	End Bates No.
1	177 Proceeding Reinman Declaration		
2	Certified US Patent No. 8,854,381	JX-0003	
3	Realtek 177 Request Letter		
4	177 Proceeding Erik de la Iglesia Decl.		
5	177 Proceeding Pfister Decl.		
6	177 Proceeding Larri Decl.		
7	Limited Exclusion Order - Investigation No. 337-TA-1318	EDIS ID 81255	
8	Initial Determination - Investigation No. 337-TA-1318	EDIS ID 799959	
9	Final Determination - Investigation No. 337-TA-1318	EDIS ID 812552	
10	ITC-1318 Order No. 31: Construing Terms of the Asserted Claims, October 18, 2022	EDIS ID 782559	
11	ITC-1318 Evidentiary Hearing Transcript (03/13-17/2023)	EDIS IDs 792389, 792472, 792626, 792642, 792753	
12	177 Deposition Transcript of Guy Larri (03/19/2024)		
13	177 Deposition Transcript of Erik de la Iglesia (03/22/2024)		
14	177 Deposition Transcript of Hanspeter Pfister (03/25/2024)		
15	ITC-1318 Deposition Transcript of Guy Larri (11/01/2022)	JX-0192C	
16	Release Note: OpenGL ES Performance enhancement DDK Patches for Mali Bifrost GPUs	RTK-177-000001	RTK-177-000008
17	no file path - doc title # Device Naming	RTK-177-000009	RTK-177-000009
18	Boilerplate Realtek SOC Software License Agreement	RTK-177-000010	RTK-177-000010
19	Boilerplate Realtek Copyright License	RTK-177-000011	RTK-177-000011
20	Boilerplate Dolby and GNU General Public Licenses	RTK-177-000012	RTK-177-000020
21	no file path - doc title Kernel/system	RTK-177-000021	RTK-177-000021
22	no file path - doc title # FAQ	RTK-177-000022	RTK-177-000022
23	no file path - doc title # How To Create Image	RTK-177-000023	RTK-177-000023
24	no file path - doc title # How To Modify Contents In Board	RTK-177-000024	RTK-177-000025

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

Exhibit No	Document Title	Doc ID Trial Exhibit No	End Bates No.
25	no file path - doc title # Design Description	RTK-177-000026	RTK-177-000026
26	no file path - doc title # Realtek Android TV	RTK-177-000027	RTK-177-000027
27	Get the Most Out of the New OpenGL ES 3.1 API by Hans-Kristian Arntzen	CX-0743	
28	ITC-1318 Respondent Initial Post-Hearing Brief	EDIS ID 794065	
29	ITC-1318 Pfister Expert Report	CX-1250C	
30	ITC-1318 Deposition Transcript of Erik de la Iglesia (01/6/2023)	CX-0675C	
31	ITC-1318 Complainant Initial Post-Hearing Brief	EDIS ID 794000	
32	U.S. Patent No. 10,186,069		
33	Certified US Patent No. 7,742,053	JX-0001	
34	ITC-1318 Deposition Transcript of Kuo Jung Wu (11/16/2022)	JX-0158C	
35	ITC-1318 Deposition Transcript of Ming Feng Chen (11/14/2022)	JX-0048C	
36	Realtek Feb. 18, 2023 Ninth Supplemental Objections and Responses to Interrogatory Nos. 1-4	JX-0336C	
37	Robert Benson Annotated Email (02/16/2024)		
38	Christopher Higgins Annotated Email (03/08/2024)		
39	Christopher Higgins Annotated Email (02/23/2024)		
40	Christopher Higgins Annotated Email (02/27/2024)		
41	Michael Chow Annotated Email (04/01/2024)		
42	ARM Mali Dvalin GPU Technical Reference Manual	CX-0773C	
43	GPU 2017 Warp Manager 2017	CX-1730C	
44	GPU Bifrost Architecture Presentation	CX-0483C	
45	Khronos Wiki (Vertex Shader) (https://www.khronos.org/opengl/wiki/Vertex_Shader)		
46	Microsoft Compute Dictionary (5th Edition).		
47	Android Debug Bridge (adb) tool (https://developer.android.com/tools/adb)		
48	“Realtime smoke rendering with OpenGL® ES 2.0” (Available at https://community.arm.com/cfs-file/_key/telligent-evolution-components-attachments/01-2066-00-00-00-00-07-02/Whitepaper_2D00_ParticleSystem.pdf)		
49	https://developer.arm.com/downloads/-/mali-drivers/bifrost-kernel		
50	GPU Compute Unit (2017)	CX-1724C	

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

Exhibit No	Document Title	Doc ID Trial Exhibit No	End Bates No.
51	ARM Mali Bifrost and Valhall Adroid DDK r26p0 Integration Manual	CX-1092C	
52	Robert Benson Annotated Email (03/18/2024)		
53	Realtek Motion to Compel Hearing Tr. (10/21/2022)		
54	U.S. Patent No. 10,891,708		
55	ITC-1318 Deposition Transcript of Zeyin Wei (12/03/2022)	JX-0143C	

I. INTRODUCTION

Complainant Advanced Micro Devices, Inc. and ATI Technologies ULC (collectively, “AMD”) respectfully submits this response to Requestor Realtek Semiconductor, Inc. (“Realtek”) request for Rule 177 ruling. The EOE Branch should deny Realtek’s Ruling Request Letter based on three independent grounds.

First, Realtek’s Ruling Request Letter is nothing more than an attempt to circumvent the ITC’s Limited Exclusion Order without making the necessary design changes to the articles that were actually found to infringe; the standalone Realtek **integrated circuit** products that incorporate the Mali GPUs (“Standalone Realtek ICs”) and printed circuit board assemblies containing the same (“Realtek Development Boards”). Recognizing that it would be too difficult to modify the infringing design of ARM’s Mali GPUs, Realtek resorts to a set of patches to **driver software** (the “Patched Mali Driver”) which it alleges disable the infringing functionality.

Realtek ignores that the Standalone Realtek ICs were found to directly infringe and are excluded from entry into the United States. Yet, there is no driver software installed on, part of, or combined with the Standalone Realtek ICs at the time of their importation or sale. At best, in the far less frequent scenario where Realtek imports or sells Realtek Development Boards containing a Standalone Realtek IC, the driver software **may** (but need not) be contained on a completely separate memory chip that is soldered onto the Realtek Development Board. Even then, the trouble is that the software driver cannot change how the infringing hardware in the Standalone Realtek IC is designed and configured.

Second, even if Realtek’s driver software changes were somehow capable of modifying the configuration of the Standalone Realtek IC containing a Mali GPU found to infringe, and they are not, the physical samples and source code Realtek provided to AMD’s counsel do not work as advertised. As to Claim 19, for instance, the physical samples and source code incorporate a secret

patch that is specifically designed to bypass the Patched Mali Driver software that Realtek described in its Ruling Request Letter and supporting declarations. When Realtek originally attempted to create patches for its software driver to prevent it from submitting certain general-compute tasks (called pilot shader programs) to the shader core for simultaneous execution with certain graphics processing tasks (called fragment shader programs), they ran into critical performance issues with the Mali GPUs. Realtek's solution? Develop a backdoor, secret, "Performance Enhancement Patch" that bypasses the patches that Realtek referenced and described in its Ruling Request Letter and submits the exact same general-compute tasks (called pilot shader programs) to the shader core for simultaneous execution with graphics processing tasks (called fragment shader programs). In other words, Realtek's secret "Performance Enhancement Patch" reversed the allegedly non-infringing benefits of the other driver patches that Realtek disclosed to the EOE Branch.

Realizing how fatal this would be to its request, Realtek made the decision to exclude all reference to this Performance Enhancement Patch from its Letter Request, thereby concealing it from AMD, CBP, and possibly its own experts. Even the author of the release note that accompanied the Performance Enhancement Patch and the architect of the other ARM software patches at issue—ARM's Mr. Guy Larri—decided not to include any reference to the Performance Enhancement Patch in his declaration, despite the fact that the Performance Enhancement Patch Release Note was published on the same day as the release notes for the other patches he included in his declaration. To date, despite AMD's requests, other than the Performance Enhancement Patch Release Note, which Realtek produced only after AMD discovered it on the Realtek source code computer, no documents have been produced which describe the use of the pilot shader programs used by the Performance Enhancement Patch.

Likewise, in the case of Claim 20, each and every one of Realtek's changes to the driver software just introduced a new form of task prioritization. Case in point, even Realtek's own expert, Dr. Hanspeter Pfister, could not support Realtek's claims that the Patched Mali Driver removes any priority from the Mali GPU:

Q. The question was: Do you agree with the statement that there is no longer any priority anywhere in the Mali GPU that can even arguably satisfy Claim 20?

MR. HIGGINS: Object to form.

A. No, I can't say that.

Ex. 14, Pfister Tr. (03/25/2024) at 248:2-7.

Indeed, given the complexity of the mechanisms of scheduling in the ARM GPUs, Realtek cannot remove consideration of priority in scheduling of jobs and their constituent work to be performed by the execution resources by simply changing one of many levels in the GPU's prioritization hierarchy. The effect of Realtek's changes simply shift the prioritization to other levels of the prioritization hierarchy, or create new priority mechanisms, that are considered by the ARM Mali GPUs and the Patched Mali Driver. To illustrate just one of many examples, by disabling Job Slot 2 to prevent prioritization, the Patched Mali Driver now prioritizes tasks in Job Slot 0 over tasks in Job Slot 1 for submission to the shader core. This was confirmed and admitted by ARM's own Mr. Larri:

Q Okay. So in that case, what's in job slot 0 will be prioritized over job slot 1, correct?

MR. VANHOUTAN: Objection, form, misstates his testimony.

A Well, if that -- it will process the one that's in -- for job slot 0 first, in that scenario.

Q So the answer is, yes, it will prioritize what's in job slot 0, correct?

MR. VANHOUTAN: Same objection.

A If the scenario which you described is possible, and if the code behaves as you've represented, then it would prioritize job slot 0 over job slot 1 in that limited scenario.

Q In the patched driver code, correct?

A This -- yes, we're looking at the patched driver code, yes.

Ex. 12, Larri Tr. (03/19/2024) at 231:19-232:12. ARM's other change to allegedly disable priority – setting all jobs to the medium priority level – also creates a new prioritization scheme. Now, instead of picking the next job to submit to the GPU based on the low/medium/high priority value, the Patched Mali Driver picks the next job to send to the GPU based on other prioritization factors including age or relative position in a red/black tree.

Moreover, ARM's changes to the driver to prevent concurrent execution of general-compute and graphics tasks adds yet another new form of task prioritization. To the extent ARM's changes to the driver software actually prevent any compute task from executing simultaneously with a graphics processing task,¹ those same changes necessarily cause the shader core to receive tasks based on a priority, as the Patched Mali Driver would necessarily need to prioritize jobs with a compute label when a job with a compute label is executing and prioritize graphics tasks when other jobs are executing.

Third, even if Realtek's driver software changes could work in theory, and they do not, the EOE Branch should still deny Realtek's Ruling Request Letter because Realtek deliberately obfuscated, delayed, and withheld discovery that is relevant to veracity of many of their otherwise unsubstantiated claims that the driver software works as advertised in an actual system. Realtek's decision to conceal the Performance Enhancement Patch from CBP and AMD was just the tip of the iceberg. Realtek's discovery failures throughout this proceeding are both pervasive and fatal

¹ Realtek has not met its burden to show that the patched driver **ever** prevents a compute task and a graphics processing task from being submitted for simultaneous execution by the shader core. As noted previously, the patched driver at least causes pilot shader program general-compute tasks to execute simultaneously with graphics-processing tasks. In addition, Mr. Larri testified that ARM's new vertex jobs allow other types of general-compute tasks, such as particle physics computations, to be submitted for simultaneous execution with graphics-processing tasks.

to Realtek's request. To illustrate just one of many examples described in more detail in AMD's initial submission, Realtek refused to produce the source code for the applications installed on the Realtek Development Board, one of the infringing articles that Realtek seeks to import. Yet, ARM's own 30(b)(6) witness, Mr. Larri, admitted that the Patched Mali Driver software does not prevent **applications** from prioritizing the submission of shader program tasks to the Mali GPU's shader core (*i.e.*, the execution engines):

Q And the driver doesn't prevent applications from prioritizing the submission of shader programs for processing on the GPU's execution engine, correct?

A Correct. Application programmers are free to write application programs as they wish, as long as they conform with the Khronos-standard APIs when they make requests into the driver.

Ex. 12, Larri Tr. (03/19/2024) at 199:12-19 (emphasis added). Without the source code for these applications, Realtek cannot possibly meet its burden of demonstrating that the Patched Mali Driver software prevents the shader core in the infringing articles from receiving the plurality of tasks based on a priority associated with the tasks in a way that infringes Claim 20. As confirmed by ARM's Mr. Larri:

Q You would need to look at the application source code to figure out if there's any application-based prioritization going on that causes the shader core to receive tasks based on a priority?

A If you're understanding that the scope of the priority question extends beyond the GPU driver and the GPU hardware out into applications themselves, then you would have to understand the operation of the applications themselves.

Id. at 199:2-11. Realtek's refusal to produce the application source code is emblematic of Realtek's numerous other discovery failures throughout this proceeding and are fatal to Realtek's Ruling Request.

In another example, Realtek initially refused to produce the same documentation that Realtek would provide to its customers who purchased a Realtek Development Board. After

repeated requests, Realtek finally copied **some** of this documentation onto the source code computer (perhaps inadvertently) as part of its production of Operating System code. AMD's experts uncovered this documentation during their review of the source code computer, finding that such documentation not only provided instructions for how to use and configure Realtek's Development Boards, but also how to make modifications that would overcome what Realtek purported was its mechanism for preventing users of its products from using a different version of the driver. *See, e.g.*, Exs. 17, 21-26 (RTK-177-000009, RTK-177-000021-27).

These and numerous other critical discovery failures are set forth in detail in AMD's initial submissions in response to Realtek's Ruling Request Letter. *See generally* AMD Submissions dated February 1, 2024, February 2, 2024, and February 7, 2024. Realtek's refusal to produce basic information related to its software patches—and its intentional concealment of other information—are not only fatal to Realtek's ability to carry its burden in this proceeding, but have prevented AMD from responding to many of Realtek's positions in its Ruling Request Letter.

Realtek's tactic of litigation by obfuscation is, unfortunately, a constant theme in this litigation. In the underlying 1318 ITC investigation, in response to a motion to compel Realtek's complete source code, Realtek submitted an opposition including sworn declarations by its Director in Multimedia Business Unit II, Hsieh, Chien-Hua, and its Manager in Connected Media Business Unit II, Chen, Ching-Hu, claiming that collecting and producing its complete source code would take 5,360 work hours (*i.e.*, 67 business days with 10 engineers working full time on collection) to collect the requested source code. *See* Realtek's Opposition to AMD's Motion to Compel Realtek to Supplement Source Code Production, EDIS Doc. ID 782466 (Oct. 14, 2022) at Ex. Nos. 1, 2. During a case management conference to address the issue, the presiding ALJ expressed concern that Realtek's counsel was being “bamboozl[ed]” by its client, and ordered

Realtek to produce the withheld source code “as soon as they can.” *See* Ex. 53, Motion to Compel Hrg. Tr. (10/21/2022) at 28:8-14, 29:5-7, 9-15; *see also id.* at 28:8-9 (“I am going to discount these declarations from the managers of Realtek”). Despite Realtek’s prior sworn declaration, that it would take at least three months to collect the code with 10 engineers working full time on collection, it had no trouble producing the code within mere days. More recently, in a proceeding before the ITC, Realtek and its counsel of record in this proceeding were formally admonished and penalized with monetary sanctions for providing discovery responses including “material misrepresentations made ‘with a culpable state of mind and with the intent to hide evidence.’” *See* Order No. 39, Inv. No. 337-TA-1350, EDIS Doc. ID 807211 (Oct. 30, 2023).

While unfortunate that Realtek has chosen to run the same old tricks and tactics in this proceeding for the purpose of making it harder on AMD’s counsel and experts to develop and present a complete record for CBP’s consideration, Realtek misses that the price tag for its gamesmanship is higher this time around. Here, unlike in prior proceedings in which Realtek was a respondent, Realtek bears the burden of production and proof. As Realtek has failed to meet this burden by failing to produce documents and information that are relevant to resolving the disputes between the parties, the EOE Branch should deny Realtek’s request.

In sum, Realtek has not met, and cannot meet, its burden to show that the software driver-based patches should permit it and its customers to continue to import infringing products in violation of the Commission’s LEO. Moreover, what Realtek has demonstrated in this proceeding is its intent on seeking a favorable outcome based on inclusion of a set of software patches, only to apply other software patches that would reintroduce infringing uses of the Mali GPUs. This is not a hypothetical scenario, but is precisely what Realtek attempted with the “Performance Enhancement Patch” that was uncovered during AMD’s review of the code, and it only serves to

foreshadow what other software patches Realtek has in store to reintroduce additional infringing functionality to the products at issue should it be granted its request to circumvent the LEO.

II. PROCEDURAL BACKGROUND

On June 7, 2022, the Commission instituted an investigation (337-TA-1318) (“ITC-1318 Investigation”) based on a complaint filed by Advanced Micro Devices, Inc. and ATI Technologies ULC (collectively “AMD”) against TCL Industries Holdings Co., Ltd.; TCL Industries Holdings (H.K.) Limited; TCL Electronics Holdings f/k/a TCL Multimedia Technology Holdings, Ltd.; TCL Technology Group Corporation; TTE Corporation; TCL Holdings (BVI) Limited; TCL King Electrical Appliances (Huizhou) Co., Ltd.; Shenzhen TCL New Technologies Co., Ltd.; TCL MOKA International Limited; TCL Smart Device (Vietnam) Co., Ltd.; Manufacturas Avanzadas SA de CV; TCL Electronics Mexico, S de RL de CV; and TCL Overseas Marketing Ltd. (collectively, “TCL”) and Realtek Semiconductor Corp. (“Realtek”) based on the sale and importation of certain graphics systems and televisions.

ALJ Elliot held an evidentiary hearing from March 13-17, 2023. *See* Ex. 11, Hrg. Tr. 1-1417. Following post-hearing briefing by the Parties, on July 7, 2023, the ALJ issued the Initial Determination (“ID”). The ID found that Realtek and TCL violated section 337 as to asserted Claims 19 and 20 of the ’381 Patent. Ex. 8, ID at 180. The Parties submitted On January 24, 2024, based on the ID’s findings and recommendations, the Commission issued a Limited Exclusion Order (“LEO”) against Realtek and TCL’s infringing products and a Cease and Desist Order (“CDO”) against TCL.² The LEO bars importation by TCL and Realtek of ““(a) integrated circuits that incorporate one or more graphics processing units (GPUs); (b) printed circuit board assemblies

² The ID recommended that the CDO apply to all Respondents, with the exception of Realtek and TTE. Ex. 8, ID at 184, 186. The Commission subsequently held that the CDO should apply to TTE. Ex. 9, FD at 68.

containing the same; and (c) digital televisions containing the same’ that infringe one or more of Claims 19 and 20 of the ’381 Patent.”

On January 26th, 2024, Realtek filed its Ruling Request Letter (“RRL”) pursuant to 19 C.F.R. 177 seeking a ruling that certain of its graphics systems, components thereof, and digital televisions containing the same do not infringe Claims 19 and 20 of the ’381 Patent. Specifically, Realtek seeks a ruling that through alleged modifications to certain aspects of certain versions of a software driver provided by ARM, the products that are subject to the LEO would no longer be infringing.

III. CATEGORIES OF PRODUCTS SUBJECT TO THE LEO

The Realtek Products that are subject to the LEO are: (a) integrated circuits that incorporate one or more GPUs; (b) printed circuit board assemblies (such as development boards) containing the same, and (c) digital televisions containing the same. These products include at least Realtek integrated circuits and development boards, and the graphics systems incorporated in the foregoing. Ex. 7, LEO at 2-3. Given that TCL has taken a license, AMD and TCL have moved to modify the LEO to now remove TCL digital televisions from the LEO. *See* Section II, *supra*.

1. Standalone Realtek Integrated Circuits (ICs)

The Commission determined that certain (standalone) Realtek integrated circuits directly infringed Claims 19 and 20 of the ’381 Patent (“Standalone Realtek ICs”) and are subject to the

Commission’s exclusion order. All of the Standalone Realtek ICs determined to infringe the asserted claims contained an ARM Mali G31, G51, G57, or G310 graphics processing unit (“GPU”). Ex. 8, ID at 10-13. Realtek does not design its own GPUs. Instead, Realtek obtains pre-designed Bifrost and Valhall GPU designs, defined in the form of register-transfer level (RTL) source code, from ARM. *See* Ex. 8, ID at 9-10, 14. Realtek integrates the RTL source code that they receive from ARM into the physical implementation of the Standalone Realtek ICs excluded by the LEO. Ex. 3, RRL at 13; Ex. 11, Hrg. Tr. (Reinman) at 92:13-94:9. Realtek then sells, ships, and distributes these Standalone Realtek ICs into the United States.

As sold or imported by Realtek, none of these Standalone Realtek ICs contained a scheduling module, or driver software of any kind. *See* Ex. 8, ID at 130. Instead, the circuitry of the GPUs incorporated in the Standalone Realtek ICs was, and continues to be, configured in **integrated circuit hardware** to practice all claim limitations. *Id.* at 100-21, 123-25. For this same reason, the Standalone Realtek ICs did not contain—and cannot possibly contain—a scheduling module or driver software of any kind when Realtek imports them into the United States or sells them to customers. Thus, because Realtek does not import the Standalone Realtek ICs with driver software, any proposed change to such driver software will have absolutely no effect on the infringing, hardware circuitry configuration the Standalone Realtek ICs at the time of their importation or sale.

2. Realtek Development Boards

The Commission determined that certain Realtek printed circuit board assemblies directly infringed Claims 19 and 20 of the ’381 Patent (“Realtek Development Boards”) and are subject to the Commission’s LEO. Ex. 9, FD at 64. In addition to containing one of the Standalone Realtek ICs, these Realtek Development Boards may also contain a software platform comprising an operating system including firmware and an ARM Mali graphics driver, as well as a number of

applications (collectively, the “Realtek Dev Board Platform Software”). At the time of their importation, the Realtek Development Boards included one of the following operating systems: Linux 4.4.3, Android v. 9.0.0, or Android v. 11.0.0. Ex. 36, JX-0336C (Realtek Feb. 18, 2023 Ninth Supplemental Objections and Responses to Interrogatory Nos. 1-4) at 39-43. These operating systems, in turn, included one of a number of ARM Mali driver code versions. *Id.* at 21-36.

Realtek Development Boards play a key role in Realtek’s RRL. Unlike Standalone Realtek ICs that do not contain, and are not combined with any driver software at the time of the sale or importation into the United States, Realtek Development Boards can (but need not) contain driver software. Thus, the Realtek Development Boards are the only category of products to which Realtek’s claims about changes to the driver software could potentially apply.

The purpose of the Realtek Development Board is to provide Realtek’s customers with a platform for evaluating the performance Standalone Realtek ICs with the customer’s specific end application, using the customer’s software platform. The process for a customer to integrate a Standalone Realtek IC into a customer-designed printed circuit board assembly is time consuming and expensive. Before Realtek customers undertake this significant effort and investment, Realtek’s customers need to evaluate the Standalone Realtek IC to make sure that it will meet the customer’s performance requirements using the customer’s software platform. To facilitate this, and to encourage customers to try out Realtek’s Standalone IC, Realtek supplies these Realtek Development Boards. While the Realtek Development Boards can often come pre-loaded with Realtek supplied software, the Realtek Development Boards are designed to allow its customers to develop their own proprietary software platform and customize and modify the Realtek-supplied software as they see fit.

