

UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.

In the Matter of:

CERTAIN GRAPHICS SYSTEMS, INVESTIGATION NO.
COMPONENTS THEREOF, AND DIGITAL 337-TA-1318
TELEVISIONS CONTAINING THE SAME

* * * CONFIDENTIAL BUSINESS INFORMATION * * *

The Oral Discussion

Taken Via Remote Teams Videoconference

Commencing at 1:03 p.m.

Wednesday, May 1, 2024

Stenographically reported by:

Joanne Marie Bugg, CSR-2592, RPR, RMR, CRR

Job No. 6676614

1 APPEARANCES:

2
3 ADAM S. RIZK

4 Mintz, Levin, Cohn, Ferris, Glovky and Popeo PC

5 One Financial Center

6 Boston, Massachusetts 02111

7 617.542.6000

8 arizk@mintz.com

9 amd-itc@mintz.com

10 Appearing on behalf of Advanced Micro Devices, Inc. &

11 ATI Technologies ULC

12
13 CHRISTOPHER J. HIGGINS

14 Orrick, Herrington & Sutcliffe LLP

15 2100 Pennsylvania Avenue NW

16 Washington, D.C. 20037

17 202.339.8400

18 chiggins@orrick.com

19 Appearing on behalf of Realtek Semiconductor Corp.

1 ALSO IN ATTENDANCE:

2

3 Aaron C. Cook

4 Dax Terrill

5 Jessica Wu

6

7 Michael Renaud

8 Sam Davenport

9 Matthew Karambelas

10 Amy LoBue

11 Tawfik Goma

12 Barbara Murphy

13 David Nickel

14 Jordan Coyle

15 Steve Baik

16

17

18

19

20

21

22

23

24

25

TABLE OF CONTENTS

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

PAGE

ORAL DISCUSSION BY MR. HIGGINS:	9
ORAL DISCUSSION BY MR. RIZK:	72
ORAL REBUTTAL DISCUSSION BY MR. HIGGINS:	147

EXHIBITS

EXHIBIT	PAGE
(Exhibits not marked.)	

1 Teams Videoconference

2 Wednesday, May 1, 2024

3 1:03 p.m.

4
5 MR. COOK: At this time, is counsel for
6 Realtek present, and could everybody who is attending
7 on behalf of Realtek please introduce themselves?

8 MR. COYLE: Yes. Good morning, everyone.
9 This is Jordan Coyle at White & Case on behalf of
10 requestor Realtek, and I'm joined today by my
11 colleague, Chris Higgins. And Chris will be conducting
12 the presentation on behalf of Realtek this afternoon.

13 MR. BAIK: Steve Baik also for Realtek.

14 MR. COOK: Thank you. Is that everybody for
15 Realtek?

16 MR. COYLE: It is, Mr. Cook. Thank you.

17 MR. COOK: Thank you for that. So could --
18 is counsel for AMD present, and could everybody
19 appearing on behalf of AMD please introduce themselves.

20 MR. RIZK: Yes, hi. Good afternoon, Mr.
21 Cook and Mr. Terrill. My name is Adam Rizk. I'll be
22 representing Advanced Micro Devices and ATI
23 Technologies ULC with the law firm of Mintz Levin.

24 I'm joined today by other attorneys also on
25 behalf of AMD, and with the law firm Mintz Levin,

1 including Michael Renaud, Sam Davenport, Matthew
2 Karambelas, and Amy LoBue. I'm also joined by our
3 co-counsel, Tawfik Goma, of the Goma Law Firm, and
4 Barbara Murphy and David Nickel of Foster Murphy Law
5 Firm also on behalf of AMD and ATI Technologies.

6 MR. COOK: Thank you for that. So before we
7 dive into it, I'm just going to go ahead and check one
8 more time. Has the court reporter gotten all the
9 information they need at this moment?

10 COURT REPORTER: Yes.

11 MR. COOK: Thank you for that. So let's go
12 ahead and get started.

13 Thank you for joining today's oral
14 discussion concerning a request for an administrative
15 ruling submitted by Realtek Semiconductor Corporation
16 under 19 CFR part 177 related to the limited exclusion
17 order issued in US International Trade Commission
18 Investigation Number 337-TA-1318.

19 Participating in today's discussion from the
20 Exclusion Order Enforcement Branch of the regulations
21 and rulings directorate of the Office of Trade at US
22 Customs & Border Protection are branch chief Dax
23 Terrill, attorney advisor, Jess Wu, and myself,
24 attorney advisor Aaron Cook.

25 With the agreement of the parties, the

1 Exclusion Order Enforcement Branch is handling this
2 ruling request on an inter partes basis with Realtek,
3 the manufacturer, and AM order of the merchandising
4 question, and Advanced Micro Devices, Inc., and ATI
5 Technologies UCL, the complainant from the underlying
6 ITC investigation. These are the parties identified as
7 parties with indirect and demonstrable interest in the
8 questions presented by this ruling request.

9 The Exclusion Order Enforcement Branch also
10 thanks the parties for their submissions up to this
11 date, and also for their participation in today's
12 discussion.

13 Who will be speaking on behalf of Realtek in
14 today's discussion?

15 MR. HIGGINS: Good afternoon, Mr. Cook.
16 This is Chris Higgins from Orrick. I'll be speaking on
17 behalf of Realtek.