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

Like with the RTL source code that defines the GPU hardware, Realtek does not develop its own source code for driver software. Ex. 8, ID at 9 10,14. Instead, Realtek sources driver source code from third parties, such as ARM. In the case of ARM’s driver, it is provided in the form of a Driver Development Kit (DDK) for the ARM Mali GPUs. Ex. 12, Larri Tr. (03/19/2024) at 136:17-22; Ex. 15, Larri Tr. (11/01/22) at 135:10-15. Realtek has represented, through its Ruling Request Letter, that it has also received certain patches to the DDK driver, and that it has incorporated the patches without further modification in what it refers to as the “New Mali Driver” into the Realtek Development Boards. Ex. 3, RRL at 1, 6-7 (“The two representative patches... are referred to herein as the “New Mali Driver.”) (“[Realtek] submits the two ARM “patches” that implement the new design as exhibits to the Declaration of Mr. Guy Larri.”) (emphasis added).

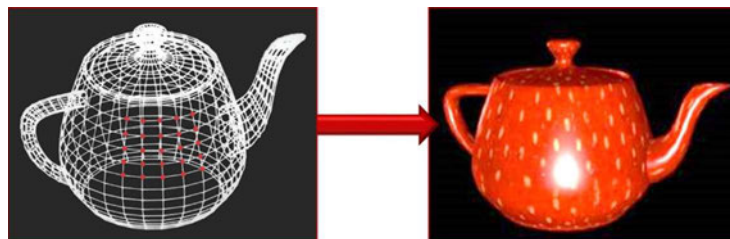
Realtek generally provides its customers with access to the DDKs. However, Realtek’s customers are free to obtain driver software from a variety of other sources, including through download from ARM’s secure portal based on the customer’s own direct or indirect contractual relationship with ARM (*see e.g.*, <https://developer.arm.com/downloads>), public download from ARM’s website (*see e.g.*, <https://developer.arm.com/downloads/-/mali-drivers/bifrost-kernel>), or other third party software developers that author and release open source drivers for the ARM Mali GPU (<https://docs.mesa3d.org/drivers/panfrost.html>). According to ARM, it does not police its driver source code, how it is used, or what modifications are made. *See* Ex. 12, Larri Tr. (03/19/2024) at 143:20-144:17. Realtek also does not police or exercise any control over how its customers integrate drivers into their end products. Ex. 34, Realtek Corporate Representative Juo Jung Wu Tr. (11/16/2022) at 47:12-21; Ex. 35, Realtek Corporate Representative Ming Feng Chen Tr. (11/14/2022) at 135:19-136:5. Accordingly, assuming Realtek end customers use ARM’s

driver source code or driver software source code from some other third party, they are free to make changes to the source code as they see fit.

IV. TECHNOLOGY OVERVIEW

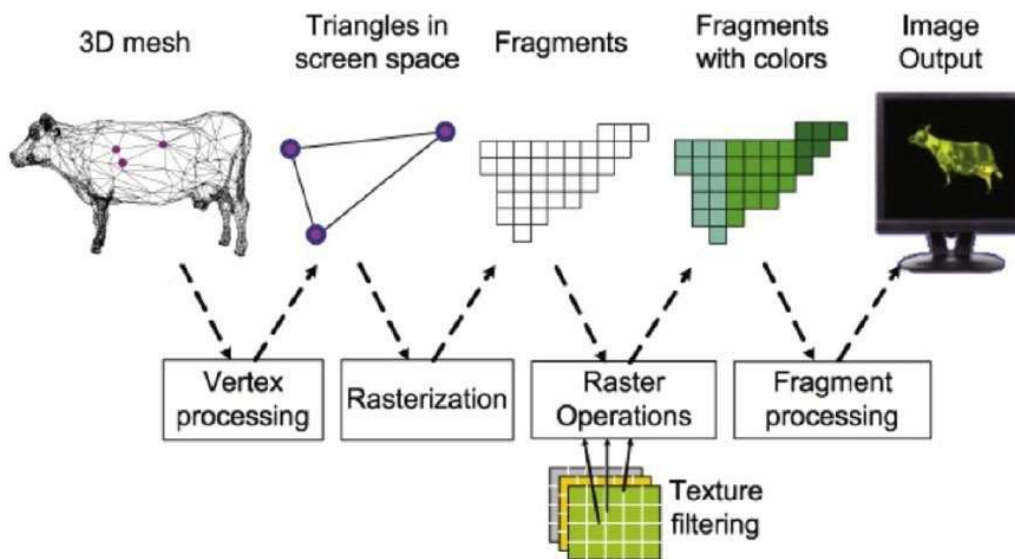
The '381 Patent is directed to novel architectures for optimizing performance and utilization of graphics processing unit (GPU) circuitry. In modern graphics processors, such as those found in TVs, GPUs are utilized to render high-definition graphics. Unlike the Central Processing Unit (CPU) that typically handles only general-purpose computing tasks, GPUs are efficient at handling multiple tasks simultaneously, thanks to their parallel processing architecture. This capability allows them to process large blocks of visual data, such as textures and shaders, more efficiently than a CPU. In the context of televisions, this means that a GPU can quickly generate lifelike images, support complex visual effects, and manage fast-moving scenes. A GPU's capability to process tasks in parallel also allows the CPU to offload other types of tasks, such as general-compute tasks, to be performed by the GPU.

For graphics processing, GPUs are used to manipulate and render 3D graphical objects onto a 2D display screen. GPUs consist of circuitry that is specially designed to convert 3D objects (typically represented by a collection of geometrical shapes such as triangles) into a 2D image (typically comprising a collection of pixels) as shown below. The 2D image represents a depiction of the 3D objects, rendered from the perspective of a given viewer, and shaded to convey the object's surface material illuminated from a given light source. Ex. 1, Reinman Decl. ¶¶ 30-31.



In the above example image, on the left, is a wireframe model of a teapot. This is a visual representation of a 3D object, that effectively serves as the skeleton of the model, composed of vertices, edges, and faces that define the shape of the 3D object in space. On the right, is the fully rendered teapot, including the pixel colors and textures applied. This is the result of the GPU's rendering pipeline, where the wireframe model has been rasterized, shaded, and textured to produce the final image. Ex. 11, Hrg. Tr. (Reinman) at 105:3-107:5.

Texture mapping is the process of wrapping a 2D image texture onto the surface of a 3D object to give it color, detail, and surface texture. *Id.* The rendering process also includes lighting calculations to create highlights and shadows, which give the object a more realistic appearance and depth. *Id.* This is often one of the initial steps in 3D rendering, where the geometric structure is established by generating the various vertices and edges to form the skeleton for later processing stages. *Id.* The conversion of a 3D shape into a 2D image happens in a pipeline of stages. *Id.* This pipeline of processing steps is explained, in connection with the diagram below.

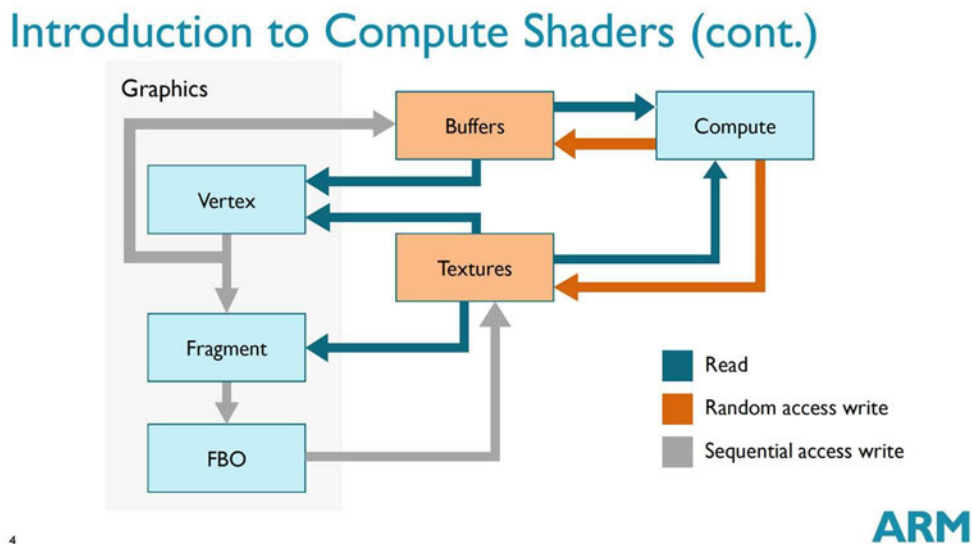


The first stage is often referred to as “vertex processing” and is a stage that computes, for each vertex and its 3D position in space, a new 2D position (optionally with depth, distance from

the camera) of that vertex in the space of the screen or output image. The process begins with a 3D mesh, which is essentially a collection of vertices (corners of 3D shapes) connected by edges and faces, resembling a wireframe model. This mesh represents the geometric structure of a 3D object, in this case, a cow. The GPU performs these vertex processing operations, where it calculates the position of each vertex within the 3D space and transforms it into the corresponding position in screen space. This involves various transformations, such as scaling, rotation, and translation, to place the object in the correct position and perspective relative to the viewer. The next “rasterization” stage receives vertices from the vertex processing stage and generates pixels that correspond to each primitive. This step is akin to filling in outlined shapes—such that triangles in the 3D model of the cow are mapped to the 2D screen. Rasterization also includes applying detailed patterns or images on the surface of the 3D model to give it color and texture. Finally, pixel processing takes the textured fragments and applies lighting, shading, and color calculations, which provides the depth perception of the 3D model as rendered to the 2D screen. This step determines the final color of each pixel, considering how light interacts with the surfaces of the 3D model. This can include simulating the effects of shadows, reflections, and transparency. The generated pixels are also called “fragments” instead of pixels because some (*i.e.*, at the edges) cover only partial screen pixels. *Id.*; Ex. 1, Reinman Decl. ¶¶ 33-37.

To perform these complex graphics processing tasks, a GPU leverages shader programs that execute on specialized hardware. Ex. 1, Reinman Decl. at Section VI.B, VI.C. These shader programs, also called shaders, are specialized programs that execute certain processes for a given graphics processing operation. *Id.* For example, a GPU typically includes a vertex shader, a geometry shader, and a fragment or pixel shader for performing graphics processing. *Id.* at Section VI.C. In addition, modern GPUs also include compute shaders for handling additional general-

purpose computing on the GPU. *Id.* at Section VI.D. Compute shaders are designed to handle a wide range of computational tasks leveraging the parallel processing capabilities of the GPU. *Id.* These computing tasks include physics simulations and complex mathematical algorithms, and can also be used to support computations that are part of graphics processing. For example, in the configuration below as described by ARM, a compute shader is used in conjunction with a vertex shader and a fragment shader. Compute shaders can be used to support computations for wave simulation, occlusion culling, physics, particle effects, and image processing. *Id.*



See Ex. 27, CX-0743 (Get the Most Out of the New OpenGL ES 3.1 API) at AMD1318_0180101, 0121. Ex. 1, Reinman Decl. ¶¶ 45-49.

V. PATENT BACKGROUND

United States Patent No. 8,854,381 (“the ’381 Patent”) is entitled “Processing Unit That Enables Asynchronous Task Dispatch,” and issued on October 7, 2014 to inventors Michael Mantor and Rex McCrary. The ’381 Patent issued from United States Patent Application No. 12/874,134, which was filed on September 1, 2010. The ’381 Patent claims priority from United States Provisional Patent Application No. 61/239,712, filed on September 3, 2009.

The '381 Patent is directed to specialized GPU hardware for facilitating the efficient parallel execution of graphics processing tasks received, for example, from a host system. As of the effective date of the '381 Patent, it was common for a host CPU to pass tasks to the GPU. Ex. 2, '381 Patent at 1:66-2:7 (“a GPU can perform the complex mathematical computations required for a graphics-processing task more quickly than a typical central-processing unit (CPU). As a result, if a computing system includes a GPU, graphics-processing tasks (and other types of data-parallel processing tasks) are typically passed to the GPU, rather than the CPU.”).

The graphics tasks for a GPU originate from end user applications that communicate with the GPU through several layers of software. As explained in the '381 Patent:

There are several layers of software between the end-user application and the GPU. The end-user application communicates with an application-programming interface (API)...The API communicates with a driver. The driver translates standard code received from the API into a native format of instructions understood by the GPU. The driver is typically written by the manufacturer of the GPU. The GPU then executes the instructions from the driver.

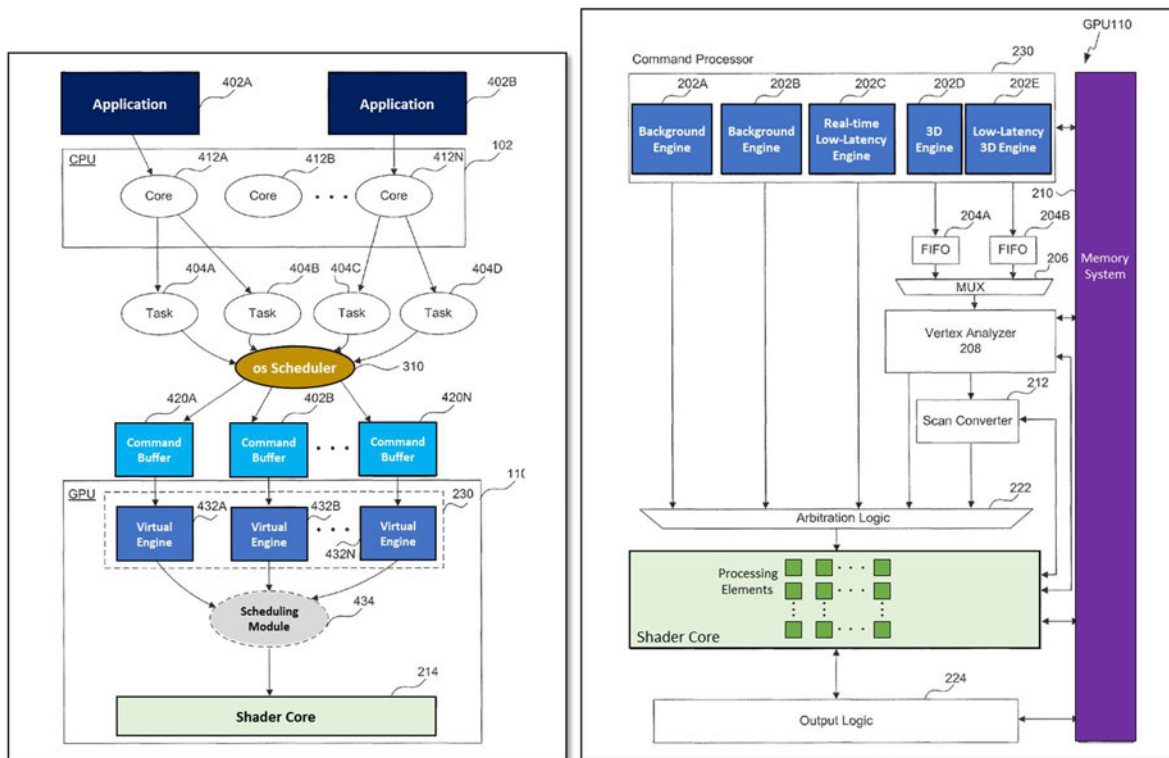
Ex. 2, '381 Patent at 1:33-46. In the conventional approach, these tasks were passed to the GPU through a single command buffer in a serialized manner. *Id.* at 10:22-25 (“In a conventional system, an OS scheduler would provide tasks 404 to a single command buffer in a serial fashion, and a conventional GPU would serially process the tasks..”); *see also id.* at 2:8-16. But this single command buffer approach did not provide enough work to fully utilize the GPU’s shader core, and resulted in significant latency in the system when switching between tasks that are to be executed in a different order than in the single command buffer. The '381 Patent addressed this through the novel architecture configuration of the GPU, which supported asynchronous task dispatch to the GPU through a plurality of engines that could service command buffers in parallel:

Unlike such a conventional system, OS scheduler 310 provides each task 404 to

one of a plurality of command buffers 420A-N based on the priority type specified by the applications 402. For example, OS scheduler 310 provides a first type of task (e.g., high-priority tasks) to first command buffer 420A, a second type of task (e.g., graphics-processing tasks) to second command buffer 420B, and so on.

GPU 110 includes a plurality of virtual engines 432A-N, each configured to service one of command buffers 420A-N. For example, a first virtual engine 432A is configured to service first command buffer 420A; a second virtual engine 432B is configured to service second command buffer 420B; and an N-th virtual engine 432N is configured to service N-th command buffer 420N. Tasks from virtual engines 432 are then (substantially) simultaneously executed by shader core 214 as described above.

Id. at 10:25-41 (emphases added). This architecture is illustrated and described in the '381 Patent at Figures 3 and 4, which are reproduced in annotated form below.



As shown in the annotated figures, the GPU 110 is coupled to a **memory system** and is configured to support tasks that are generated by a set of user **applications** 402A and 402B. These **applications** 402A and 402B are executed by a CPU's processing cores 412A-C to generate the plurality of tasks 404A-D. These tasks are scheduled by an **Operating System (os) Scheduler** 310 which dispatches the tasks to the GPU 110 through a plurality **command buffers** 420A-420N.

Each of these command buffers is serviced by one of a plurality of **engines** (432A-432N, Fig. 4 and 202A-202E, Fig. 3) of the GPU 110, such that the **engines** dispatch the tasks for execution through an internal GPU scheduling module (434, Fig. 4) or Arbitration Logic (222, Fig. 3) to the **shader core** 214.

The GPU architecture and its corresponding configuration to support the simultaneous execution of tasks are the subject matter of independent Claim 15, and its dependent Claims 19 and 20, that are at issue in this proceeding. Specifically, Claim 15 recites:

15. A computing system comprising:

a memory;

a first processing unit; and

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

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

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

In the context of the '381 Patent specification and figures as reproduced and annotated above, the claimed first processing unit corresponds to the GPU 110, which includes the **plurality of engines** and the **shader core**. As will be discussed in greater detail below, the referenced scheduling module that is associated with the second processing unit corresponds to the **OS Scheduler 310** which is associated with a second and separate processing unit (*e.g.*, the CPU 102 of Fig. 4).

Claim 19 depends directly from Claim 15 and recites:

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

As described and disclosed in the '381 Patent, a general-compute task includes “mathematical algorithms, physics simulations, or the like.” Ex. 2, '381 Patent at 9:24-26. As the '381 Patent also explains, a graphics processing task typically includes the execution of “complex mathematical computations, such as matrix and vector operations” and can make use of “a plurality of different shader programs (*e.g.*, geometry shader, vertex shader, pixel shader, tessellation shader, or the like). Ex. 2, '381 Patent at 1:47-79; 8:34-38. Ultimately, given its plain and ordinary meaning, a general-compute task is a task that performs general computations. *See* Ex. 1, Reinman Decl. at Section VII.B.3; Ex. 12, Larri Tr. (03/19/2024) at 73:3-7 (“I would say that general purpose compute is compute that can be used for a wide range of different applications.”)

Claim 20 depends directly from Claim 15 and recites:

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

Claim 15 sets forth a plurality of tasks which include the first task and the second task that a shader core is configured to execute. Claim 20, refers back to “the plurality of tasks.” In contrast, each of claims 16-19 refer, by way of antecedent basis, to the first task and the second task of Claim 15. Therefore, for the purposes of Claim 20, each of the plurality of tasks must be received based on their associated priority, but the shader core must be configured to execute only two of the plurality of tasks simultaneously to satisfy the limitation of Claim 15. That is, for example, if there are three tasks, Task 1, Task 2, and Task 3, Claim 20 is satisfied if each of Tasks 1-3 are received by the shader core based on their respective priority, and any two of Tasks 1-3 are executed at the same time by the shader core. Thus, one of the three tasks may be executed alone in the shader core based on its priority while still satisfying the requirements of Claim 15 and 20.

The embodiments described in the '381 Patent describe, for example, the reception of tasks by the shader core based on priorities for a task type, priorities associated with the engines that receive the tasks, or based on the processing of priorities by other components of the GPU. Specifically, the '381 Patent discloses, in one example embodiment illustrated and described with reference to Figure 2, that:

Real-time low-latency engine 202C has priority access to the resources of GPU 110 in order to process high-priority task. Primary 3D engine 202D processes standard graphics-processing tasks, and low-latency 3D engine 202E processes high-priority graphics-processing tasks. Low-latency 3D engine 202E has priority access to the graphics-processing resources of GPU 110.

Ex. 2, '381 Patent at 7:14-19. As yet another example of prioritization, the '381 Patent describes prioritization of tasks by enabling them to “(substantially) fully consume the resources of the GPU” temporarily:

For example, resources (*e.g.*, SIMDs) of the GPU shader core can be divided between concurrent tasks **based on demand, priority, and/or preset limits-while temporarily enabling any one task to (substantially) fully consume the resources of the GPU.**

Id. at 5:34-39 (emphasis added). Providing still another example, the '381 Patent states:

For example, scheduling module 434 may divide the resources of shader core 214 between concurrent tasks 404 **based on demand, priority, and/or preset limits-while temporarily enabling any one of tasks 404 to fully consume the resources of GPU 110.**

Id. 10:47-51. The '381 Patent further discloses, in some embodiments, that the priorities of the tasks are considered, for example, by a scheduling module 434 of the GPU that is not part of the shader core 214:

Tasks from virtual engines 432 are then (substantially) simultaneously executed by shader core 214 as described above. In an embodiment, **a scheduling module 434 of GPU 110** schedules the tasks to be executed by shader core 214 based on at least the following conditions: (i) the priority type specified by applications 402; (ii) **the relative priority between the tasks 404 processed by virtual engines 432**; and (iii) the availability of resources within shader core 214.

Id. at 10:41-51 (emphasis added).

VI. LEGAL PRINCIPLES

1. Customs' Scope of Review/Enforcement

Pursuant to 19 U.S.C. § 1337(d)(1), U.S. Customs and Border Protection (“CBP”) is tasked with excluding infringing relevant articles that are subject to an exclusion order that resulted from an ITC investigation. *E.g.*, Ruling Request; U.S. International Trade Commission; General Exclusion Order; Investigation No. 337-TA-739; *Certain Ground Fault Circuit Interrupters and Products Containing Same*, 2023 U.S. CUSTOM HQ LEXIS 45, *15 (Feb. 13, 2023). Such an exclusion order extends to all related models and products to those adjudicated at the ITC, and is not limited by specific models. *E.g.*, Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1265; *Certain Fitness Devices, Streaming Components Thereof, and Systems Containing Same*, 2023 U.S. CUSTOM HQ LEXIS 120, *20-21 (Sept. 6, 2023) (quoting *Certain Optical Disk Controller Chips and Chipsets*, Inv. No. 337-TA-506, Comm'n Op. at 56-57, USITC Pub. 3935, Doc ID 287263 (July 2007)). An importer may request a ruling from CBP that certain related products are no longer subject to the exclusion order because of a redesign that allegedly avoids infringement. 19 C.F.R. Part 177. But, CBP may not issue a noninfringement ruling where 1) the redesign is hypothetical, or 2) the issue is pending before the U.S. Court of International Trade, the U.S. Court of Appeals for the Federal Circuit, or any appeals court therefrom. 19 C.F.R. Part 177.7.

Pursuant to 19 C.F.R. Part 177, an administrative ruling is limited to the scope of infringement as determined in the underlying ITC investigation. Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1200; *Certain Electronic Devices, Including Streaming Players, Televisions, Set Top Boxes, Remotes Controllers, and Components Thereof*, 2022 U.S. CUSTOM HQ LEXIS 15, *25 (Feb. 8, 2022)

(“[T]he scope of the remedy is dependent upon the scope of the investigation, which is determined by the notice of investigation.”) (citing *In re Certain Automated Mech. Transmission Sys.*, 2007 ITC LEXIS 1300, *5-6 (Int’l Trade Comm’n Aug. 1, 2007)); see also 19 C.F.R. § 210.10 (“The notice will define the scope of the investigation.”).

It is within CBP’s authority to evaluate what new questions of infringement have been raised through an importer’s redesign. Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1191; *Certain Audio Players and Controllers, Components Thereof, And Products Containing the Same*, 2022 U.S. CUSTOM HQ LEXIS 45, *32, *35 (June 3, 2022) (stating that a CBP *inter partes* proceeding under 19 C.F.R. Part 177 is alternative to the Commissions modification and advisory opinion proceedings).

Similarly, the relevant articles in a Part 177 administrative ruling proceeding are “any articles—new, modified, or otherwise— that are ‘covered by’ the patent claims included in the exclusion order.” Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1012, *Certain Magnetic Data Storage Tapes and Cartridges Containing the Same*, 2018 U.S. CUSTOM HQ LEXIS 354, *46-47 (Aug. 9, 2018) (“The Commission’s long-standing practice is to direct its remedial orders to all products covered by the patent claims as to which a violation has been found, rather than limiting its orders only to those specific models selected for the infringement analysis ... the Commission’s jurisdiction extends to all models of infringing products that are imported at the time of the Commission’s determination and to all such products that will be imported during the life of the remedial orders.”) (citing *Certain Optical Disk Controller Chips and Chipsets*, Inv. No. 337-TA-506, Comm’n Op. at 56–57, USITC Pub. 3935, Doc ID 287263 (July 2007)).

All determinations of the Commission in the underlying ITC investigation, including claim construction, are “binding authority on CBP and, in the case of conflict, will by operation of law modify or revoke any contrary CBP ruling or decision regarding application of an exclusion order[.]” *E.g.*, Ruling Request; U.S. International Trade Commission; General Exclusion Order; Investigation No. 337-TA-739; *Certain Ground Fault Circuit Interrupters and Products Containing Same*, 2023 U.S. CUSTOM HQ LEXIS 45, *1-2 (Feb. 13, 2023); Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1265, *Certain Fitness Devices, Streaming Components Thereof, and Systems Containing the Same*, 2023 U.S. CUSTOM HQ LEXIS 120, *1-2 (Sept. 16, 2023).

2. Realtek’s Burden of Proof

CBP will continue to exclude all relevant articles under the exclusion order “until [the importer] either: (1) disables or renders inoperable the [infringing feature] addressed in [this Part 177] ruling such that it can no longer be used with the relevant articles at issue and provides notice as to that disabling or inoperability; or (2) receives a non-infringement determination pursuant to an ancillary proceeding at the Commission or a ruling in an *inter partes* proceeding from CBP under 19 C.F.R. Part 177 that addresses operation of the [infringing feature] on the articles at issue.” Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1191; *Certain Audio Players and Controllers, Components Thereof, And Products Containing the Same*, 2022 U.S. CUSTOM HQ LEXIS 45, *3 (June 3, 2022). Where, as here, a patent owner has already satisfied its burden of proof infringement at the ITC, at Customs “the burden remains squarely on the importer ‘as a condition of entry’ to establish admissibility for purposes of Section 337.” *Id.* at *33 (citing *Hyundai Elecs. Indus. Co. v. ITC*, 899 F.2d 1204, 1210 (Fed. Cir. 1990.)) (“[P]lacing the risk of unfairness associated with a prophylactic order upon potential importers rather than American manufacturers that...have demonstrated their entitlement

to protection from unfair trade practices—applies here [in the context of a limited exclusion order] with increased force.”); Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1265; *Certain Fitness Devices, Streaming Components Thereof, and Systems Containing Same*, 2023 U.S. Custom HQ LEXIS 120, *21 -22 (Sept. 16, 2023) (quoting and citing the same); Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1200; *Certain Electronic Devices, Including Streaming Players, Televisions, Set Top Boxes, Remotes Controllers, and Components Thereof*, 2022 U.S. CUSTOM HQ LEXIS 15, *18 (Feb. 8, 2022) (quoting and citing the same).

In the context of an exclusion order granted on the basis of patent infringement, it is on the importer to prove noninfringement in a Part 177 *inter partes* proceeding. To sufficiently prove noninfringement, the importer must have sufficient control over the redesign such that CBP has reasonable certainty that the subject products do not infringe. *See e.g.*, Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1265; *Certain Fitness Devices, Streaming Components Thereof, and Systems Containing Same*, 2023 U.S. Custom HQ LEXIS 120, *30-31 (Sept. 16, 2023) (“Without modification, that **only iFit is able to perform**, these products cannot operate in an infringing mode and, as such, are not subject to the exclusion order.”) (emphasis added); Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1012, *Certain Magnetic Data Storage Tapes and Cartridges Containing the Same*, 2018 U.S. CUSTOM HQ LEXIS 354, *56-57 (Aug. 9, 2018) (finding that the evidence provided by the importer was “not sufficient to allow [CBP] to **determine with certainty**, after the fact, whether the application of the [redesign] would have eliminated all potentially infringing articles.”) (emphasis added).

Additionally, even where the importer has shown that a separate previous redesign does not infringe, the importer must still show that any subsequent redesign products also do not infringe and cannot merely rely on a past noninfringement finding. *See* Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1191; *Certain Audio Players and Controllers, Components Thereof, And Products Containing the Same*, 2022 U.S. CUSTOM HQ LEXIS 45, *32, *49-50 (June 3, 2022) (“If reliance on a certification was dispositive, it would allow a respondent to self-judge that it was complying with the Commission’s remedial orders. CBP has previously refused to permit such a self-certification.”) (citing *Wirtgen Am., Inc. v. United States*, 443 F. Supp. 3d 198 (D.D.C. 2020)) (“[A]dmisibility turns on whether Google has adhered to the Commission’s non-infringement findings by continuing to implement the adjudicated redesigns without, **for example, returning to the legacy functionality found to infringe or developing new functionalities that, as unadjudicated features, raise an open question as to infringement** and, consequently, an open question whether the products incorporating the new functionalities are entitled to entry for consumption into the United States.”) (emphasis added) (finding that products that were subsequently modified after the ITC’s partial noninfringement finding were subject to exclusion until the importer proved that the subsequent modifications did not put the products back within the scope of the exclusion order) .

3. Infringement of Apparatus Claims

Generally, determining infringement is a two-part analysis. *E.g.*, Ruling Request; U.S. International Trade Commission; General Exclusion Order; Investigation No. 337-TA-739; *Certain Ground Fault Circuit Interrupters and Products Containing Same*, 2023 U.S. CUSTOM HQ LEXIS 45, *19 (Feb. 13, 2023) (citing *Advanced Steel Recovery, LLC v. X-Body Equip., Inc.*, 808 F.3d 1313, 1316 (Fed. Cir. 2015)). First, claims are construed as a matter of law to define the claim limitations as understood by a person of ordinary skill in the art (“POSITA”). *E.g.*, *id.* at

*19-20 (citing *SIMO Holdings, Inc. v. H.K. uCloudlink Network Tech., Ltd.*, 983 F.3d 1367, 1374 (Fed. Cir. 2021) and *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005)). Second, the accused product is compared to the claim limitations as a matter of fact. *E.g., id.* at *21 (citing *Apple Inc. v. Samsung Elecs. Co., Ltd.*, 839 F.3d 1034 (Fed. Cir. 2016), cert. denied). Where a product practices all claim limitations, either literally or via a functional equivalent as understood by a POSITA, that product infringes and must be excluded under the exclusion order. *E.g., id.* at *19 (citing *Microsoft Corp. v. GeoTag, Inc.*, 817 F.3d 1305 (Fed. Cir. 2016), cert. denied).

The Federal Circuit has clarified that apparatus claims that have functional claim limitations with “configured to” language require that the capability to perform the functional limitation be intended or designed-for. *Silicon Graphics, Inc. v. ATI Techs., Inc.*, 607 F.3d 784, 794 (Fed. Cir. 2010) (explaining that where a product “includes the structural means for performing a claimed function, [an accused product] can still infringe separate and apart from the operating system that is needed to use that product”); *INVT SPE LLC v. Int’l Trade Comm’n*, 46 F.4th 1361, 1375-76 (Fed. Cir. 2022) (distinguishing cases where capability was possible and where the capability was expected such that infringement was ultimately shown). To determine if an accused device is designed to be capable of operating in the infringing mode, the infringement analysis only applies to the claim language that specifies the capability to perform that function. *See Finjan, Inc. v. Secure Computing Corp.*, 626 F.3d 1197, 1204-05 (Fed. Cir. 2010) (internal quotations omitted) (finding that the circuitry for an engine that was configured to access an operating system satisfied the “infringing mode” test even without the operating system present); *Fantasy Sports Props. v. Sportsline.com, Inc.*, 287 F.3d 1108, 1118 (Fed. Cir. 2002).

Further, where an apparatus claim that includes software components with specific purposes, all that is required is that the software contain code or circuitry written to “cause” the

specified purpose, even if that code or circuitry is not “active or enabled” in the accused product. *See Silicon Graphics*, 607 F.3d at 794; *Finjan*, 626 F.3d at 1204 (“Thus, it is undisputed that software for performing the claimed functions existed in the products when sold—in the same way that an automobile engine for propulsion exists in a car even when the car is turned off.”). Additionally, where a product is intended to be modified and designed to be modified by a downstream user, such as in the case of activation software, the accused product may still infringe because it is configured to be capable of an infringing function. *See Fantasy Sports Props. v. Sportsline.com, Inc.*, 287 F.3d 1108, 1118 (Fed. Cir. 2002).

VII. IDENTIFICATION OF THE ARTICLES SUBJECT TO REALTEK’S RULING REQUEST LETTER

In its Ruling Request Letter, Realtek did not identify any specific models of products that are subject to its request. In response to AMD’s request for an identification of the specific models for which Realtek is seeking a ruling, and physical samples, technical documents, and source code of the same, Realtek furnished three development boards: “Sample Board 1,” “Sample Board 2,” and “Sample Board 3” (the “RRL Development Board Articles”). Realtek’s counsel further represented via email that Sample Board 1 contains the RTD2851MPJ chip, which contains the Mali-G31 GPU and corresponds with the non-patched driver code, r26p0. Ex. 39, Christopher Higgins Annotated Email (02/23/2024) at 2. Sample Board 2 contains Realtek’s “new chip,” RTD2951M, which contains the Mali-G31 GPU and corresponds with the Patched Mali Driver code, r26p0. *Id.* Sample Board 3 also uses Realtek’s “new chip,” RTD2951M, which contains the Mali-G31 GPU, but instead corresponds with the non-patched driver code, r26p0. *Id.* The RRL Development Board Articles, together with the RTD2851MPJ and RTD2951M Standalone Realtek ICs incorporated therein, shall be referred to collectively as the “RRL Articles.”

As noted in Sections IX.A.3. and X.D., *infra*, Realtek furnished a source code computer

for inspection by AMD’s experts. Ex. 1, Reinman Decl. ¶ 19. The source code computer contains an incomplete collection of portions of the source code that Realtek allegedly incorporates onto the RRL Articles. *See* Section XIII.3; *see, e.g.*, Ex. 1, Reinman Decl. ¶ 117, ¶¶ 127-129.

**VIII. REALTEK’S PATCHES TO THE DRIVER SOFTWARE DO NOT ALTER THE
HARDWARE CONFIGURATION UNDERLYING THE COMMISSION’S
FINDINGS OF INFRINGEMENT**

The categories of Realtek articles excluded from entry into the United States are: (1) Standalone Realtek ICs containing graphics processing units (“GPUs”); (2) printed circuit board assemblies (“Realtek Development Boards”) containing the same; and (3) digital televisions containing the same. *See* Ex. 7, LEO ¶ 2.

Realtek concedes that it has not made any changes to any standalone Realtek Integrated Circuit articles that fall under category (1), and is therefore not entitled to a ruling authorizing the entry of these Realtek Integrated Circuits. Instead, Realtek modified certain “driver software” to create what Realtek calls a “patched driver.” Realtek also concedes that its driver software (as well as the new Patched Mali Driver software) is not part of, or even combined with, the Standalone Realtek ICs at the time of their importation into the United States. For these reasons, alone, Realtek’s request for a ruling authorizing the entry of Standalone Realtek ICs should be denied.

But Realtek is similarly not entitled to a ruling authorizing the entry of any articles that fall under categories (2) and (3) above, i.e. Realtek Development Boards and unlicensed digital televisions that incorporate a Realtek Integrated Circuit. This is because, just as with Realtek’s Standalone ICs, Realtek’s Patched Mali Driver software is not part of, stored on, or combined with, the Standalone Realtek ICs that are included in the Realtek Development Boards and unlicensed televisions. In fact, the driver software does not even execute on the GPU that the Commission found to infringe. Instead, the driver software is stored on a separate memory chip that is located on the Realtek Development Board, and it is executed by a central processing unit

(“CPU”)—not the infringing GPU. During operation, the driver software acts as a mere conduit, to allow the operating system and user applications that are stored on the Realtek Development Board (and executing on the CPU) to transmit shader programs to the GPU for execution. However, at no point does the driver software (or Patched Mali Driver software) ever modify the GPU functionality or combine with the GPU in any way. *See*, Ex. 1, Reinman Decl. ¶132; Ex. 5, Pfister Decl. ¶26; Ex. 11, Hrg. Tr. (Pfister) at 561:11-20.

As explained above, the Commission found that the configuration of the Standalone Realtek ICs, in hardware, directly infringed claims 15, 19, and 20 of the '381 Patent. In reaching its decision, the Commission relied on evidence submitted by AMD, including its expert Dr. Glenn Reinman, which proved that the Standalone Realtek ICs directly infringe each of these claims. Recognizing that its Patched Mali Driver software does not and cannot modify the actual infringing features of the products, Realtek's Ruling Request Letter attempts to rewrite the Commission's decision. Specifically, Realtek argues that the Commission's infringement findings were based on “a combination of hardware and software,” attempting to tie the Commission's infringement findings to features and functions found in the Mali Driver. Realtek is simply wrong.

As explained below, nowhere in its final determination did the Commission find that **Mali Driver** or any other software in the Standalone Realtek ICs infringed Claims 19 and 20. Just the opposite, the Commission expressly held that all of the infringing features of accused Realtek products were implemented in the **Mali GPU** (which is a hardware component within the Standalone Realtek IC).

1. The Commission Found that the Mali Driver is not Part of the Infringing Devices but Instead Serves as the Scheduling Module in the Environment in which the Infringing Devices Operate

As a threshold matter, the Commission found that the Mali Driver works with, and is not part of, the infringing products. *See* Ex. 1, Reinman Decl. ¶¶ 53-54 (citing ID at 127, 129). More

specifically, the Mali Driver—which is part of the Operating System—serves as part of the environment in which the infringing products operate, and is a possible source of the tasks that are received by the claimed plurality of engines in the Standalone Realtek ICs. *See id.* ¶¶ 62-63. This is precisely why the Commission held that Standalone Realtek ICs (which do not include the Mali Driver), by themselves, directly infringe the claims and are subject to the exclusion order due to their configuration in hardware.

In fact, Realtek argued repeatedly during the 1318 Investigation that the scheduling module recited in Claim 15 was part of the claimed “computing system.” *See, e.g.*, Ex. 28, RIB at 96-100. These arguments were rejected by the ID, and that decision was adopted by the Commission. Ex. 8, ID at 129-130. First, the ID correctly found that the preamble “a computing system comprising” was not limiting, and noted that Respondents had conceded as much. Ex. 8, ID at 129. Second, the ID correctly found that the scheduling module is outside of the first processing unit (*e.g.*, the GPU). *Id.* (“A person of ordinary skill in the art would understand the ‘scheduling module’ and ‘second processing unit’ are outside of the GPU, as every relevant ’381 Patent figures shows and the prosecution history records—a point Respondents effectively concede.”) (citing in part RIB at 96-100). This is precisely why the Commission found that the LEO not only applied to the system products that include both a Realtek Integrated Circuit and the Mali Driver, but to the Standalone Realtek ICs that **do not** include the Mali Driver. Ex. 11, LEO ¶ 2.

Despite these findings, Realtek now again argues that the Commission required the scheduling module (*i.e.*, the Mali Driver) be a part of the claimed “computing system” to find infringement. Ex. 3, RRL at 2 (“Together, the GPU, the device driver, and the CPU comprise the ‘computing system’ that is claimed in the ’381 Patent.”); *id.* at 19-20 (“Claim 15 claims a ‘computing system’ that includes a GPU (the ‘first processing unit’) and the device driver for the

GPU, which runs on a CPU (the ‘scheduling module associated with a second processing unit’’). Realtek continues to advance this false premise because it has made **zero** modification to the actual Standalone Realtek ICs found to infringe at the ITC, and has only made modifications to the driver software that is not required by the claims, and included only on Realtek Development Boards.

When interpreted consistently with the Commission’s findings, the Mali Driver serves as part of the environment for the Standalone Realtek ICs that are subject to the LEO. *See, e.g.*, Ex. 8, ID at 130, 132 (holding that AMD met its burden in connection with its showing of domestic industry by demonstrating that “*some* operating system driver is used” and that “the question is whether a preponderance of all evidence shows that the ‘plurality of engines’ within the DI Products are ‘configured to’ ‘receive...a plurality of tasks’ from ‘a scheduling module associated with a second processing unit.’”) (emphasis in original); *see also* Ex. 1, Reinman Decl. at Section VII.B.2.

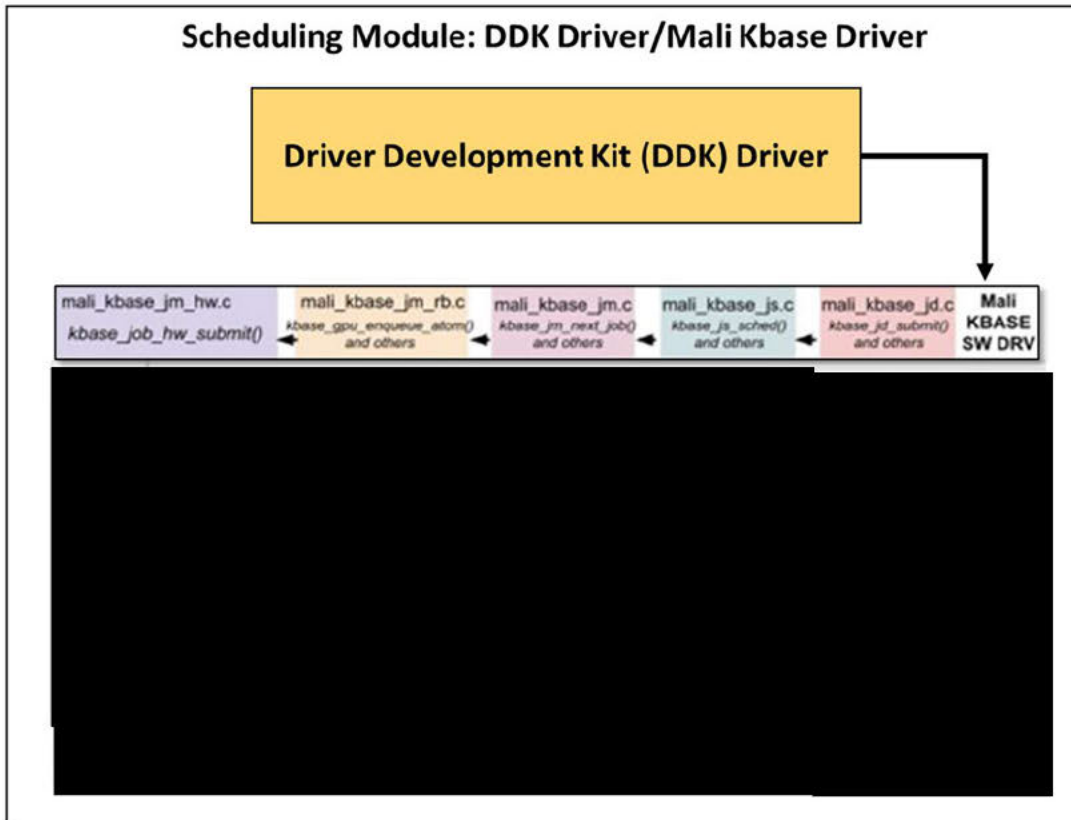
The use of the Mali Driver to serve as part of the environment of the infringing apparatus is consistent with the directly analogous findings of the Federal Circuit in *Silicon Graphics, Inc. v. ATI Techs., Inc.*, 607 F.3d 784 (Fed. Cir. 2010). In *Silicon Graphics*, the claims were also directed to a computer system that includes “a processor for performing geometric calculations on a plurality of vertices of a primitive; [and] a rasterization circuit...that rasterizes the primitive according to a rasterization process which operates on a floating point format.” *Id.* at 788. Similar to this case, the accused products were graphics chips and processors. The Federal Circuit held that, “[w]here, as here, a product includes the structural means for performing a function, it can still infringe ‘separate and apart’ from the operating system that is needed to use the product.” *Id.* at 794. The Court went on to hold, that with respect to dependent claims that were directed to “a rasterization circuit coupled to the processor that rasterizes’ primitives in a certain way and ‘a

frame buffer coupled to the rasterization circuit for storing’ certain data,” the claims would be infringed “[e]ven if the products cannot rasterize or store absent an operating system.” *Id.* at 795.

Additionally, just as in *Finjin*, AMD has valid claims that are broadly construed to cover hardware components (*i.e.*, engines) that are not just capable, but are configured to perform by design a specific function of receiving from a scheduling module. *Finjin* is straightforward, the focus is on the code and circuitry of the claimed component only. Here, just as in *Finjin*, the claimed component is an engine, thus, the focus is what that engine’s code and circuitry is written to perform. Where circuitry exists to perform receiving a plurality of tasks from a scheduling module, infringement occurs and the CBP should look no further because the infringing configuration is sufficient to satisfy Federal Circuit precedent regardless of if that code is “active or enabled” by a driver. *See Finjan*, 626 F.3d at 1204-05. Indeed, the very nature of Realtek’s Integrated Circuit products dictates that they are designed to be integrated into downstream customers’ devices such as digital TVs. The downstream customers import, modify, and compile, their own software for use in combination with what is provided by Realtek’s integrated circuits without any downstream control by Realtek.

The undisputed evidence presented at the hearing with respect to the Mali Driver software (aka the DDK Driver/Mali Kbase Driver) was that it serves as part of the environment for processing application tasks that are received by the plurality of engines of the GPU. *See* Ex. 11, Hrg. Tr. (Reinman) at 135:7-16:1 (“when [the application processor] requires graphics to be displayed on the screen, it uses the graphics device driver, that would be the scheduling module...to send a graphics job to the GPU.”). This relationship of the Mali Driver, as it relates to the actual infringing configuration of the Standalone Realtek ICs is demonstrated by Dr. Reinman with reference to the below source code diagram, which illustrates the DDK Driver/Mali Kbase

Driver as the software layer that serves to provide Jobs to the plurality of engines (*i.e.*, Job Slots). *See also*, Ex. 15, JX-0192C (Larri Tr. (11/01/2022)) at 161:5-162:8 (“Q. This module [mali_kbase_jd.c] submits jobs to the job scheduler, correct, that’s one of its functions? A. That’s correct.”). And as explained below, the configuration of the Job Slots to receive these Jobs (*i.e.*, claimed tasks) from such a driver is what is required by the claims.



Realtek’s reliance on *Nazomi Communications, Inc. v. Nokia Corp.*, 739 F.3d 1339 (Fed. Cir. 2014) and ALJ Elliot’s decision in 337-TA-1170 is inapposite. In *Nazomi*, the Court expressly held that the claims required “a hardware-software combination that must perform the described functions.” *Nazomi*, 739 F.3d at 1345. The accused products in *Nazomi* included an optional Jazelle hardware feature that could process Java’s stack-based instructions, but this hardware feature required specific software known as the Jazelle Technology Enabling Kit (“JTEK”). *Id.* at 1342. Without this enabling software, the Jazelle hardware could not process the stack-based

instructions **at all**. *Id.* That is, and as was central to the Court’s decision in *Nazomi*, the **hardware alone** could not process stack-based instructions. *Id.* The Court distinguished this particular hardware-software combination from the holding in *Silicon Graphics*—explaining that “[w]e have repeatedly distinguished a description of the environment in which a claimed invention operates from a limitation on the claimed invention itself.” *Id.* at 1345. *See also id.* (distinguishing the claims at issue from those in *Silicon Graphics* on the grounds that “the claims do not cover hardware that contemplates an environment where it could be combined with software, but rather require a hardware-software combination that must perform the described functions.”).

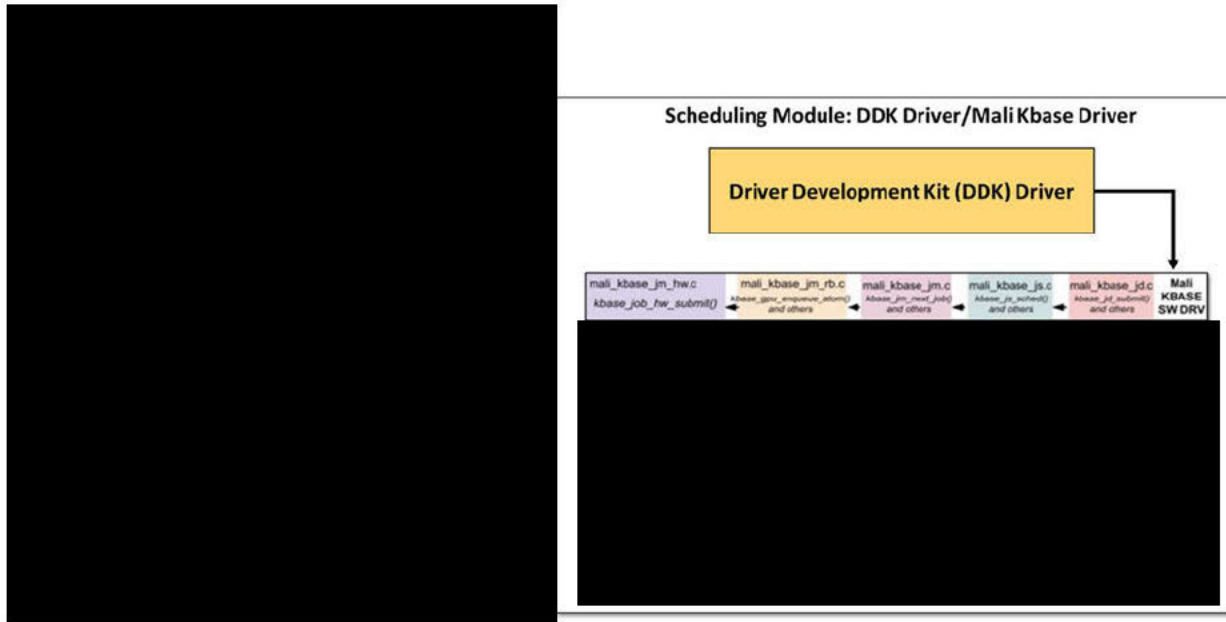
Similarly, ALJ Elliot’s decision in the 1170 Investigation was predicated on a finding that every claim of the asserted patents required software to carry out the functional limitations of the claims. *Certain Mobile Devices with Multifunction Emulators*, Inv. No. 337-TA-1170 Initial Determination (Mar. 16, 2021), *aff’d* Notice, 86 Fed. Reg. 38762 (July 22, 2021). Specifically, Judge Elliot held that the claimed features directed to “track of magnetic strip data” and “plurality of digital representations” could only be performed in software. *Id.* at 44-45. In stark contrast to the facts in the 1170 Investigation, the claims at issue and the Commission’s finding of infringement here are based on the actual design and programming **in hardware** to carry out the recited functions of the claims. And this hardware based design and programming has not been modified by Realtek.³

³ Realtek suggests that “if the Mali driver software (the claimed ‘scheduling module’) does not provide a ‘graphics-processing task’ and a ‘general-compute task’ as part of the ‘plurality of tasks’ there can be no infringement of Claim 19.” But as discussed in Section IX.A, Realtek has not disabled the driver from providing a graphics-processing task and a general-compute task to the GPU. Further, as discussed in Section IX.B, Realtek’s modifications to the Mali Driver did not eliminate all priorities associated with the ARM Jobs, and in fact the patches introduced additional priorities associated with the ARM Jobs.

Not only is Realtek’s position inconsistent with controlling case law, but it is inconsistent with its own expert, Dr. Pfister’s previous opinion as to whether the Mali Driver, as a collection of software, could serve to satisfy the limitations of Claim 15. Specifically, Dr. Pfister, in his expert report submitted during the ITC investigation, opined that the claimed “scheduling module” is one that “schedules the tasks to be executed by the shader core.” Ex. 29, CX-1250C (Expert Report of Dr. Hanspeter Pfister (12/28/2022)) ¶ 184. Dr. Pfister stated conclusively that a POSITA would not “view a graphics driver, which is a collection of software, to constitute a ‘scheduling module’ as recited in claim 15.” *Id.* Now faced with the Commission’s determination and an LEO, Realtek and Dr. Pfister take the position that the Mali Driver was not only required to finding infringement of Claim 15, but that a modification to the Mali Driver can disable configuration of the GPU.

2. The Commission Found that the Claimed Plurality of Engines in the Infringing Devices Were Implemented in Hardware—not Software

The Commission also found that **hardware** architecture of the Standalone Realtek ICs—not the Mali driver—constituted the plurality of engines of Claims 15, 19, and 20. Specifically, the Commission found that the ARM Mali GPUs each included a plurality of Job Slots which serve as the plurality of engines of Claim 15. Ex. 8, ID at 103-104; Ex. 8, ID at 17. This structure is illustrated and described in connection with ARM’s Technical Documents (annotated below to the left) and Dr. Reinman’s source code diagram of the Dvalin GPU (reproduced below to the right), which demonstrate the use of a plurality of Job Slots and respective [REDACTED] by the GPU to receive tasks from a scheduling module, such as the Mali Driver.

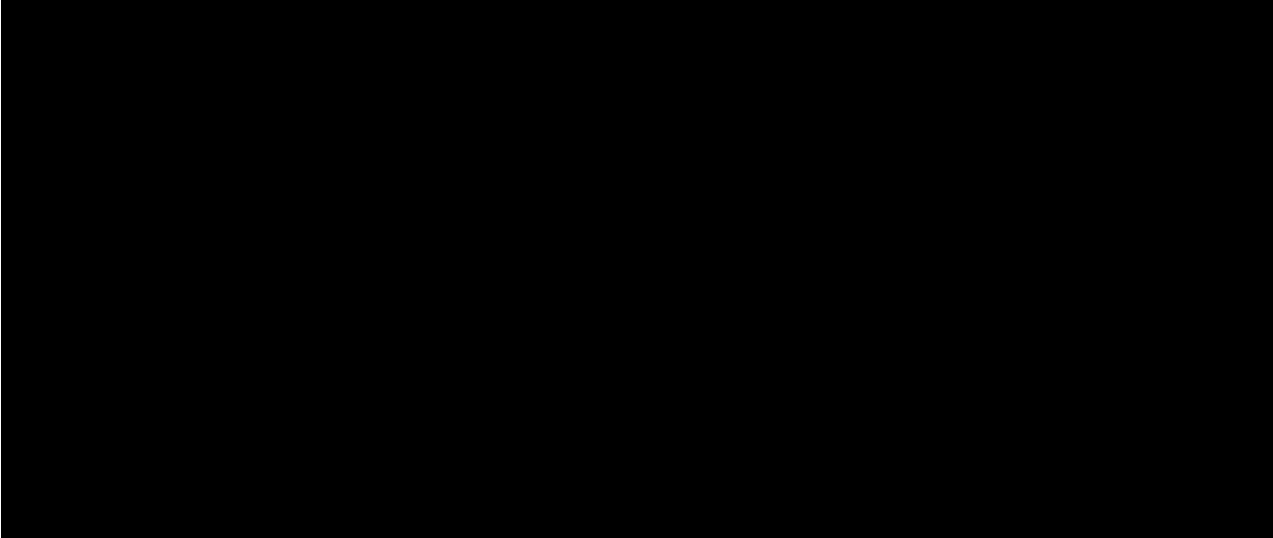


Ex. 42, CX-0773C (ARM Mali Dvalin GPU Technical Reference Manual) at CX-0773C.0026

As acknowledged in Realtek’s Ruling Request Letter, the ID and FD both construed the term “configured to” to require a particular “design or implementation that achieves the stated effect(s).” Ex. 8, ID at 98; Ex. 9, FD at 40. Throughout the investigation, first at the Markman proceeding, then in its pre-hearing brief, at the hearing, and its post-hearing brief, Realtek took the position that the plurality of engines as recited in the claims required software or were implemented only in software. *See, e.g.*, Ex. 10, Order No. 31 at 39-40; Ex. 28, RIB at 69-71. These arguments were rejected, both during Markman, and again following the evidentiary hearing. Ex. 10, Order No. 31 at 39-41; Ex. 8, ID at 42-43. Specifically, the ID held that “Respondents have not shown that the plain and ordinary meaning of ‘engine’ requires a software component.” Ex. 8, ID at 94.

The design and implementation of the engines, in hardware, was central to the benefits of the novel architecture of the engines as claimed in the ’381 Patent. As the inventor Mr. Rex McCrary explained:

[REDACTED]



The ID relied on AMD's evidence, including Dr. Reinman's unrebutted testimony, regarding the configuration of the Mali GPUs to receive and process jobs into work interpretable by the GPU. Ex. 8, ID at 103-105. That is, the Job Slots configuration, in hardware, to receive and process different types of tasks from the driver satisfied the requirements of Claim 15. The undisputed evidence at the hearing also included testimony from Dr. Reinman that described the actual design and implementation of the plurality of engines (*i.e.*, Job Slots) in hardware to receive the plurality of tasks by servicing command buffers.

Q. Okay. I believe, earlier, you talked about the job slots, the engines receiving the job from a driver; is that right?

A. Yes, sir.

Q. Did you review the source code for that driver?

A. I did.

Q. What did the source code tell you about the functionality of that driver?

A. It showed me that the engines -- first of all, the hardware source code showed me that the engines were... That the engines themselves were configured to receive from a scheduling module.

Ex. 11, Hrg. Tr. (Reinman) at 137:1-15.

Q. What was the basis for your conclusion that the plurality of engines service a command buffer?

A. Sure. So looking at the individual RTL and driver code, I found that there were ring buffers that would be provided from -- the Mali driver would be providing tasks through the ring buffers that would be consumed by the job manager itself, and the individual job slots of the job managers.

So there are command buffers in the form of ring buffers that facilitate that communication.

Id. at 141:5-15.

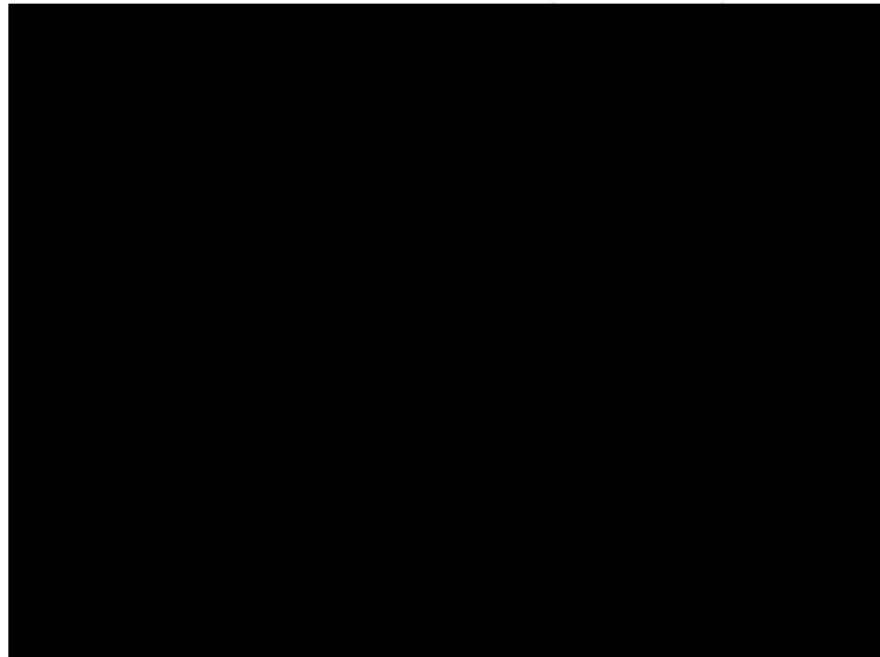
Realtek now encourages CBP to ignore the basis of the Commission's decision and underlying claim construction by suggesting yet again in this proceeding that the claims require a software-hardware combination. This is plainly false, and was expressly rejected in the 1318 Investigation. There is no requirement in the claims for software to carry out the functions of receiving tasks by the plurality of engines. Rather, the findings of the Commission were based on the configuration of the engines as part of their design and implementation in hardware to receive the plurality of tasks from an environment that includes a CPU, and a driver that is part of a complete operating system.

3. The Commission Found that the Claimed Shader Core in the Infringing Devices Was Implemented in Hardware—not Software

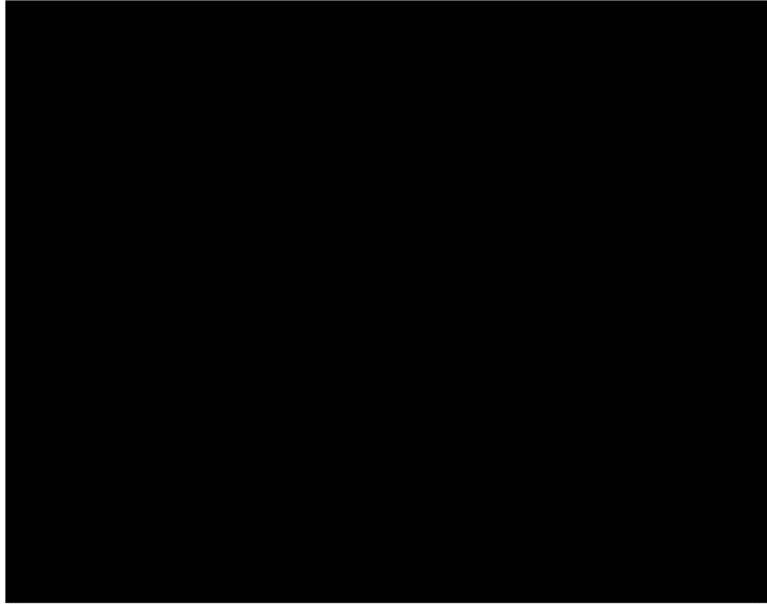
The Commission likewise found that the shader core, and its configuration to execute a first task while executing a second task of the plurality of tasks, was met through the hardware configuration of the Mali GPUs. *See* Ex. 1, Reinman Decl. at Section VII.B.2. Specifically, the Commission found that the execution engines of ARM Dvalin (Bifrost) GPUs and the processing units of the ARM Natt (Valhall) GPUs are the claimed shader core. Ex. 8, ID at 113-118. Referencing the Dvalin architecture as representative, the ID found, and the Commission agreed, that that the execution engines operate in parallel and receive work associated with different ARM Jobs such that those Jobs are executed in overlapping manner. Ex. 9, FD at 51-52, 51 n. 40. Specifically, the ID found that “[t]he evidence shows that execution engines 0, 1, operate in

parallel, and that each holds or operates on more than one warp in any given moment albeit in different stages of the pipeline.” Ex. 8, ID at 116 (citing testimony from Respondents’ source code expert Mr. de la Iglesia, Dr. Reinman, and ARM’s corporate representative Mr. Larri). The ID also found, and the Commission agreed, that the shader core was configured to receive the claimed tasks through the design, in hardware, of the ARM GPUs, and the execution engines’ configuration to receive work from different jobs through the operation of a thread group creator (TGC) arbiter (`dvalin_wm_tgc_arbiter`). Ex. 9, FD at 18, 21, 52.

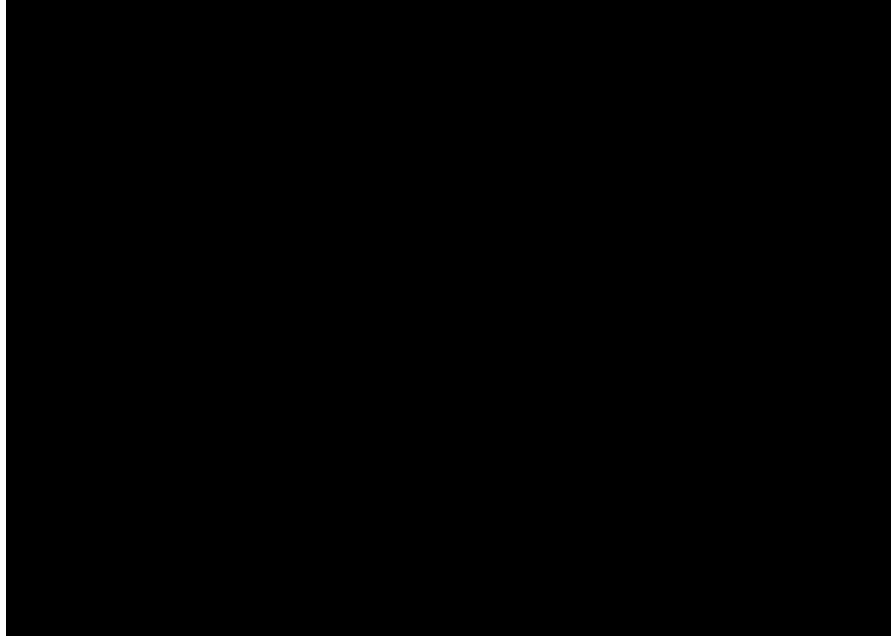
This architecture is demonstrated in the ARM Technical documentation and source code diagrams prepared by Dr. Reinman, entered into evidence, and relied upon by the ID and the Commission, which demonstrate the architecture in hardware of the execution engines and their configuration to receive tasks from the [REDACTED]. As shown in the diagram below, the Dvalin GPU includes two execution units, labeled Execution engine 0 and Execution engine 1 by ARM.



CX-0773C (ARM Mali Dvalin GPU Technical Reference Manual) at 30. As shown in the below
RTL source code diagram, the execution engines are implemented as [REDACTED] in the
hardware code.



These execution engines receive portions of [REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]. This hardware
architecture, and the functions performed by [REDACTED], is illustrated below.



As shown in the above diagram, the [REDACTED]
[REDACTED]
[REDACTED]. Ex. 1, Reinman
Decl. ¶ 136. These requests are passed to the execution engines as warps in [REDACTED]
[REDACTED]
[REDACTED]. *id.*; Ex. 11, Hrg. Tr. at 161:17-163:22; *see*
also Ex. 9, FD at 51-52, 51 n. 40.

This hardware configuration is precisely what the Commission relied upon in finding that the shader core was configured to, and is not just capable of, executing a first task while executing a second task in connection with the limitations of Claim 15. It is this specific configuration, in hardware, that the Commission found satisfied the requirements of Claim 19 for where the first task is a graphics task and the second task is a general-compute task. Ex. 8, ID at 116; Ex. 9, FD at 40 (“a deliberate feeding of alternating vertex/compute or fragment/pixel warps by the [REDACTED] [REDACTED] to the collective plurality of engines will satisfy the ‘configured to’ claim limitation.”). The ID went on to explain, with reference to ARM’s documentation, that the [REDACTED]

██████████ Ex. 8, ID at 118 (citing CX-1730C (Warp Manager) at 30).

Further, the ██████████ within each execution engine (for Bifrost GPUs) or Processing Unit (for Valhall GPUs) is specifically designed to service the requests that are ██████████ ██████████ of the execution engine. *See* Ex. 11, Hrg. Tr. (Reinman) at 160:19-161:6 (identifying the interfaces programmed in source code between ██████████ ██████████); 242:15-24 (describing the ██████████ ██████████); 143:2-21 (describing the design of the Accused Products which achieves the transfer of Jobs from the Job Slots to the shader core ██████████). This configuration of the shader core, which Realtek ignores, is a part of the design and implementation to achieve the stated effect of executing multiple tasks at the same time.

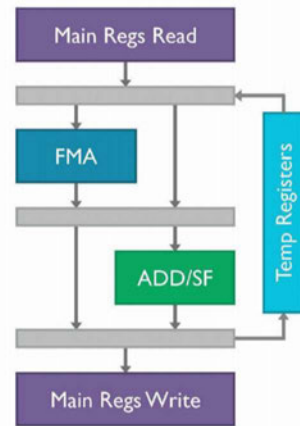
Realtek argued extensively at the ITC that the shader core was not configured to execute a first task while executing a second task as recited in the claims, and was merely capable of doing so. Ex. 9, FD at 45, 51, 59. This argument was rejected. As discussed above, the ID and the Commission found that the execution engines of the Mali GPU were configured to execute the tasks in overlapping manner through their hardware configuration and pipeline design. Ex. 8, ID at 95-98, 114-115. The pipeline architecture of each execution engine provides the requisite configuration, a specific design in hardware, to execute multiple tasks during different stages of the pipeline. As Dr. Reinman explained, the Accused Products include an execution pipeline which includes multiple stages of execution, where quads from different Arm Jobs are within the execution pipeline at the same time. Ex. 11, Hrg. Tr. (Reinman) at 406:17-407:21. Dr. Reinman's testimony, that there are multiple stages of the pipeline where execution occurs, is supported by

the ARM technical documents. *See, e.g.*, Ex. 44, CX-0483 (Bifrost GPU Architecture) at CX-0483.14 (reproduced below, illustrating at least two different execution stages including the FMA and ADD/SF units, *i.e.*, “ALUs”); *See also, e.g.*, CX-1724C at CX-1724C.7.

Bifrost execution engine

- Executes quad-parallel scalar operations
 - 4x32-bit multiplier FMA
 - 4x32-bit adder ADD
 - Adder includes special function unit
- Smaller and more area efficient
- Simplified layout eases compilation
 - Better scheduling in today's code
 - Better utilization
- One instruction word contains two instructions

©ARM2016



ARM

Dr. Reinman’s testimony was not disputed. In fact, both Mr. de la Iglesia and Dr. Pfister agree that the execution engines are configured to execute warps in a pipeline design. For example, Respondents’ own source code expert, Mr. de la Iglesia, testified that “[t]he warp exist[s] and it’s executing...through all of those stages. There is obviously a different warp, you suspect, behind and in front of it that are all executing, but they are all at – but **they are at different stages of the processing pipeline.**” *See* Ex. 11, Hrg. Tr. (de la Iglesia) at 867:12-868:14 (quoting and referencing CX-0675C (Iglesia Tr. (01/06/2023)) at 152:17-153:9) (emphasis added); *see also* Ex. 30, CX-0675C (Iglesia Tr. (01/06/2023)) at 153:11-156:19. This is precisely the hardware configuration that the ID and the Commission relied upon when referencing the “different stages of the pipeline.” Ex. 8, ID at 116; Ex. 9, FD at 52.

The evidence shows that the execution engines 0, 1 operate in parallel, and that each holds or operates on more than one warp in any given moment albeit **in different stages of the pipeline.** *See, e.g.*, Hr’g Tr. (de la Iglesia) at 865:20-867:20 (“I think we all agree how it works”); Hr’g Tr. (Reinman) at 406:17-407:23; JX-

0192C (Larri) at 128:14-23; CX-1724C.9-10.

Ex. 8, ID at 116 (emphasis added).

This pipeline architecture of the shader core in the Accused Products is a configuration (*i.e.*, a design or implementation) of the shader core in the Accused Products that achieves the stated effect of executing multiple different tasks at the same time. Moreover, this configuration of the processing elements of the execution engine (Bifrost GPUs) or Processing Unit (Valhall GPUs) in a pipeline aligns with the '381 Patent's disclosure regarding spatial and temporal partitioning as discussed above. Indeed, this is precisely what that ID found, and which Realtek now disregards:

Importantly, none of this hardware configuration and design has been modified by Realtek as a result of the patches introduced to the DDK Driver. Instead, Realtek's software modification to the driver is premised on the previously rejected arguments regarding capability, rather than configuration, of the ARM Mali GPUs.

4. The Commission Found that Hardware—not Software—in the Infringing Devices Processed Compute Tasks and Graphics Tasks as Required by Claim 19

In connection with Claim 19, the ID and the Commission also found that the Mali GPUs were designed, in hardware, to process tasks for general computation, including compute ARM Jobs and Tasks, through Job Slots 1 and 2 (and their respective endpoint), and fragment tasks (which are specific to graphics) through Job Slot 0 and its respective endpoint. Ex. 8, ID at 123-124. The ID's rationale was based on Dr. Reinman's unrebutted testimony and the "clear disclosure in the ARM documentation that each job slot is reserved for certain types of jobs, differentiating between, for example, fragment (claimed "graphics-processing task") and compute (claimed "general-compute task"), and that expert testimony presented above regarding "round-robin" issuance to

execution engines (claimed ‘shader core’).” *Id.* (Citing CX-0773C ARM Mali Dvalin GPU Technical Reference Manual at 27).

Given the configuration of the Mali GPUs, again in hardware, to execute graphics tasks (*i.e.*, Fragment Jobs) in Job Slot 0 (██████████), as well as general-compute tasks (*i.e.*, tasks performing general-computations including compute tasks in the form of ██████████ ██████████ requested by ARM’s ██████████ and graphics tasks in the form of vertex shader programs requested by ARM’s ██████████) in an overlapping manner, the purported changes to the Mali Driver have no impact on this finding of infringement. Importantly, Realtek submitted no expert testimony at the hearing with respect to the limitations of Claim 19, and instead relied exclusively on its arguments in connection with independent Claim 15. *See* Ex. 9, FD at 51. As such, there was no distinction made between the types of ARM Jobs that serve as general-compute tasks, and the ID’s decision should properly be interpreted to include as general-compute tasks all tasks which perform general computation, regardless of ARM’s nomenclature, and its driver changes to ██████████, to allow them to request execution of both compute tasks and graphics tasks.

5. The Commission’s Decision in Connection with Prioritization for Claim 20 was Found in the Infringing Hardware Code and Not Just in the Software Driver

In connection with Claim 20, the ID highlighted and cited to AMD’s briefing which referred to two priority classifications, one in the hardware of the Mali GPUs and the other in the driver. Importantly, as noted in the ID, Realtek again presented no evidence or testimony at the hearing to rebut Dr. Reinman’s expert testimony in connection with prioritization by the Mali GPUs or the Mali Driver as it relates to the reception of tasks by the shader core.

The first form of prioritization relied on by the Commission’s decision was in connection with the prioritization of Job Slot 2 over Job Slot 1 by the module, in hardware, *_jm_job_epalloc. Specifically, the ID, expressly referenced the following passage from AMD’s Post-Hearing Brief:

Further, each Accused Product has three Job Slots. Jobs (*i.e.*, ’381 Patent tasks) in Job Slot 2 will be received by the shader core based on having a higher priority than jobs in Job Slot 1, as the *_jm_job_epalloc module checks all non-fragment Job Slots (e.g. Job Slots 1 and 2), and will always select the highest numbered slot (*i.e.*, Job Slot 2) for the receipt and further execution by the shader core. Hrg. Tr. (Day 1) (Reinman) at 176:25-177:25; CX-1850C; CX-1851C.

Ex. 8, ID at 124 (citing CIB at 105-106); *see also* Ex. 31, CIB. The ID found that “AMD has met its burden to show, as testified to by Dr. Reinman, that ARM jobs (claimed ‘tasks’) located in Job Slot 2 have priority over those in Job Slot 1, and this priority impacts their reception by the execution engines (claimed ‘shader core’).”

As explained by Dr. Reinman, the module [REDACTED] is part of the job manager, within the hardware of the Mali GPUs, and is implemented in the hardware’s RTL code. Specifically, for the Dvalin GPUs, this module is part of the hardware file [REDACTED] Ex. 1, Reinman Decl. ¶ 107, 173, *see, e.g.*, [REDACTED] produced as RTK-1318-SC-00000973-RTK-1318-SC-00000988. Realtek’s Ruling Request Letter allegedly “disables” this prioritization by masking how jobs are submitted from the driver to the GPU, without actually removing job slot 2 from the GPU or disabling any of the programming of the [REDACTED] module in the hardware code. Ex. 1, Reinman Decl. at XI.D.3. Moreover, the net effect of Realtek’s patch results in the prioritization, in the hardware code, of Job Slot 0 as acknowledge by ARM’s Mr. Larri and as further explained in Section IX.B.3 below. *See* Ex. 12, Larri Tr. (03/19/2024) 231:19-232:12.

The second form of prioritization presented at the hearing and cited by the ID was in connection with the function [REDACTED] of the Mali Driver, which checks the prioritization of ARM Jobs when scheduling them to the GPU.

[T]he scheduling module (*i.e.*, Mali Driver) considers the priority of tasks (*i.e.*, ARM Jobs) by using function [REDACTED] to check a specific job slot at a specific priority level. *Id.*; CX-1850C at CX-1850C.1 ([REDACTED]). This in turn results in the tasks received by the shader core based on their priorities. *Id.*; *See also, e.g.*, CX-0529 at CX-0529.12 (“atom ‘X’ has a higher priority than atom ‘Y’ . . . when choosing the next atom to run, atom ‘X’ will always be chosen instead of atom ‘Y’”). Higher priority ARM Jobs will have first access to the Job Slots of the GPU, and therefore have a greater opportunity to obtain the shared compute resources of the GPU (*i.e.*, the shader core). *Id.*

Ex. 8, ID at 124 citing AMD’s Post-Hearing Brief at 105-106. As discussed in Section IX.B.3 below, Realtek’s modification to this second prioritization associated with job chains (atoms) made no change to the function [REDACTED], which still checks and considers the priority of each job chain (atom) and prioritizes jobs at least based on age or sequence number. Ex. 1, Reinman Decl. ¶¶ 107-109, 147-149. Moreover, as discussed in Section IX.B.6, the effect of the patches applied by Realtek is a shifting of the prioritization that is considered to the Mali GPU hardware, rather than actually removing prioritization of jobs.

6. Realtek Admits that it does not make any changes to its Hardware Source Code that was Found to Infringe at the ITC.

The configuration of the Standalone Realtek’s ICs is defined in RTL source code. *See* Ex. 1, Reinman Decl. ¶ 115. As Realtek has admittedly made no changes whatsoever to the RTL Source Code, Realtek has failed to meet its burden. *See* Reinman Decl. at Section X.B.

IX. THE PATCHED MALI DRIVER DOES NOT OPERATE AS REALTEK ADVERTISED IN ITS RULING REQUEST LETTER

As discussed in Section VIII, *supra*, the Standalone Realtek ICs excluded from entry into the United States do not incorporate any driver software at the time of importation into the United States. Thus, Realtek’s patches to the driver software are not relevant to that category of excluded

article. Even Realtek Development Boards that incorporate a software driver on a separate memory chip, Realtek's changes to the Mali Driver have no effect on the configuration of the Mali GPU found to infringe, and therefore cannot circumvent the Commission's findings of infringement. But even if the EOE Branch were to entertain Realtek's arguments that the Patched Mali Driver software somehow modifies the configuration of the Mali GPU in Realtek Development Boards found to infringe (and it does not), the EOE Branch should still deny Realtek's Ruling Request Letter because the Patched Mali Driver software does not have the effect that Realtek claims for the reasons set forth below.

A. Claim 19: The "New Mali Driver" Patches Simply Reconfigure Vertex Jobs to Initiate the Execution of General-Compute Tasks

As mentioned, Realtek's first attempt to remove simultaneous execution of graphics and compute tasks in the Mali GPUs by using a software patch failed. Originally, ARM used compute jobs to submit compute tasks, and vertex/fragment Jobs to submit graphics processing tasks, to the shader core for execution. Realtek's initial version of a software patch prevented compute jobs from submitting any compute task to the shader core for simultaneous execution with any graphics processing tasks submitted by fragment/vertex jobs. Realtek and ARM tested this initial version of the patch and concluded that it did not meet their performance requirements. Ex. 16, Performance Enhancement Patch at RTK-177-000004 ("the patches ensure that Compute jobs cannot run in parallel with graphics jobs...which may cause a drop in performance."); Ex. 12, Larri Tr. at 30:20-32:6. Specifically, Realtek and ARM found that preventing certain general-compute tasks (called [REDACTED] programs) from executing simultaneously with certain graphics processing tasks (called fragment shader programs) significantly decreased the Mali GPU's performance. *See* Larri Tr. at 83:1-6. As a result, Realtek and ARM worked together to develop the "Performance Enhancement Patch" that Realtek concealed from its Ruling Request Letter. Ex.

12, Larri Tr. at 97:13:-98:9 (“we produced a release note with the title, Performance Enhancement Patches. You might understand from that that they achieved performance enhancement.”).

To provide Realtek with cover, ARM designed the Performance Enhancement Patch to modify the driver software in a way that causes vertex jobs to submit both compute tasks and graphics processing tasks. Specifically, the Performance Enhancement Patch reconfigures vertex jobs to submit both compute tasks (in the form of the [REDACTED] programs) and graphics processing tasks (in the form of vertex shader programs) to the shader core for simultaneous execution. *See* Ex. 1, Reinman Decl. at Sections IX.B., X.C. In this way, the “Performance Enhancement Patch” is designed to cause the same exact general-compute tasks (called [REDACTED] programs) to be executed simultaneously with fragment shader programs. *See id.* at Section X.C. In other words, this Performance Enhancement Patch—which Realtek concealed from the EOE Branch—**reversed** the allegedly disabling effects of the redesign patch that Realtek disclosed in its Ruling Request Letter. *See generally* Ex. 16, Performance Enhancement Patch at RTK-177-000001-8; Ex. 12, Larri Tr. (03/19/2024) at 110:18-111:10 (“If the application processor does all the right things to cause those two types of shader programs to run on the GPU at the same time, then it’s possible that the pixel shader for a vertex shader could be running as a fragment shader.”). The only difference is Realtek now tries to disguise the compute tasks using its modified “vertex jobs.”

The Performance Enhancement Patch is embedded in the code of the physical samples that Realtek furnished to AMD, for which Realtek is seeking a ruling of non-infringement pursuant to its Ruling Request Letter to the EOE Branch. As admitted by ARM’s corporate representative, Mr. Guy Larri, in reconfiguring vertex jobs to cause them to submit [REDACTED] programs (compute tasks), ARM made no changes whatsoever to the [REDACTED] programs themselves. Ex. 12, Larri

Tr. at 86:8-12, 94:18-95:17, 96:1-18, 97:2-12. Thus the general-compute tasks (in the form of [REDACTED] programs) that execute simultaneously on the RRL Articles alongside fragment tasks (in the form of fragment shader programs) are the same exact general-compute tasks and graphics processing tasks that the Commission found to infringe Claim 19. Because Realtek’s sleight of hand does not actually prevent general-compute tasks from executing simultaneously with graphics processing tasks, Realtek is not entitled to a ruling of non-infringement of the RRL Articles. Ex. 1, Reinman Decl. ¶ 143-144.

1. Realtek’s Ruling Request Letter Fails to Describe the Effect of the Performance Enhancement Patch Embedded in the RRL Articles and Source Code on Realtek’s Claims

In its Ruling Request Letter, Realtek argues that RRL Articles furnished to AMD use the ARM patches to the DDK Driver to prevent general-compute tasks from executing simultaneously with graphics tasks by blocking compute jobs from executing at the same time as fragment jobs. First, ARM allegedly “disables” Job Slot 2 by removing access to Job Slot 2 through the patch ARM-MALI-PATCH-04-V03<DDK Version>.tar. With this patch, there were still two job slots, Job Slot 0 for fragment jobs and Job Slot 1 for other types of jobs, including vertex jobs and compute jobs. ARM then attempted to prevent the simultaneous execution of tasks submitted by a compute job, which would originate from Job Slot 1, simultaneously with 1) graphics jobs that would originate from Job Slot 1, or 2) fragment jobs, which would originate from Job Slot 0. To implement these changes, the ARM patches rely on a “helper” function ([REDACTED] that scans a [REDACTED] to determine whether the [REDACTED] contains a [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED] Ex. 1, Reinman Decl. ¶¶ 86-92, Ex. 6, Larri Decl. ¶ 22, 24.

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED] applicable code that attempts to prevent simultaneous execution of compute and graphics jobs in the patches does not apply to that job. Similarly, if the compute shader is not a [REDACTED] [REDACTED] is not set. *Id.*

During inspection of the RRL Articles and the source code on the source code computer, AMD's experts discovered that both the RRL Articles and the source code that Realtek furnished incorporate a "Performance Enhancement Patch." Ex. 1, Reinman Decl. ¶ 19. Remarkably, although the "Performance Enhancement Patch" is incorporated in the RRL Articles and their corresponding source code, Realtek never once referenced it in Realtek's Ruling Request Letter. Nor was this patch cited to or relied on in the declaration of its author, ARM's corporate representative Mr. Guy Larri. *See* Ex. 6, Larri Decl. Realtek's experts, Mr. de la Iglesia and Dr. Pfister, likewise make no reference to—and as of the date of submission of the Ruling Request Letter, apparently have not considered—this "Performance Enhancement Patch" or the corresponding Performance Enhancement Patch Release Note. *See* Ex. 14, Pfister Tr. (03/25/2024) at 13:3-15 ("I reviewed the ones [patches] that I was privy to have access to, which is a subset, I believe, of all of the ones that Mr. de la Iglesia listed."); Ex. 13, Iglesia Tr. (03/22/2024) at 29:1-15 ("I reviewed the source code that I was given"); Ex. 4, Iglesia Decl. ¶ 7 (listing the patches considered and excluding the Performance Enhancement Patch and corresponding Release Note). In fact, there is not a single reference to the use of [REDACTED] programs, or any change to [REDACTED]

Putting aside Realtek's decision not to disclose the Performance Enhancement Patch, the fact remains that the RRL Articles, and the source code furnished to AMD in support of Realtek's Ruling Request Letter, continue to infringe Claim 19 of the '381 Patent. Accordingly, the EOE Branch should deny Realtek's Ruling Request Letter.

As explained, when Realtek and ARM tested the patches described in Realtek's Ruling Request Letter, they realized that the performance of the Mali GPUs was unacceptably degraded. Ex. 12, Larri Tr. (03/19/2024) at 80:11-20. Specifically, the Mali GPUs include support for a certain type of compute task, which ARM refers to as a [REDACTED] program, that was impacted by the patches ARM developed in its first attempted design around. These [REDACTED] programs are formed as a distinct shader stage by the Mali Driver which [REDACTED]

Ex. 1, Reinman Decl. ¶¶ 96-99; RTK-177-SC-003051 () at lines 11, 13-15. As explained in ARM's patent filings:

The technology described herein executes constant program expressions of an original shader program in an initial “pilot” shader program for the original shader program in advance of a separate subsequent “main” shader program. The subsequent shader program can then use the results of the initial pilot shader program. Thus, the instructions emitted for a constant program expression may be executed only once in the initial pilot shader program, rather than having to be executed multiple times in the corresponding original shader program each time the result for the expression in question is required. This can therefore remove repeated

⁴ The only such reference to the Performance Enhancement Patch and to the use of [REDACTED] is found in Exhibit 4 to Realtek's Ruling Request Letter (Release Note: Behavior Change DDK Patches for Mali-G57 Natt), which states that a Performance Enhancement Patch was *not* necessary for the ARM Mali G-57 Natt GPUs.

calculations from the shading process.

Ex. 32, U.S. Patent No. 10,186,069 at 3:31-42. These pilot shader programs are effectively a mechanism by which ARM’s Mali GPUs perform generic computations on behalf of other shader programs to reduce redundancy of having to reperform these generic computations. Following the initial attempt to patch the GPUs to avoid infringement, however, ARM realized that continuing to execute these pilot shader programs simultaneously with fragment shader programs (a type of graphics processing task) was critical to the performance of the GPUs, seeing as they saved hundreds of thousands of computation cycles. Ex. 16, Performance Enhancement Patch at RTK-177-000004 (“It is wasteful to perform these calculations thousands to hundreds-of-thousands of times, when they produce the same result each time.”).

To address this performance issue, ARM developed the “Performance Enhancement Patch” Discussed above. *See* Ex. 12, Larri Tr. (03/19/2024) at 97:13-98:9. In the Performance Enhancement Patch Release Note that accompanied the release of this patch, Mr. Larri explains the relationship between Pilot Shaders to Vertex Shaders with reference to the below illustration.

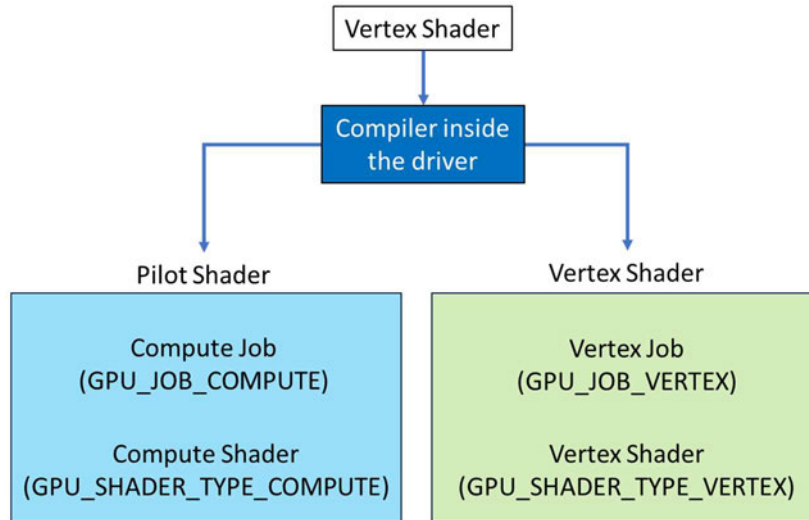


Ex. 16, Performance Enhancement Patch at RTK-177-000004.

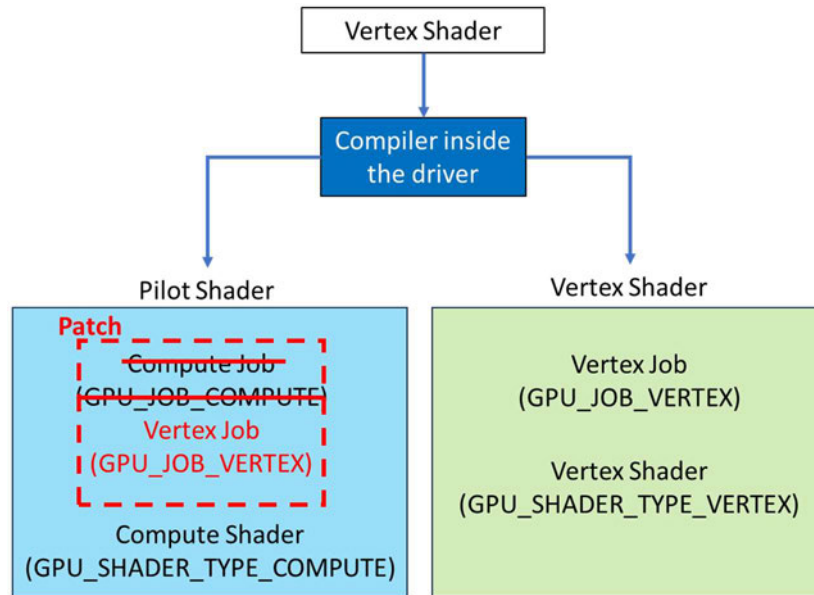
As described, the Pilot Shaders are generated by a compiler inside of ARM’s driver software to support performing generic computation that would otherwise be repeatedly executed for each vertex shader. As explained by Mr. Larri, “[t]he driver software then arranges for the Pilot shader to be run once, and the main Vertex shader to be run once for every vertex.”). *Id.*; *see also* Ex. 12, Larri Tr. (03/19/2024) at 92:10-20. Although Mr. Larri’s Performance Enhancement

Release Note attempts to frame the pilot shader programs as part of a vertex shader program, there were good reasons why the engineers who designed the ARM Mali GPU and driver thought it was more appropriate to distinguish between pilot shader programs and vertex shader programs, and to label the former as a compute shader and the other as a vertex shader. As explained by Dr. Reinman, when implemented in code, the Pilot Shaders are a distinct shader stage. Ex. 1, Reinman Decl. ¶¶99. That is, a vertex shader is setup as a separate shader stage, with a separate shader program, from the pilot shader. *Id.* In fact, even after the relabeling of the pilot shader programs to use the GPU_JOB_VERTEX label, the pilot shader programs still use a compute shader type (*i.e.*, GPU_SHADER_TYPE_COMPUTE). Ex. 1, Reinman Decl. ¶¶ 102-103. In stark contrast, a vertex job is executed using a vertex shader type (*i.e.*, GPU_SHADER_TYPE_VERTEX), which demonstrates that ARM's reconfiguration of vertex jobs to submit these pilot shader programs does not actually change the general-compute nature of the pilot shader programs and their execution. *See, id.* ¶¶ 88; 104.

As acknowledged by Mr. Larri, prior to the patch, the ARM Mali Driver used compute jobs to submit these pilot shader programs to the Mali GPU for execution by the Mali GPU's shader core. *See* Ex. 16, Performance Enhancement Patch at RTK-177-000004 ("Unpatched Mali driver software runs the pilot shader using a Compute job."). This is shown in the illustration below, which demonstrates that Pilot Shader programs were classified as compute jobs having a "GPU_JOB_COMPUTE" header and a "GPU_SHADER_TYPE_COMPUTE" shader type. In contrast, Vertex Shader jobs are classified as a vertex job with a header GPU_JOB_VERTEX and a shader type GPU_SHADER_TYPE_VERTEX.



To address the degradation of performance that was a result of applying the “Behavior Change” patches which restricted the execution of compute jobs simultaneously with graphics jobs, ARM discovered that vertex jobs could be repurposed to submit both compute tasks and graphics-processing tasks. ARM thus configured vertex jobs to submit compute tasks, in the form of the pilot shader programs, without making any change to the pilot shader programs themselves, much less the character and function of the actual computations they perform when executed by a shader core. The net effect of ARM’s “Performance Enhancement Patch” is shown in the illustration below.



Other than shifting the submission of pilot shader programs from GPU_JOB_COMPUTE to GPU_JOB_VERTEX, ARM made no change to the pilot shader programs and the actual general computing functions they perform. In fact, these pilot shader programs are still indicated in ARM's source code as corresponding to a compute shader (GPU_SHADER_TYPE_COMPUTE). *See* Ex. 1, Reinman Decl. at Section IX.B, ¶¶ 143-144. Importantly, as acknowledged by Mr. Larri, 1) the computational work that is actually performed by these pilot shader programs has not changed, 2) the binary of pilot shader programs has not changed, and 3) the only difference between the pilot shader executed as a compute Job vs. a vertex job are minor differences in the setup of the job.

Q But the net of it is that whether the pilot shader program compiled from a vertex shader program is run as a compute job or it is run as a vertex job, that the computations in the pilot shader program will be the same, correct?

A With -- **when using the Mali Bifrost GPUs and the associated drivers, the pilot shader binary will be the same regardless. Whether it's submitted with a compute job or whether it's a vertex -- or a vertex job, it will run the same program, and the program will perform the same computations.**

Q Well -- so -- okay. Let me make sure I understand that.

So will there be any additional computation that is performed if a pilot shader job is run as a vertex job versus a compute job, if it's the same program?

MR. VANHOUTAN: Objection, form.

A As I just testified, the pilot shader in the scenario we're describing will perform the same computations regardless of whether it is submitted as a compute job or as a vertex shader job.

Q Will the –

A And so there will be no difference in the computations performed in those two scenarios.

...

Q. Is there any difference in the work performed by a pilot shader program running as a vertex job versus a pilot shader program running as a compute job?

A The only difference in the work performed in the Mali drivers and in the different components of the GPUs is that **there are some small differences in the values that need to be put into the descriptors to specify a compute job rather than specifying a vertex job.**

So there will be -- in order to specify which type of job you want to run, you need to create the descriptors in a slightly different way. So apart from that difference in work, no, there will be no other difference in work.

Ex. 12, Larri Tr. (03/19/2024) at 94:18-96:15 (emphasis added).

Q. After the job setup function of the ARM Mali driver, there is no difference in the work performed by a pilot shader program running as a compute job and a pilot shader program running as a vertex job, correct?

MR. VANHOUTAN: Objection, form, asked and answered.

A Yes, with the -- with -- certainly with the -- within the Mali Bifrost GPUs and the drivers associated with the Mali Bifrost GPUs, your statement is correct.

Id. at 97:2-12.

As a result of the patches, the very same general-compute tasks that were found to infringe (*i.e.*, Pilot Shader Programs submitted for execution by compute jobs) are still submitted to the Mali GPU's shader core for simultaneous execution with graphics-processing tasks (*i.e.* Fragment Shader Programs submitted for execution as Fragment Jobs).

Realtek's trick is to bypass ARM's "helper" function referenced in the Ruling Request Letter by modifying the driver to use vertex jobs as the vehicle for submitting those same exact

compute tasks, in the form of Pilot Shader Programs, to the Mali GPU's shader core for simultaneous execution with graphics-processing tasks. Specifically, since these compute tasks, in the form of pilot shader programs, have been relabeled for submission as vertex jobs, they can bypass the ARM Behavioral Change Patches and continue to execute at the same time as fragment jobs and other vertex jobs in the shader core. This renaming is entirely insufficient to overcome the finding of infringement by the Commission, and is nothing more than an attempt to circumvent the LEO without changing the actual infringing configuration and functionality of the products at issue.

Realtek resorted to this “name game” based argument repeatedly before the ITC, and in each instance such arguments were rejected. In one example, Realtek argued that the claimed state data was not found in the Mali GPUs because “state data” only exists within the execution engines of the Mali GPUs. Ex. 28, RIB at 78-79. In another example, Realtek argued that the claimed engines could not encompass the frontends of the Mali GPUs because the “shader core” of the Mali GPUs included the frontends. *Id.* at 77-78. The ID, and the Commission, correctly rejected these arguments. Ex. 8, ID at 107, 111 (citing *Plastic Omnium Advanced Innovation and Research v. Donghee Am., Inc.*, 943 F.3d 929, 935-6 (Fed. Cir. 2019) for “emphasizing infringement analysis is between claim terms and accused product, not feature names used in defendant’s product literature.”); Ex. 9, FD at 40. The relabeling of the submission of pilot shader programs from GPU_JOB_COMPUTE to GPU_JOB_VERTEX by ARM should likewise be rejected as nothing more than an attempt to evade an actual analysis of the infringing configuration and function of the products at issue.

Moreover, the execution of general-compute tasks and graphics tasks simultaneously by the RRL Articles was confirmed by Dr. Reinman’s testing of the execution of programs on each

of Sample Boards 1 and 2 which included general-compute tasks (such as mathematical computations and physics simulations) executed as ARM vertex shader programs at the same time as graphics tasks, executed as ARM fragment shader programs. Ex. 1, Reinman Decl. at Section XII.1. What these tests clearly demonstrate is that when the claim construction for general-compute task is properly considered, the naming conventions used by ARM do not negate the fact that the Mali GPUs are still configured to execute general-compute tasks with graphics tasks. It is simply not possible for a software driver to disable this configuration, given that the GPU hardware and driver does not dictate what the intended computation of a particular set of work that is to be performed as requested by an application. Indeed, ARM's own Mr. Larri confirmed the applications can define the computations to be performed and the shader program that is used, not the driver software. Ex. 12, Larri Tr. (03/19/2024) at 48:7-22, 49:1-15 ("If someone is writing a particle physics application and they decide that the computations performed by a vertex shader are useful for their application, then that use is defined by the application code.")

3. The EOE Branch Should Deny Any Attempt By Realtek to Seek a Ruling Based on Further Modification to the Ruling Request Letter Articles or Argument of Non-Infringement of the Performance Enhancement Patch

To the extent Realtek's reply submission claims yet further modifications to the RRL Articles (by way of modifying the RRL Articles to modify or remove the Performance Enhancement Patch), or submits any factual evidence or expert testimony in support of their position that the Performance Enhancement Patch does not infringe Claim 19, the EOE Branch should still deny Realtek's request because it did not meet its burden of proof and production in this proceeding. *See* Ex. 1, Reinman Decl. at Section X.

Realtek could have and should have addressed the Performance Enhancement Patch in their Ruling Request Letter, and the supporting declarations and evidence submitted with the Ruling

Request Letter, but deliberately chose not to. Worse, since filing its incomplete Ruling Request Letter, Realtek has only doubled-down on its strategy of obstructing and obfuscating relevant discovery in an effort to prevent AMD's counsel from developing a complete factual record concerning Realtek's claims about the articles subject to its Ruling Request Letter. Realtek's discovery misconduct in this proceeding has been pervasive, but specifically related to the Performance Enhancement Patch, has included at least the following:

- **Withheld Testing:** Realtek and ARM conducted testing of articles subject to the remedial orders, as well as various versions of the Patched Mali Driver software, both with and without the Performance Enhancement Patch. However, Realtek and ARM have improperly withheld this testing, and any testimony concerning the testing, as privileged. See Sections IX.A.2 *infra*.
- **Withheld ARM Internal Technical Documentation:** AMD requested, but Realtek did not obtain and furnish ARM's internal technical documentation relevant to the issues implicated by Realtek's Patched Mali Driver. This includes ARM's internal technical documents related to the structure, function, and operation of the driver software/Patched Mali Driver software, compiler software, the use of Vertex/Fragment Jobs to submit compute tasks, and pilot shader programs. See Section X.A *infra*.
- **Withheld Realtek Fact Witness Testimony:** Although Realtek fact witnesses were involved in the development of the RRL Articles subject to its Ruling Request Letter, and various testing activities related to those articles, Realtek has refused to allow AMD's counsel to take the deposition of any Realtek fact witnesses. See Section X.A.1 *infra*.

- **Realtek’s Experts:** In addition to not disclosing any opinion concerning the Performance Enhancement Patch in the declarations attached to Realtek’s Ruling Request Letter, Realtek’s counsel appears to have decided not to show the Performance Enhancement Patch code to Realtek’s experts, Mr. dela Iglesia and Dr. Pfister. As a result, AMD’s counsel was not given the opportunity to cross-examine Realtek’s experts concerning the substance of any opinion they may now offer on the Performance Enhancement Patch. Section X.A.3 *infra*.

At this point, if Realtek were to submit a reply brief in which it proposes yet further modifications to the RRL Articles, proposes to remove the Performance Enhancement Patch from the RRL Articles, or submits new fact or expert witness declarations addressing these currently undisclosed further modifications or argument regarding what the Performance Enhancement Patch does, it will be the first time Realtek is revealing its positions to the EOE Branch and AMD. Not only would a one-week sur-reply period be insufficient for AMD to consider and respond to whatever position Realtek decides to reveal for the first time in its reply submission, AMD’s ability to respond will be impaired by Realtek’s deliberate and strategic decision to prevent AMD from obtaining at least the categories of discovery set forth above and developing a complete factual and expert record.

B. Claim 20: The “New Mali Driver” Patches Simply Create New Prioritization Schemes or Shift the Priority Considered by the System Without Eliminating Consideration of Priority that Infringe Claim 20

Just as Realtek’s new Mali driver patches have not disabled the RRL Articles’ infringement of Claim 19, they have also not disabled those articles’ infringement of Claim 20, either.

Prioritization of Jobs, and their constituent tasks and warps, is deeply embedded in the configuration of the Mali GPUs through a hierarchy of prioritization mechanisms. First, as acknowledged by Mr. Larri, application developers can incorporate prioritization of work outside

of the Mali Driver through the application software, thereby dictating how the units of work (*i.e.*, claimed tasks) are ultimately received by the GPU and its corresponding execution resources (*i.e.*, claimed shader core). Ex. 12, Larri Tr. (03/19/2024) at 199:12-19. Second, the patches only partially attempt to address two portions of the prioritization hierarchy within the Mali Driver, while leaving unchanged the very function [REDACTED] that was identified by Dr. Reinman at the hearing and cited by the ID. Ex. 1, Reinman Decl. ¶¶ 156-182. Third, in its attempt to overcome the finding of infringement of Claim 19, Realtek has introduced new prioritization mechanisms which allow jobs to gain exclusive access to the GPU's resources. *Id.* ¶¶ 150-155. Fourth, the patches to the Mali Driver make no change to the prioritization of work by the Mali GPU's hardware. *Id.* ¶¶ 172-182.

The net effect of Realtek's attempts to apply patches to remove prioritization in the Mali Driver simply shifts the prioritization of jobs, and their associated work, to other parts of the hierarchy. This is precisely why Realtek's own expert, Dr. Pfister, could not support Realtek's position that all priorities have been limited from the Standalone Realtek ICs.

Q. The question was: Do you agree with the statement that there is no longer any priority anywhere in the Mali GPU that can even arguably satisfy Claim 20?

MR. HIGGINS: Object to form.

A. No, I can't say that.

Ex. 14, Pfister Tr. (03/25/2024) at 248:2-7.

At bottom, with the Patched Mali Driver or without it, prioritization of Jobs, and their constituent tasks and warps, is pervasive throughout the configuration of the driver software and the Mali GPU itself. Even with Realtek's Patched Mali Driver, the shader core in the Mali GPU prioritizes received tasks for at least six (6) independent reasons set forth below.

**1. Application Programmers May Implement Prioritization in
Application Software to Circumvent the ARM Patches**

The patches to the Mali Driver do nothing to prevent application programmers from including prioritization of work in the application software, which in turn results in the GPU and its shader core receiving the corresponding work based on the software application based prioritization. Recognizing this flaw in its request, Realtek has refused to provide the relevant source code for current versions of application software that would actually be used with the Patched Mali Driver. This is despite ARM's own corporate representative, Mr. Larri, agreeing that in order to determine whether prioritization is implemented in application software, one would need to review the source code for such applications.

Q And the driver doesn't prevent applications from prioritizing the submission of shader programs for processing on the GPU's execution engine, correct?

A Correct. **Application programmers are free to write application programs as they wish**, as long as they conform with the Khronos-standard APIs when they make requests into the driver.

Ex. 12, Larri Tr. (03/19/2024) at 199:12-19.

Q You would need to look at the application source code to figure out if there's any application-based prioritization going on that causes the shader core to receive tasks based on a priority?

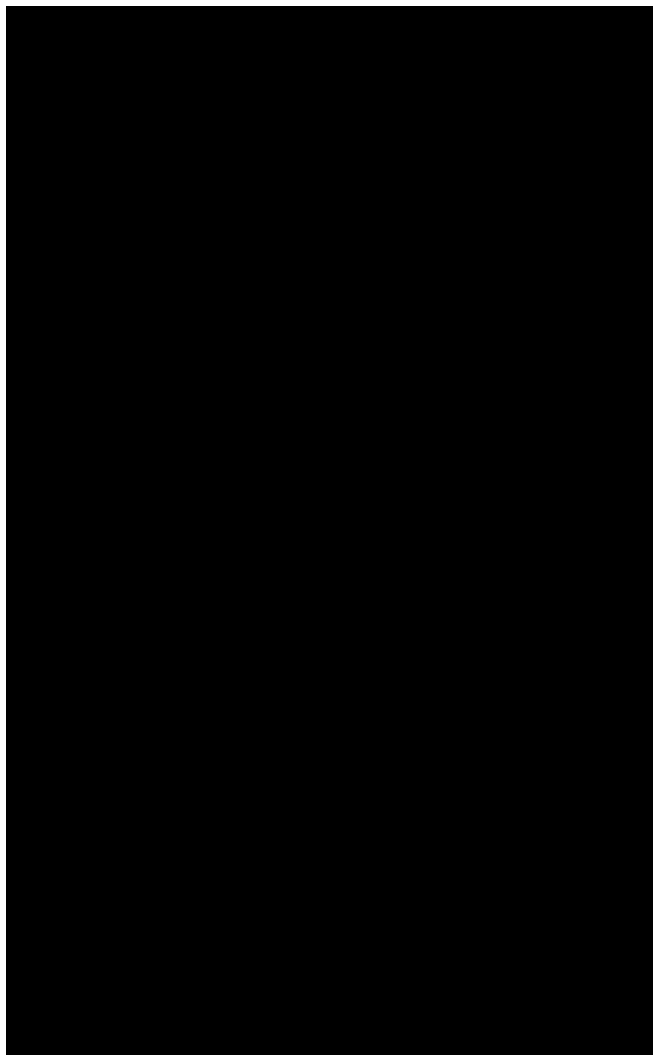
A If you're understanding that the scope of the priority question extends beyond the GPU driver and the GPU hardware out into applications themselves, **then you would have to understand the operation of the applications themselves.**

Id. at 199:2-11. Without the production of the source code for these applications, Realtek cannot possibly meet its burden to show that its redesign overcomes infringement. Moreover, the use of application based prioritization is not only likely, Dr. Reinman's testing of Realtek's sample boards shows that the execution engines (*i.e.*, shader core) of the very sample development boards Realtek introduced in this proceeding do receive tasks and execute the corresponding work based on priorities associated with those tasks by the application software. *See* Ex. 1, Reinman Decl. at

Section IX.3 (demonstrating prioritization through the use of memory barriers in the application software for compute jobs).

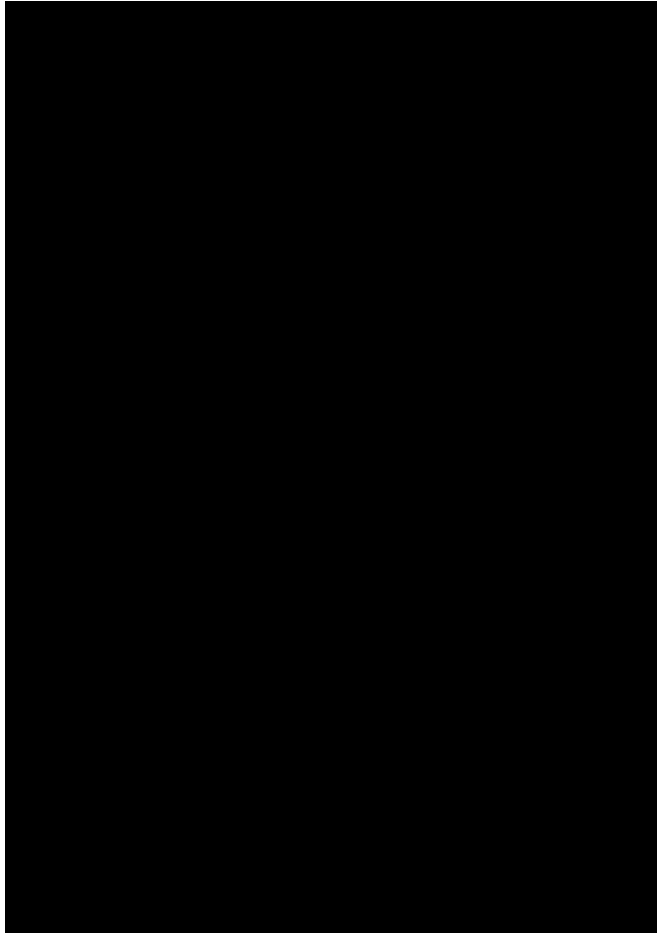
2. The Patches Only Partially Address the Priorities Considered in the Hierarchy of Prioritization of the Mali Driver and Mali GPUs

As Realtek's own expert Dr. Pfister conceded, the patches applied by Realtek did not in fact remove all priorities from the Mali GPUs. Ex. 14, Pfister Tr. (03/25/2024) at 248:2-7. Rather, the patches only partially apply to two layers of the deeply embedded multi-layer prioritization hierarchy of the Mali Driver and Mali GPU hardware. This prioritization hierarchy within the Mali Driver and Mali GPUs, and the relevance of the patches, can be summarized using the illustration below.



As illustrated, the ARM design, which Realtek incorporates, considers prioritization in at least three hierarchical portions of the architecture as jobs propagate through the driver to the GPU's execution resources (*i.e.*, the claimed shader core). Ex. 1, Reinman Decl. ¶¶ 156-182. First, prioritization is considered at the Mali Driver through function [REDACTED]. This function considers settings for Job Slot Priority, Job Priority, Context Priority, and Age or Sequence Priority in order to prioritize how jobs are dispatched to the Job Manager. *Id.* ¶¶ 159-171. Second, prioritization is considered at the Job Manager within ARM Mali GPUs (*e.g.*, [REDACTED]) in determining how Jobs, and their corresponding tasks are passed to the [REDACTED] [REDACTED] *Id.* ¶¶ 172-175. Third, prioritization is considered at the [REDACTED] [REDACTED] within the ARM Mali GPUs in issuing warps, which are the executable portions of the Jobs, to the execution resources. *Id.* ¶¶ 176-183.

Contrary to Realtek's assertions, the patches did not eliminate consideration of prioritization of jobs and their associated work (*i.e.*, in the form of tasks and warps). Rather, the patches only address two discrete mechanisms in the Mali Driver; prioritization of Job Slot 2 over Job Slot 1 (an example of Job Slot Priority) and priority settings for job chains by setting all job [REDACTED] to the same default priority level of medium priority (an example of Job Priority). By attempting to remove these two discrete priority considerations, ARM's patches simply shifted the prioritization considered to other parts of the hierarchy. Specifically, as shown below, the Mali Driver, and the scheduling function [REDACTED] still considers context priority as well as age or sequence priority when scheduling jobs to the Job Manager. Ex. 1, Reinman Decl. ¶¶ 159-171.



3. The Patched Mali Driver Has Shifted Job Slot Prioritization to Job Slot 0 Over Job Slot 1 and Job Slot 2

ARM's patches disable the Mali Driver from scheduling jobs to [REDACTED] thereby shifting prioritization of [REDACTED]. This in turn prioritizes [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED], as explained by Dr. Reinman, the Patched Mali Driver still prioritizes jobs based on their [REDACTED]. *See* Ex. 1, Reinman Decl. ¶¶ 173-175.

Specifically, the Patched Mali Driver prioritizes jobs that originate from Job Slot 0 over those that originate from Job Slot 1 through a function named `ffs()`, which stands for “find first set.” This function identifies the first bit set to 1 in a sequence of bits, each representing a [REDACTED].

The [REDACTED] are checked in ascending order, starting from [REDACTED], to find where jobs are pending. By applying ffs() to a bitmask [REDACTED], where each bit corresponds to a [REDACTED] with pending jobs, the scheduler can identify the lowest-numbered [REDACTED] to which jobs can be scheduled. *Id.*

ARM's corporate representative, Mr. Larri, conceded that even he agrees this is a form of prioritization that remains in the New Mali Driver during his deposition.

Q Okay. So in that case, what's in [REDACTED] will be prioritized over [REDACTED] correct?

MR. VANHOUTAN: Objection, form, misstates his testimony.

A Well, if that – it will process the one that's in – for [REDACTED] first, in that scenario.

Q So the answer is, yes, it will prioritize what's in [REDACTED] correct?

MR. VANHOUTAN: Same objection.

A If the scenario which you described is possible, and if the code behaves as you've represented, then it would prioritize [REDACTED] over [REDACTED] in that limited scenario.

Q In the patched driver code, correct?

A This – yes, we're looking at the patched driver code, yes.

Ex. 12, Larri Tr. (03/19/2024) 231:19-232:12.

Moreover, the use of a single [REDACTED]
[REDACTED] that are scheduled to
the Mali GPUs. That is, in the Unpatched Mali Driver, [REDACTED] was one of three [REDACTED]
contending for GPU resource allocation of its [REDACTED]. As the Patched Mali Driver allegedly disables
[REDACTED] is now one of only two remaining [REDACTED] that are enabled by the Patched
Mali Driver. As a result, fragment jobs have greater access to the GPU resources, and are
prioritized, through this change to the scheduling mechanism of the Patched Mali Driver. Ex. 1,
Reinman Decl. ¶ 153.

4. The Patches Have Not Disabled the Consideration of Priority by the Function Identified by AMD and Cited by the Commission () Which Prioritizes Jobs Based on Age or Sequence Number

As explained by Dr. Reinman, in setting the priority of all () to a default priority (), the ARM patch set the priority level to a medium priority level. *See* Ex. 1, Reinman Decl. at Section VII.C and VIII.D.3 This patch, however, did nothing to the function () and code that considers the priority assigned to each job. This is the very function that Dr. Reinman identified during the evidentiary hearing, and which the ID expressly cited in its decision. *See* Ex. 11, Hrg. Tr. (Reinman) at 176:25-179:5; Ex. 8, ID at 124 (citing CIB at 105-106).

As Dr. Reinman explains, the function () still considers the priority for each job by iterating through the high priority, medium priority, and low priority job queues in determining which job should be scheduled next. The fact that all jobs are set to the default state of medium priority does not negate that the shader core is still configured to receive the jobs based on their associated medium priority setting. *See* Ex. 1, Reinman Decl. ¶ 145. And nothing in Claim 20 requires that each of the plurality of jobs include a different priority.

Moreover, with the default setting of medium priority for all jobs, this same () function selects jobs based on their age and/or sequence number. *See* Ex. 1, Reinman Decl. at Section VIII.D.3.ii. During the ITC Investigation, ALJ Elliot, in connection with his claim construction of the term “relative priorities” for Claim 1 of AMD’s U.S. Patent No. 7,742,053 determined that age is in fact a form of priority. Ex. 10, Order No. 31 at 16-17; *see also* Ex. 33, U.S. Patent No. 10,186,069. As explained by Dr. Reinman, the Mali Driver, with the patches applied, maintains a prioritization mechanism that orders and reorders jobs by job age in a binary search tree which is referenced in the source code as a “Red/Black” tree. Ex. 1, Reinman

Decl. ¶¶ 147-149. (citing, *e.g.*, function [REDACTED] which returns the earliest sequence number in a runnable tree.). [REDACTED] are inserted into the binary search tree and are ordered and reordered based on a sequence number that is associated with the age of the [REDACTED]. *Id.* As a result of the setting of all jobs to the default medium priority, the Patched Mali Driver has simply emphasized this form of prioritization in the driver software, which in turn results in the receipt of the work associated with these jobs by the execution resources of the GPU based on their age or sequence number priority.

5. The Patched Mali Driver Prioritizes Jobs Based on Their Associated Context

Even if jobs are set to the same default priority, in addition to their prioritization based on age, the Patched Mali Driver prioritizes jobs based on their associated contexts. A “context” in GPU computing includes the data for a GPU to execute tasks. Ex. 1, Reinman Decl. ¶¶ 159-169. Each context can have multiple jobs queued for execution. As explained by Dr. Reinman, the Patched Mali Driver will select jobs for execution from sets of job contexts that are in an ordered “head” to “tail” list. Each of these contexts has an associated job tree that the Patched Mali Driver selects jobs from. The contexts, including their [REDACTED] are reordered by the Patched Mali Driver. Additionally, the Patched Mali Driver includes specific functions to “fast-start” jobs that are associated with an “active” context. *Id.* ¶ 167. Through prioritization of jobs by job context, the Mali Driver in turn prioritizes jobs for dispatch to the GPU and its execution resources (*i.e.*, claimed shader core).

6. The GPU Hardware Includes Multiple Mechanisms to Prioritize Work Associated with Jobs for Receipt by the Execution Resources

The ARM Mali GPUs include multiple hardware based configurations that prioritize work associated with jobs. These hardware based configurations cannot be modified by modification to the Mali Driver software. First, the Mali GPUs include a hardware configuration that prioritizes

[REDACTED]

[REDACTED]. Ex. 1, Reinman Decl. ¶¶ 176-179. This same configuration in hardware was expressly relied on by the ID in concluding that the ARM Mali GPUs, and the products that incorporate them, infringe Claim 15. Ex. 8, ID at 118 (“In fact ARM documentation states the [REDACTED]

[REDACTED]”).... Thus, the “configured to” limitation is met.”) (citing ARM’s Internal Document “[REDACTED],” CX-1730C at 11, 39). Since the patches do nothing to modify the hardware of the Mali GPUs, this prioritization of jobs in the GPU hardware remains unchanged.

As explained in ARM’s internal technical documents, the [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

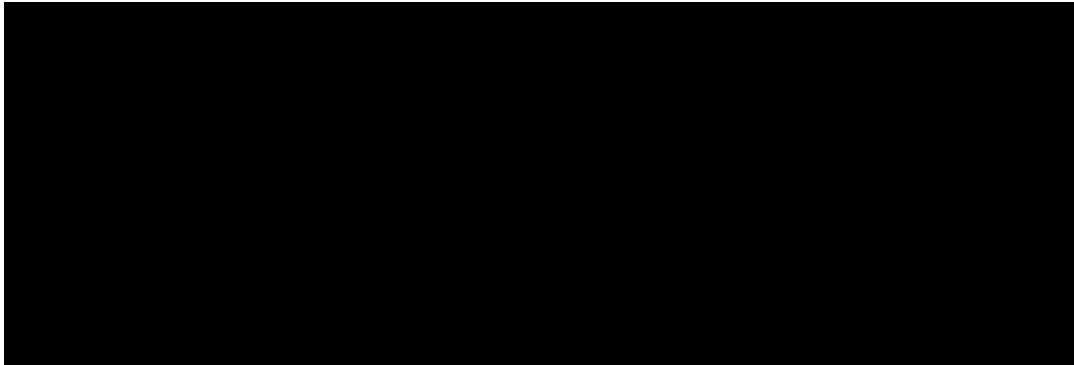
[REDACTED]

[REDACTED]

[REDACTED]

Ex. 43, CX-1730C ([REDACTED]) at 39. As explained by Dr. Reinman in connection with the source code excerpt reproduced below, this configuration is implemented in hardware

through a signal “[REDACTED]” setting in the hardware register. Ex. 1, Reinman Decl. at ¶¶ 172-179.



This fairness based prioritization in hardware is even more relevant now that the Patched Mali Driver attempts to restrict the processing of fragment jobs when [REDACTED] [REDACTED] are being executed by the GPU by serving to [REDACTED] [REDACTED] mechanisms imposed through the patches introduced by ARM.

Second, the GPU hardware [REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED] which is the hardware components expressly relied upon by the Commission in finding that the Mali GPUs were configured to execute tasks simultaneously, prioritizes work that is received by the execution engines of the Mali GPUs for execution.

Realtek has elected to ignore each of these hardware based prioritization mechanisms since it could not modify or change its implementation through a software based patch to the separately provided ARM Mali Driver.

C. Claim 20: The ARM Behavioral Change Patches to Avoid Claim 19 Introduces Yet an Additional Priority Scheme That Infringes Claim 20

In addition to the prioritization discussed in Section IX.B., *supra*, Realtek's implementation of ARM's "Behavioral Change" patches to avoid Claim 19 introduces a new priority scheme that infringes Claim 20. Ex. 1, Reinman Decl. at Section XI.D.2. The reason for this is simple. Realtek's approach for trying to avoid Claim 19 and its approach for trying to avoid Claim 20 are fundamentally orthogonal to one another: to the extent Realtek's Patched Mali Driver prevents the submission of certain⁵ general-compute tasks to the Mali GPU's shader core while the Mali GPU's shader core is executing graphics processing tasks; the net result is that the Patched Mali Driver will prioritize the submission of task types that are currently in execution by the shader

⁵ As noted in Section IX.A., *supra*, the patched driver software is at least specifically designed to submit certain general-compute tasks, in the form of [REDACTED] programs, to the shader core for simultaneous execution with graphics processing tasks, in the form of fragment shader programs.

core. *See id.* If a shader core is executing graphics processing tasks, the Patched Mali Driver will prioritize the submission of graphics processing tasks to the shader core. *Id.* And if the shader core is executing compute tasks, the Patched Mali Driver will prioritize the submission of compute tasks to the shader core. *Id.* The shader core will, of course, receive those tasks based on the Patched Mali Driver’s built-in prioritization.

1. Graphics Jobs Are Prioritized When a Graphics Job is Executing and Jobs Marked as Compute Jobs Are Prioritized When a Job Marked as Compute is Executing

First, ARM’s “Behavioral Change” patches prioritize jobs marked by the “helper” function as compute jobs when a [REDACTED] that includes at least one compute job is executing while deprioritizing graphics jobs (*e.g.*, fragment jobs) from executing. **Second**, the “Behavioral Change” patches prioritize graphics jobs (*e.g.*, fragment jobs and jobs marked as vertex jobs) when a graphics job chain is executing while deprioritizing jobs marked by the “helper” function as compute jobs. This reprioritization is implemented without any change to the hardware configuration of the Mali GPUs, and therefore, the execution engines of the Mali GPUs (*i.e.*, the claimed shader core) remains configured to at least execute graphics tasks (*e.g.*, fragment jobs and vertex jobs) at the same time to satisfy the requirements of Claim 15, while allowing at least some tasks of the plurality of tasks to temporarily consume all the execution resources of the Mali GPUs. This type of prioritization is expressly disclosed as an embodiment in the ’381 Patent, which discloses that “resources (*e.g.*, SIMDs) of the GPU shader core can be divided between concurrent tasks **based on demand, priority, and/or preset limits-while temporarily enabling any one task to (substantially) fully consume the resources of the GPU.**” Ex. 2, ’381 Patent at 5:34-39 (emphasis added); *see also, id.* at 10:47-51

By allowing certain jobs to consume all of the execution resources temporarily, these jobs and their associated work can “fully consumer the resources of the GPU,” and thus complete faster

than when sharing resources with other jobs. ARM's corporate representative, Mr. Larri, suggested that it would be "possible" for a job to finish faster when its not sharing resources, but that one would have to run a workload to make the determination.

Q So in the scenario that I described where fragment jobs are executing and a compute job comes in but it's blocked because there's a fragment job that's running, the fragment jobs would finish faster and finish executing faster in that scenario where they're blocked as compared to the scenario where they have to share resources with the compute job in the unpatched driver, correct?

A It's possible that, in that scenario you describe, the fragment job might finish faster because it's not sharing resources on the shader core with another type of job.

...

Q So what would you need to do – what would you need to look at to answer that question? Would you need to look at application use cases?

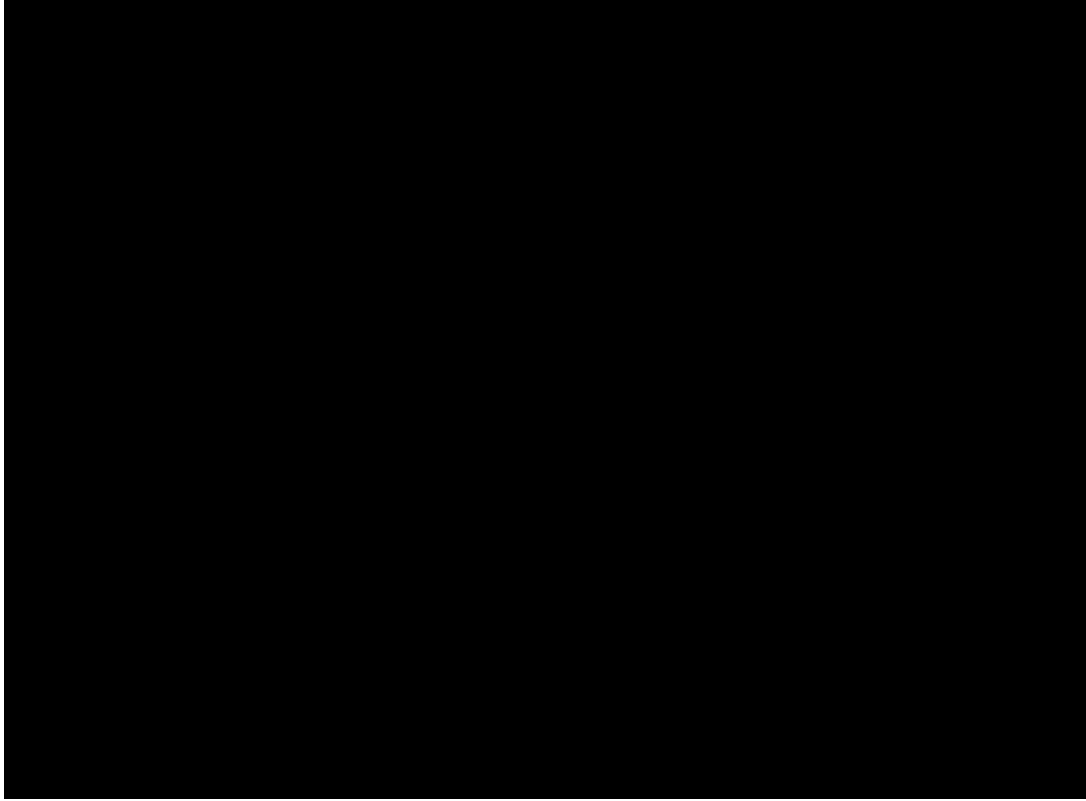
A It's such a complicated scenario that you would have to -- you'd have to run a particular workload which rendered a particular scene to see whether the performance was different. I don't think it's predictable in any other way than actually running the workload.

Ex. 12, Larri Tr. (03/19/2024) at 195:15-196:8. Dr. Reinman confirmed, through testing of such a workload, that the Mali GPUs do indeed complete jobs faster when those jobs are able to consume all resources of the Mali GPUs without sharing resources with work from other jobs. Ex. 1, Reinman Decl. at Sections VIII.D.2, IX.2.

This new form of prioritization, introduced by the "Behavioral Change" patches in a (failed) attempt to avoid Claim 19, is demonstrated using the example scenario below. As shown in the example scenario, when a fragment job (orange circle) is executing in the shader core, subsequently arriving jobs that are marked by the "helper" function as compute jobs (blue cross) are blocked by the Patched Job Scheduler, while arriving fragment jobs are scheduled for execution through [REDACTED] and dispatched to the shader core based on this prioritization in the Patched Job Scheduler.



Similarly, when tasks submitted for a job that is marked as a compute job by the “helper” function is executing, tasks submitted for jobs that are marked as non-compute jobs (*e.g.*, graphics jobs) are deprioritized until tasks submitted for the job marked as a compute job complete execution. This prioritization based on what the “helper” function marks as compute jobs does not impact the configuration of the Mali GPUs to execute tasks submitted for fragment jobs and what the “helper” function deems non-compute (*e.g.*, tasks submitted for vertex jobs) simultaneously as demonstrated in the illustration below.



The simultaneous execution of tasks submitted for fragment and vertex jobs, which was found to infringe claim 15, was also confirmed by Dr. Reinman in tests performed on Sample Board 1 and Sample Board 2. Ex. 1, Reinman Decl. at Section IX.1.

What these scenarios, and Dr. Reinman's tests demonstrate, is exactly the type of prioritization described in the '381 Patent—the driver enabling a task to fully consume all resources of the GPU. This new prioritization mechanism, which has been introduced through ARM's patches in this proceeding, infringes the combination of Claim 15 and 20, since the Mali GPUs are still configured to execute tasks submitted for at least two jobs (*i.e.*, a first task and a second task) marked as non-compute by the “helper” function at the same time, while the shader core is configured to receive the work associated with all jobs (*i.e.*, the plurality of tasks) based on the priorities associated with all jobs as a result of the changes implemented by the Behavioral Change patches.

X. REALTEK HAS FAILED TO SATISFY ITS BURDEN OF PROOF GIVEN ITS DEFICIENT EVIDENTIARY PRODUCTION

For the reasons set forth in Sections VII and VIII, the EOE Branch should deny Realtek’s Ruling Request Letter because the Patched Mali Driver software does not affect the configuration of the Mali GPU found to infringe, and in any event, does not work as Realtek claimed in its Ruling Request Letter.

But even if Realtek could prove that the changes it made to the Patched Mali Driver software somehow modify hardware configuration of the Mali GPU found to infringe (it does not), **and** even if Realtek could prove the Patched Mali Driver could plausibly operate as Realtek described in its Ruling Request Letter (it could not), the EOE Branch should still deny Realtek’s request because Realtek deprived AMD and the EOE Branch of their ability to develop a complete record of fact and expert testimony that Realtek needed to meet its burden of proof. Realtek’s failure to meet its burden of proof and production in this proceeding is fatal to Realtek’s Ruling Request Letter, and provides independent grounds for the EOE Branch to deny Realtek’s request.

The burden is on Realtek to demonstrate the products it intends on importing are non-infringing. *See Hyundai Elecs. Indus. Co. v. Int’l Trade Comm’n*, 899 F.2d 1204, 1210 (Fed. Cir. 1990) (explaining that, where an exclusion order issues, the burden of establishing non-infringement “effectively shifts to would-be importers[.]”); Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1191; *Certain Audio Players and Controllers, Components Thereof, And Products Containing the Same*, 2022 U.S. CUSTOM HQ LEXIS 45, *3 (June 3, 2022). But from the onset of this proceeding Realtek has implemented a strategy of obfuscating and withholding relevant evidence and documentation—both evidence that the EOE Branch would need to determine that the RRL Articles do not infringe Claims 19 and 20, and evidence that AMD needs to rebut many of Realtek’s claims.

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

For example, despite AMD's repeated requests for discovery, Realtek's many failures to produce pertinent and relevant source code and documentation include: 1) Realtek failed to produce RTL code for the "new" RTD2951M chip that is used in its Sample Board 2 and Sample Board 3 (*see* Ex. 38, Christopher Higgins Annotated Email (03/08/2024) at 1-4); 2) Realtek failed to produce documentation that specifies the testing, including testing protocols, that was performed for the Patched Mali Driver and the results of that testing (*see* Larri Tr. (03/19/2024) at 99:10-100:7); 3) Realtek's experts did not review the Patched Mali Driver as actually applied (*see* Ex., 13, Iglesia Tr. (03/22/2024) at 15:5-12; Ex. 14, Pfister Tr.(03/25/2024) at 96: 18-97:12); 4) Realtek refused to permit AMD to ask Mr. Larri questions based on the source code for the driver that is used with the Valhall Architecture (Natt G57) GPU (*see* Ex. 12, Larri Tr. (03/19/2024) at 113:9-114:17, 115:4-10); 5) Realtek refused to produce communications between Realtek and ARM related to the performance enhancement patch or a privilege log for any of the documents where Realtek and ARM asserted privilege(*see* Ex. 12, Larri Tr. (03/19/2024) at 121:16-21); 6) Realtek failed to produce the preceding release versions (*e.g.*, dated 10/3/23 of the patch release notes (*see* Ex. 12, Larri Tr. (03/19/2024) at 121:22-122:22; 152:17-153:5; 155:11-156:4); 7) Realtek failed to produce internal engineering documents that describe the structure and function of the arm code directed to pilot shader programs (*see* Ex. 12, Larri Tr. (03/19/2024) at 127:5-128:13); and 8) Realtek failed to produce source code for the operating system and applications that would be used with and that could access the GPU in an imported product (*see* Ex. 12, Larri Tr. (03/19/2024) at 139:20-140:17; 198:18-199:17).

As explained below, not only has Realtek failed to satisfy its burden because of its strategy to withhold relevant discovery, but its strategy has deliberately prejudiced AMD in preparing its

response. As such, AMD submits that all inferences should be drawn in AMD's favor in connection with how the relevant documents and source code that Realtek elected to withhold.

A. The Declarations of Mr. Larri, Mr. de la Iglesia, and Dr. Pfister Fail to Substantiate Realtek's Claims

1. Realtek Failed to Produce a Witness That Can Authenticate How the Source Code is Actually Implemented in Realtek Products

In support of its request, Realtek submitted declarations and presented for deposition ARM's corporate representative, Mr. Guy Larri, as the only fact witness. Despite AMD's requests, Realtek refused to make available a Realtek witness for depositions during this proceeding. Ex. 37, Robert Benson Annotated Email (02/16/2024). As a result, Realtek did not present a single fact witness in this 177 Proceeding that could explain how Realtek actually designs its products, how Realtek integrates the patches to the driver in its products, and how Realtek's customers would integrate the Realtek ICs, including the Patched Mali Driver, into downstream products following importation into the United States.

As Mr. Larri conceded, ARM has no control over how customers, including Realtek and Realtek's own downstream customers, actually use any particular driver version or patches that ARM provides, and has not even looked at any Realtek code.

Q. Well, I'm not asking if anyone communicated to you; I'm just asking you if you've seen any Realtek code, if you've literally looked at Realtek code.

A No, I have not looked at any Realtek.

Ex. 12, Larri Tr. (03/19/2024) at 159:16-20; *see also Id.* at 143:17-144:17. And as Mr. Larri previously admitted, he has no knowledge of whether any customer uses the r26p0 version of the Mali Driver that Realtek is relying on as the representative driver for the patches.

Q And that module is part of the kbase driver for the Bifrost GPUs used in this case?

MR. VanHOUTAN: Objection. Form.

CONTAINS CONFIDENTIAL BUSINESS INFORMATION – AND CBI SOURCE
CODE - SUBJECT TO NDA – NOT SUBJECT TO FOIA

THE WITNESS: This is a portion of the source code for the Bifrost driver at Revision r26p0 that ARM makes available to its customers for use with Bifrost GPUs. I have no way of knowing if any particular customer has used this version of the driver.

Id. at 174:5-12.

The declarations of Mr. de la Iglesia and Dr. Pfister, and their testimony, fail to bridge this gap in Realtek's case. In both cases, Dr. Pfister and Mr. de la Iglesia simply reviewed what they were given, and did not conduct an independent investigation as to whether the products continue to infringe the claims of the '381 Patent. This is evident by the fact that neither Dr. Pfister nor Mr. de la Iglesia mention or cite to the Performance Enhancement Patch or its corresponding Release Note in their declarations. *See generally*, Ex. 5, Pfister Decl.; Ex. 4, Iglesia Decl. In fact, neither Mr. de la Iglesia nor Dr. Pfister reviewed Realtek source code with the patches applied, so they would have not had any way of knowing whether there were additional patches, such as the Performance Enhancement Patch, beyond what was provided to them for review. Ex. 13, Iglesia Tr. (03/22/2024) at 9:6-10:9, 29:1-11; Ex. 14, Pfister Tr. (03/25/2024) at 118:23-119:3. Instead, Realtek apparently relies on Mr. de la Iglesia "applying" the patches to the old version of the driver in his head. Ex. 13, Iglesia Tr. (03/22/2024) at 14:25-16:5. AMD cannot take Realtek at their word, and neither should CBP, especially in light of Realtek's attempt to conceal its covert reconfiguration of vertex jobs to submit compute tasks in the form of pilot pilot shader programs without any change to the substance or function of the actual computations and shader programs that are executed. *See supra*, Section VIII.A. Since both experts only considered what they were provided, and were apparently only provided with a selection of patch files that excluded at least the Performance Enhancement Patch, they cannot possibly be relied on by Realtek to substantiate what code is implemented in the Realtek products at issue in this investigation.

As such, and as noted in AMD's previous submissions, Realtek's "redesign" is hypothetical at best. Just as in Sony, "the evidence provided by [Realtek] not only does not match the scope of the products at issue (*i.e.*, the products that [Realtek] indicates it will import), but also is not sufficient to allow [the EOE] to determine with certainty, after-the-fact, whether the application of the [Realtek new driver software] would have eliminated all potentially infringing articles." Ruling Request; U.S. International Trade Commission; Limited Exclusion Order; Investigation No. 337-TA-1012, *Certain Magnetic Data Storage Tapes and Cartridges Containing the Same*, 2018 U.S. CUSTOM HQ LEXIS 354, *57 (August 9, 2018))

2. Mr. Larri's Declaration, Release Notes, and Testimony Should be Viewed as a Self-Serving Extension of Realtek's Attorney Arguments

Realtek's Ruling Request Letter relies heavily on the testimony of ARM's witness, Mr. Guy Larri. But Mr. Larri is not an ordinary fact witness. In his role at ARM, which he has held for several years, he is employed by and reports directly to ARM's head of patent litigation, Mr. Philip Price. Ex. 12, Larri Tr. (03/19/2024) at 17:9-14, 18:1-10. Mr. Larri was the "architect" and the "product manager" of the effort to develop the ARM patches in response to the Commission's exclusion order. *Id.* at 16:15-17. Yet despite this role, when asked basic questions about how ARM worked with Realtek to test and implement the patches, he was repeatedly instructed by ARM's outside counsel not to answer on the grounds of privilege. *See, e.g., Id.* at 159:22-160:6, 211:9-22, 124:17-125:122. Despite its requests, AMD has yet to receive a privilege log of the testing documents that Mr. Larri was instructed not to answer questions about during his deposition.

Mr. Larri, as part of the patent litigation team at ARM, authored each of the release notes that correspond to the patches in this proceeding. These release notes include the Performance Enhancement Release Note, which is dated the same day (October 13th, 2013) as the other Behavioral Change Release Notes Mr. Larri authored. Yet Mr. Larri either deliberately excluded,

or was instructed by Realtek and ARM to exclude, any and all reference to the Performance Enhancement Release Note from his declaration. When questioned about why there was no mention of the Performance Enhancement Release Note that he authored, he was instructed repeatedly not to answer, again on account of privilege.

Q. But that performance enhancement patch isn't mentioned anywhere in your declaration, correct?

MR. VANHOUTAN: Objection, form, misstates testimony, compound.

A It's correct that the declaration only describes the behavior change patches and doesn't --does not describe performance enhancement patches.

Q Okay. Why didn't you describe that patch in your declaration?

MR. VANHOUTAN: Objection, form, calls for attorney-client communication. I instruct the witness not to answer.

Q Are you going to take your counsel's instruction?

A Yes.

Q Are there any other patches that ARM implemented related to this litigation that you don't mention in your declaration?

MR. VANHOUTAN: Same objection, same instruction, and outside the scope.

Q Are you able to answer the question?

A I'm going to --

MR. VANHOUTAN: The instruction is, you're not to answer the question.

A I'm going to take my attorney's instruction.

Ex. 12, Larri Tr. (03/19/2024) at 118:10-119:14. Mr. Larri was repeatedly instructed not to answer questions on account of privilege, including his review of any Realtek testing of the patches and whether Realtek requested any documents describing the pilot shader from ARM. *Id.* at 211:9-212:11, 124:17-125:1.

Realtek and ARM's counsel interference with Mr. Larri's testimony during his deposition went beyond just instructing him not to answer questions about relevant and unprivileged facts. At his deposition, Mr. Larri originally testified that his declaration was not accurate insofar as it

suggests that the Patched Mali Driver prevents the shader core from receiving tasks based on a priority associated with the tasks. Ex. 12, Larri Tr. (03/19/2024) at 232:22-233:13. Following a break in testimony, Mr. Larri returned with a “correction” to retract the testimony he had provided right before the break. *Id.* at 249:13-250:18. When asked basic questions about whether he conferred with counsel regarding his testimony during the break, Mr. Larri was instructed not to answer, again on account of privilege.

Q Okay. Have you conferred with anybody about the substance of the testimony that you just provided?

MR. VANHOUTAN: Objection, form, calls for attorney-client communication. I instruct you not to answer.

A I'm taking my attorney's instruction.

Q Other than the conference that you had with Mr. VanHoutan and ARM's counsel concerning the privilege instruction issue, have you conferred with anybody during the day about the substance of your testimony?

MR. VANHOUTAN: Same objection, same instruction.

A I'm taking my attorney's instruction.

Q Did you confer with Realtek about the substance of your testimony today?

MR. VANHOUTAN: Same objection, same instruction.

A I'm taking my attorney's instruction.

Q Have you spoken with Realtek's attorneys today?

MR. VANHOUTAN: You can answer yes or no.

A Yes.

Q Have you spoken with Realtek's attorneys in connection with the substance of your testimony today?

MR. VANHOUTAN: Objection, calls for attorney-client communication. I instruct you not to answer.

A I'll take my attorney's instruction.

MR. RIZK: I think that's a bad instruction, and I think it's inappropriate that if there were communications about the substance of your testimony, but we'll have to take that up.

Q So let's -- let me ask you this, sir. What made you decide to provide the clarifying

statement that you just made on the record?

MR. VANHOUTAN: Same objection, same instruction.

Instruct you not to answer.

A I will take my attorney's instruction.

Q What prompted you to decide to modify your testimony?

MR. VANHOUTAN: Same objection, same instruction.

Ex. 12, Larri Tr. (03/19/2024) at 250:19-252:21. Given Mr. Larri's role as effectively an extension of the litigation team of ARM and Realtek in this proceeding, and the egregious interference with his testimony during his deposition, Mr. Larri's declaration, the Release Notes, and his testimony should be viewed in that context and should be afforded little to no weight in connection with Realtek satisfying its factual burden with respect to how the patches are actually integrated with Realtek products and whether they truly disable the functions found to infringe.

3. Realtek's Submission Amounts to Nothing More than Attorney Arguments That Are Not Supported by a Credible Expert Opinion Regarding Non-Infringement

Beyond the glaring failures of proof discussed above, none of Realtek's declarants can credibly support by way of expert testimony Realtek's position regarding non-infringement of Claims 19 and 20 of the '381 Patent. Mr. Larri, as noted above, is an ARM witness whose declaration and testimony was limited to ARM's role in developing the patches and how they function from the perspective of ARM's legal department. He did not provide any opinion regarding non-infringement of the claims, and was in fact instructed not to answer whether he had an opinion on infringement when asked during his deposition:

Do you have an opinion as to whether claim 15 of the '381 Patent is infringed?

MR. VANHOUTAN: Objection, form, outside the scope of the declaration, calls for attorney-client communication.

I instruct the witness not to answer.

Q Do you have a personal opinion as to whether claim 15 of the '381 Patent is not

infringed that is not informed by a communication from counsel?

MR. VANHOUTAN: Same objection, same instruction.

Ex. 12, Larri Tr. (03/19/2024) at 24:11-20. Mr. de la Iglesia similarly did not consider the patent claims or the claim construction, nor did he provide an opinion regarding non-infringement.

Q. And let's go back to paragraph 1. So you were asked to review source code and other materials and to form opinion regarding the meaning of such source code and materials. So you did not apply the claim construction order in the 1318 investigation to your analysis for the purposes of this declaration, correct?

MR. JOHNSON: Objection. Form.

THE WITNESS: Well, shoot, I'm not aware of a construction order that impacts the functionality of source code.

Q. Did you apply any claim construction order when you were analyzing source code?

MR. JOHNSON: Objection, form.

THE WITNESS: I analyzed the source code, and the analysis of source code outside of any construction. That's for an infringement expert to do.

Q. And you're not the infringement expert?

A. That's right, I am not.

Q. So you don't have an opinion on whether the Realtek products with the patches you reviewed would infringe Claims 19 and 20 of the '381 Patent?

A. I understand --

MR. JOHNSON: Objection. Form.

THE WITNESS: I understand my investigation might be relevant to such a conclusion, but that's not a conclusion that I was asked to make.

Ex. 13, Iglesia Tr. (03/22/2024) at 10:22-11:15. That leaves Dr. Pfister to carry the burden of providing an expert opinion regarding non-infringement. But Dr. Pfister admitted **it's been a year** since he even read the '381 Patent that he provides his non-infringement opinion about.

A. Again, I'd have to read more of the patent. I haven't read it in a while. I'd be happy to do that if you like.

Q. When is the last time you read the patent?

A. A year ago, or thereabouts.

Q. And you didn't reread the patent in preparation for providing your opinion as found in Exhibit 1 of today's deposition; correct?

A. I did read the claims, obviously, especially claim 15, 19, and 20.

Ex. 14, Pfister Tr. (03/25/2024) at 205:14-23. Dr. Pfister's declaration is devoid of any claim interpretation or analysis that is based on the actual teachings of the '381 Patent in connection with his conclusions regarding non-infringement of Claims 19 and 20. When asked, for example, about what meaning he applied to the terms "priority" in Claim 20, he testified that he applied whatever meaning ARM applies to those terms, which effectively meant there was no priority because ARM said so.

Q. All right. You understand the word "priority" appears in Claim 20; right?

A. Yes.

Q. All right. So what understanding did you give the word "priority" as it appears in Claim 20, when it says "priority associated with each of the plurality of tasks"? What meaning did you give that --

MR. HIGGINS: Object to form.

Q. -- in reaching your opinions found in Exhibit 1?

A. Well, as you know, my opinion is based on the patches of the code, and the understanding of the working of the products or of the Mali GPU.

So my interpretation of the word "priority" in preparing my declaration with regarding to the patches was basically how, you know, ARM uses the term "priority."

Q. How does ARM use the term "priority"?

A. Well, I should rather say they're actually saying there is no priority of tasks -- or jobs, as we would say.

Ex. 14, Pfister Tr. (03/25/2024) at 207:19-208:12. Dr. Pfister's opinion amounts to nothing more than parroting what ARM determined was a non-infringing design, without considering the relevant terms of the patent claims as they would be interpreted by a POSITA based on a review of the patent, and then applying them to the analysis of the Realtek products at issue. And as the Federal Circuit has repeatedly warned, this type of conclusory opinion is entirely inadequate to

substantiate a non-infringement analysis. *See Yoon Ja Kim v. ConAgra Foods, Inc.*, 465 F.3d 1312, 1320 (Fed. Cir. 2006) (affirming JMOL of noninfringement where “conclusory [expert] testimony” was the basis for infringement); *Alpex Computer Corp. v. Nintendo Co.*, 102 F.3d 1214, 1222-23 (Fed. Cir. 1996) (holding that conclusory expert testimony did not support a finding of infringement). In fact, in the underlying ITC investigation, the ID rejected a similar conclusory analysis by Realtek in connection with the construction the term “engine” which relied exclusively on “inadequate” testimony by Dr. Pfister. Ex. 8, ID at 93 (citing *Phillips v. AWH Corp.*, 415 F.3d 1303, 1318 (Fed. Cir. 2005) for the proposition that “[c]onclusory, unsupported assertions by experts as to the definition of a claim term are not useful to a court.”). The same holds true here, as Dr. Pfister did not even bother to perform the first step of the infringement analysis of construing the claims to form his opinion on non-infringement. *See Advanced Steel Recovery, LLC v. X-Body Equip., Inc.*, 808 F.3d 1313, 1316 (2015) (“Determining infringement requires two steps: construing the claims and comparing the properly construed claims to the accused product.”) (citing *Abbott Labs. v. Sandoz, Inc.*, 566 F.3d 1282, 1288 (Fed. Cir. 2009).

4. Realtek Failed to Produce the Relevant Documentation to Describe How the Patched Mali Driver Uses the Pilot Shader Programs

Beyond its deliberate attempt to conceal the Performance Enhancement Patch, Realtek’s production of documents in connection with the use of compute tasks, in the form of pilot shader programs, amounted to the single self-serving Performance Enhancement Patch Release Note. Realtek’s strategy here is emblematic of the same strategy it deployed during the ITC investigation, which led AMD to move to compel in order to gain access to the relevant ARM documents that describe how the products actually work. These documents were then produced by ARM after expert discovery, and served to confirm Dr. Reinman’s opinion and undercut Realtek’s arguments. *See, e.g.*, Hrg. Tr. at 172:10-12 (ALJ Elliot recognizing that “I remember we had an issue with

ARM not – or ARM not giving us evidence in a timely fashion, so I'll allow it. Overruled.”). Given that it is now Realtek's burden to show non-infringement, all inferences regarding the use of pilot shader programs as general-compute tasks should be drawn in AMD's favor given Realtek's failure to produce the most relevant documents.

Realtek does not design the Mali GPUs or the driver. It receives these designs directly from ARM, and ARM maintains the technical engineering documents that fully describe how these products function as explained by the engineers that designed them. In the underlying investigation, the use of compute tasks and pilot shader programs was not at issue, and AMD had no reason to seek additional discovery of ARM documents that specifically describe how pilot shader programs are implemented. With its Ruling Request Letter, and the use of the Performance Enhancement Patch in its sample products, Realtek has placed these compute jobs and pilot shader programs directly at issue. Yet Realtek has failed to produce or procure from ARM the necessary documents to substantiate its claims of non-infringement.

ARM's own Performance Enhancement Release Note refers to “ARM engineering documentation” that describe the pilot shader.

1. A pilot shader binary that performs the initial calculations and stores in them in memory. The pilot shader binary is often referred to in Arm engineering documentation as “the pilot shader”.

Ex. 16, Performance Enhancement Patch Release Note at RTK-177-000004 (annotated). Moreover, each of the release notes that Realtek relies on include multiple versions that are in ARM's possession, but were also not produced.

Q. And this is the -- let's direct your attention to this page 3.

A Yes.

Q And the only -- section 1.1, it lists two versions of this document.

Do you see that?

A That's correct.

Q Okay. And then there's a -- an October 3, 2023 first release version, correct?

A That's correct.

Q Okay. And is there a document that exists within ARM that corresponds to that first release version?

A Yes, there is.

MR. RIZK: Okay. We would ask for that to be produced as well to us for the record.

Ex. 12, Larri Tr. (03/19/2024) at 122. Despite AMD's repeated requests for these documents, no such documents were ever produced by ARM or Realtek.

MR. Rizk: ...And we weren't given ARM's internal technical documents that we asked for that would help resolve some of these disputes as to Mr. Larri's change in testimony.

In addition to that, we had requested on the record materials to be produced during today's deposition that have been previously requested from Realtek, and from Realtek to ARM, including all testing and test-related documentation, including test results, test protocols, privilege log of testing documents that have been withheld, all communications between ARM and Realtek related to the patches, and especially to the performance enhancement patch, ARM documents describing the pilot shaders, prior release versions of all release notes.

Id. at 293:7-22. Given that it is Realtek's burden to show non-infringement, the most logical conclusion for its failure to produce the relevant technical documents is that they would contradict Realtek's position.

B. Realtek Has Failed to Present the Evidence Necessary to Substantiate its Basis for Representativeness of the R26 DDK Driver Version and its Patches

Realtek contends that the ALJ and the Commission's findings based on representativeness of the ARM Mali Bifrost Android OpenGL ES DDK r26p0 driver code should equally serve as the basis of a finding of non-infringement through the combination of a patched version of this driver code and the infringing Standalone Realtek ICs. Ex. 3, RRL at 8. But Realtek's attempt to shortcut its burden of proof is inconsistent with the actual findings of the ALJ and the Commission based on the issues presented during the investigation. Specifically, in the underlying investigation, AMD analyzed and presented un rebutted evidence regarding the software and driver code to

substantiate the finding of representativeness relied on by the ALJ and the Commission in connection with the limited purpose of the scheduling module serving as a source of tasks for the GPU. That is, to substantiate the representative nature of the ARM Mali Bifrost Android OpenGL ES DDK r26p0 driver, AMD simply had to show that the driver's functionality in serving as a scheduling module as a source of tasks for the ARM Mali GPUs was representative.

Realtek now presents an entirely different hypothetical, by allegedly placing specific disabling functions in a patch of the driver, and suggesting that this Patched Mali Driver is representative in its capacity to disable the infringing features of the Mali GPUs in the Standalone Realtek ICs. Even if this patched version could disable access to features of the GPU found to infringe, Realtek cannot rely on the representative findings of the ALJ and the Commission which never addressed the required factual findings to show representativeness under its new design. *See In re Certain Television Sets, Television Receivers, Television Tuners and Components Thereof*, 2019 ITC LEXIS 361, *42-43 (Feb. 2019) (affirming without further review the ALJ's finding that the burden to prove representativeness was not met because although infringement was shown for specific products there was a substantial lack of analysis between the products in the alleged representative categories and those that were judged to be infringing). That is, the ALJ and the Commission did not need to analyze issues as to how the driver interacted with the other software beyond the driver's role as a scheduling module from which the GPU receives tasks. Realtek's new Patched Mali Driver allegedly now serves a host of other functions that are specific to the infringing aspects of the claims, and allegedly serve to disable access from applications executing on the CPU to these infringing functions of the GPU.

As Dr. Reinman explains, in order to determine whether any one version of a driver is representative of other versions in connection with a particular set of functions, a comparison of

the code that would show the differences should be generated and analyzed. Ex. 1, Reinman Decl. at Section III.A. Indeed, ARM's own corporate representative, Mr. Larri, emphasized during the investigation that different version of the driver code behaved differently when questioned about functionality beyond the driver's basic function of serving as a scheduling module and source of tasks. For example, when questioned about the driver's capability to schedule tasks to a particular Job Slot (Job Slot 2) of the Mali GPUs, Mr. Larri responded that "[t]he only place that I know of to go to answer that question is the source code of the driver development kit for each...**to check each version of the driver development kit, because the relevant source code is different in different versions of the driver development kit.**"). Ex. 15, JX-0192C (Larri Tr. (11/01/2022)) at 110:14-111:25 (excerpt at 111:20-25, emphasis added); *see also id.* at 173:13-17, 174:8-12 (reiterating that ARM has no way of knowing if any particular customer has used a particular version of the driver). Yet despite this background, Realtek now suggests that the r26p0 driver version, as patched, is representative not only as to all version of the driver, but also with respect to all functions of the driver beyond the basic function of serving as a source of tasks.

Beyond failing to substantiate its own basis of representativeness, AMD was not even permitted to use the source code produced by Realtek for the driver versions related to the Valhall GPUs in connection with the deposition of Mr. Larri. *See* Ex. 52, Robert Benson Annotated Email (03/18/2024). That is despite the fact that Mr. Larri could not answer questions about how the driver functioned without reference to the source code.

Q Yeah. There's separate -- there's a separate device driver source code for the Natt, right?

A. Well, it's used for the Bifrost GPUs that we've been discussing, because it's not -- yeah.

Q Okay. Is it the same exact driver for the Natt?

A So sorry, just remind me, which Mali GPU is Natt?

Q Natt is the G57.

A Well, it's used for the Bifrost GPUs that

A So it's the Valhall -- it's the Valhall.

Q Yes.

A Right. **So the device driver for the Valhall GPUs differs in various ways from the device driver for the Bifrost GPUs.**

Q Yeah. I said the device driver. Let me repeat my question again.

You're not going to be able to answer my question unless I show you the device driver source code for the Natt G57 GPU, correct?

A Correct.

Ex. 12, Larri Tr. (03/19/2024) at 117:5-8 (emphasis added).

C. Realtek Failed to Produce Source Code to Substantiate its Claims that the RTD2851MPJ is Representative of the RTD2951M or How Sample Boards 2 and 3 Operate

Sample Boards 2 and 3 both contain Realtek's new chip RTD2951M. Sample Board 2 contains the Patched Mali Driver code, r26p0, whereas Sample Board 3 contain the non-patched driver code, r26p0. Realtek has not met its burden of proof to obtain a non-infringement ruling with respect to Sample Board 2, Sample Board 3, or the RTD2951M Standalone Realtek IC. To date, Realtek has not produced any hardware source code, RTL code, for its "new chip," RTD2951M. Instead Realtek has only produced the driver source code, the "patches," and has merely reproduced from the ITC-1318 Investigation the hardware source code for its old chip, RTD2851MPJ. Ex. 38, Christopher Higgins Annotated Email (03/08/2024) at 1, 3. In response to AMD's repeated requests for the RTL code for the "new chip" and the legacy chip RTD2851MPJ **as used in** the RRL Articles, Realtek refused because "there is no new [RTL] code" compared to what was produced at the ITC-1318 Investigation. *Id.* at 1. The representation by Realtek's counsel, by itself, is not evidence that the two Standalone Realtek ICs are the same as he claims. *See* Reinman Decl. ¶¶ 117-18. Moreover, based on testing, there appear to be differences between

Sample Board 1 and Sample Board 2. As Realtek has not produced the RTL Source Code for RTD2951M, Realtek has failed to meet its burden of proof of showing that the RTD2951M Standalone Realtek IC, or the Sample 2 and Sample Boards incorporating the same do not infringe. *See id.* at Sections X.A-D.

D. Realtek Failed to Produce Evidence that its Downstream Consumer Manufacturers Cannot Modify Realtek’s Driver Patches as They Wish

As part of its Ruling Request Letter, Realtek has submitted that an “option” can be implemented that “would prevent any old version of the Mali driver code from being used with the New Realtek Products.” Ex. 3, RRL at 52. But Realtek fails to substantiate how this option would actually be implemented given that its expert Mr. de la Iglesia, cited to code that doesn’t perform as he describes. Moreover, recognizing that this “option” cannot actually prevent its own sophisticated customers from modifying the driver versions used with its products, Realtek elected to withhold relevant documents that instruct its customers how to replace the driver files used with the Realtek products.

As a threshold matter, as described by Realtek and its expert Mr. de la Iglesia, this “option” is targeted to end users in order to prevent them from “hacking” the Realtek Products to use an old driver. While using a different version of a driver or operating system is far from hacking, as a threshold matter, Realtek’s Products are not sold directly to end users. Rather, Realtek sells its integrated circuits and development boards to sophisticated downstream original equipment manufacturers (OEMs), including, for example, television OEMs. And there is nothing to prevent Realtek’s actual customers from modifying the driver. In fact, Realtek provides these customers with the source code and all the tools necessary for them to make any necessary modifications to the software, including the driver. Ex. 34, Realtek Corporate Representative Juo Jung Wu Tr. (11/16/2022) at 47:12-21 (“Compiling the kernel driver is done by the vendor for each platform.

It's not actually done by Realtek.”); Ex. 35, Realtek Corporate Representative Ming Feng Chen Tr. (11/14/2022) at 135:19-136:5 (“To my knowledge, TCL has the right to modify anything incorporated in their televisions.”).

Realtek's experts provided their opinions apparently on the incorrect assumption that Realtek's customers cannot make modifications to the driver. Despite Realtek's experts Dr. Pfister and Mr. Iglesia adamantly asserting that any downstream change to the driver software is impossible, Realtek's own engineers testified in the 1318 proceeding that Realtek's customers can make any modifications to the source code and can compile their own binaries of the operating system, including the driver. *Compare* Ex. 14, Pfister Tr. (03/25/2024) at 112:1-9 *and* Ex. 13, Iglesia Tr. (03/22/2024) at 25:21-26:1, 33:9-34:3, *with* Ex. 12, Larri Tr. (03/19/2024) at 199:7-19 *and* Ex. 34, Realtek Corporate Representative Juo Jung Wu Tr. (11/16/2022) at 47:12-21 *and* Ex. 35, Realtek Corporate Representative Ming Feng Chen Tr. (11/14/2022) at 135:19-136:5. So there is nothing preventing a Realtek customer from using an unpatched version of the driver with the Realtek products, and in fact the entire purpose of a development board is for OEMs to test different software to test how the Standalone Realtek ICs would perform in their systems. Perhaps due to their concern about this issue, Realtek refused to present any Realtek fact witnesses for deposition in this proceeding.

Moreover, even assuming that Realtek's customers could not modify the driver, Realtek's request for consideration of this option is, at best, theoretical. Mr. de la Iglesia provides, in two paragraphs, his entire analysis of how this option works, and references a “one time programmable boot key” that is apparently in the processor. Ex. 4, Iglesia Decl. ¶¶ 25-26. Meanwhile, Realtek's counsel has represented that Sample Board 2 and Sample Board 3 were provided to demonstrate this option, and that the one-time programmable boot key has “been produced.” Ex. 38, Chris

Higgins Annotated Email (03/08/2024) at 1, 3; Ex. 39, Chris Higgins Annotated Email (02/23/2024) at 2-3). But following exhaustive review of the source code of the Sample Boards, which Mr. de la Iglesia never examined, Dr. Reinman determined that the source code produced for these boards does not actually include a different one-time programmable boot key. Ex. 1, Reinman Decl. at Section XIII.2. Rather, the only keys that were present appear to be the RSA keys that are identical between Sample Board 1 and Sample Board 2. *Id.* (citing RTK-177-SC-003096 and RTK-177-SC-003097 which include the identical RSA keys for Sample Board 2 and Sample Board 3). And when Dr. Reinman followed the code directories that are cited by Mr. de la Iglesia which allegedly check these keys in his report, he found that the code isn't even used for the television products. Ex. 1, Reinman Decl. ¶¶ 212-13.

Apparently recognizing the trouble with this option, Realtek refused to produce the documentation that it provides its customers with the development boards, going as far as representing that there were no such documents. Ex. 40, Christopher Higgins Annotated Email (02/27/2024) (referring to Ex. 39, Christopher Higgins Annotated Email (02/23/2024)). Following AMD's repeated requests for certain operating system code, Realtek finally provided some directories on the source code computer that pertain to certain libraries of the operating system code used with the Realtek products. These directories also included certain documents, perhaps inadvertently, in a subdirectory "`mln5_11_with_Patch.tar\mln5_11_with_Path_kernel_system_docs_`." Realizing that some of these documents do in fact include instructions for development boards, Realtek maintained the baseless position that these documents were source code under the NDA and refused to produce them. It was not until April 1st, just 4 days before AMD's response was due, that Realtek finally

agreed to produce these development board documents. Ex. 41, Michael Chow Annotated Email (04/01/2024).

These documents not only include instructions for the development boards, but provide directions to customers on how to replace the very driver images that Realtek suggests cannot be modified. Ex. 1, Reinman Decl. at Section X.4. As just one example, these documents include a file that is literally entitled “How to Modify Contents in Board” which includes specific instructions for replacing certain image files that Realtek suggests cannot be modified.

```
# How To Modify Contents In Board
## Fastboot
```

Ex. 24, RTK-177-000024.

```
## ADB
```

```
To build single android module and replace the one on board
--> build method is the same as before( source, lunch, and mm), replacement step by
step as follows:
```

1. need to do OEMLock off to disable avb(android verify boot)

```
bootcode:
```

```
Realtek> env set OEMLock off
Realtek> env save
Realtek> reboot
```

2. execute adb under PC server to disable dm-verity

```
adb connect your_device_ip:5555
adb root
adb disable-verity
adb reboot
```

3. execute adb under PC server to replace you ko/so file

```
adb root
adb remount
adb push host_path device_path
adb reboot
```

id. at RTK-177-000024-025. And there is little doubt that the documents that AMD discovered on the source code computer are just a small subset of documents in Realtek's possession that contradict Realtek's positions in connection with the use of its development boards.

XI. CONCLUSION

For the reasons stated herein, AMD respectfully submits that Realtek's request should be denied as it has failed to show that its proposed software based design would render the RRL Articles non-infringing.