18 MR. COOK: Thank you for that. And who will
19 be speaking on behalf of AMD in today's discussion?

20 MR. RIZK: Yes, hi, Mr. Cook, Adam Rizk. I
21 will be speaking on behalf of AMD today.

22 MR. COOK: Thank you. So I'm going to go
23 over kind of the format of today's discussion. First we
24 will hear from counsel for Realtek, who will have one
25 hour and 30 minutes to present, followed by a 10 minute

1 break. Next, counsel for AMD will have one hour and 30
2 minutes to present, followed by a 10 minute break.
3 Finally, counsel for Realtek will have 30 minutes for a
4 rebuttal.

5 The parties are welcome -- or parties are
6 welcome but not required to use the entire amount of
7 time allotted. However, any allotted time that is not
8 used cannot be saved or allocated for later use.

9 We understand that technical difficulties
10 may arise, and we'll attempt to give back any time lost
11 due to those.

12 Additionally, I will give a five minute
13 warning, and a one minute warning through the Teams
14 chat to keep you updated on your time.

15 The last thing to note is if it's not your
16 side's turn to make the argument, or if you're not
17 speaking, please make sure that you have your device
18 muted.

19 Did any parties have any questions regarding
20 today's discussion before we begin?

21 MR. HIGGINS: Nothing from Realtek.

22 MR. RIZK: Not for AMD.

23 MR. COOK: Seeing that there are none,
24 counsel for Realtek, please proceed.
25

1 ORAL DISCUSSION BY MR. HIGGINS:

2 MR. HIGGINS: Thank you. And just give me
3 one second so I can share my screen and get our slides
4 up.

5 Good afternoon. Again, this is Chris Higgins
6 from Orrick on behalf of requestor Realtek. And before
7 I get started, we first want to thank the branch for
8 its time in consideration of this matter. As you can
9 see from our briefing, this is extremely important to
10 my client, Realtek.

11 And while we may spend a good deal of time
12 discussing various issues today, what we want to
13 emphasize is that our ruling request was intended to
14 be, and still is, very straightforward.

15 In essence, we completely removed and
16 disabled the specific features of the accused products
17 that the ID and commission found to infringe. And there
18 really isn't any dispute that we've done that.

19 This is what I intend to cover today. And
20 theoretically, we would stop after sections three and
21 four because those cover the actual two patches that we
22 have requested a ruling on, and those are what disabled
23 the features that were found to infringe. What you'll
24 see in part six later on, and what I would expect AMD
25 to spend a lot of its time on, are talking about issues

1 that are simply irrelevant to this proceeding, and
2 don't matter.

3 And I'll spend some time in my opening part
4 discussing AMD's numerous new arguments. But when it
5 comes down to it, these are not issues the branch need
6 even consider, because they were not raised below in
7 the investigation, and AMD could have raised them.

8 AMD is essentially trying to open up an
9 entirely new infringement case, and this is not the
10 forum, nor is there jurisdiction for it here.

11 That said, I can address as much or as
12 little of these new arguments as the branch desires.
13 What I want to avoid, however, is spending the time
14 today addressing the needless rhetoric that AMD has
15 filled its sur-reply with. This request should be
16 decided based on the facts, and not based on AMD's
17 rhetoric and baseless allegations.

18 And just briefly, we've covered this in our
19 briefing, but I'll step through this in less than a
20 minute. Just want to orient us on what we'll be
21 discussing today. And this is the scope of the
22 investigation.

23 We're going to be talking about products
24 that include an integrated circuit, and have an ARM
25 Mali GPU on them, as well as the necessary components

1 that were found by the ID and the commission to
2 infringe.

3 And I'll come back to this later. But there
4 really shouldn't be a dispute here based on the ID and
5 the commission's ruling on what is at issue.

6 This was essentially the scope of the case,
7 and I'll flip through this quickly. This is where we
8 started in this case. We had five patents, and almost
9 40 asserted claims.

10 Prior to and during the hearing, AMD dropped
11 a lot of its claims, a lot of its patents, including
12 the '628 patent which the ALJ suggested AMD drop during
13 the hearing. Following the hearing, all we have left
14 are the two claims of the '381 patent, claims 19 and
15 20, and that will be the focus of our discussion today.

16 But before we get into all the details of
17 the patches, and what was found to infringe, I want to
18 spend a second going over the layout of the GPU in the
19 accused products that we'll talk about today. And the
20 focus will be on these -- on the three components that
21 I've highlighted here. And the first component is the
22 driver. That's in the blue box at the top. The driver
23 is what configures the GPU. It essentially controls its
24 operation, and it tells it what to do. Without it, the
25 GPU is just a tiny useless brick.

1 And then we'll spend some time talking about
2 the job manager and the job slots, of which there were
3 originally three. But as we'll see from the patches
4 that we've applied to the accused products, there are
5 now only two job slots effectively in accused products.
6 And that will be important to claims 19 and 20.

7 And then finally in orange are the execution
8 engines. We'll discuss these in more detail when we
9 get to claim 19. But the important point to remember
10 here is this is where the computations occur within the
11 GPU. And these will be what are called the shader core
12 when we get to claim 15 and 19.

13 In addition to the components in that high
14 level overview, there are a few key terms and concepts
15 that will come up quite often today, so I want to cover
16 them up front, and at least introduce them.

17 The first are ARM jobs. This is what gets
18 created by the driver. Remember that blue box on the
19 last slide. It gets sent to the GPU. These are
20 essentially units of work given to the GPU. And in the
21 investigation, the claim here as you can see on the
22 left, it recited tasks.

23 The ID and the commission found that the ARM
24 jobs in the accused products sent from the driver,
25 these are what equated to the claim tasks in the claim.

1 And, again, this was based on what AMD argued. The ID
2 and the commission adopted their argument.

3 And then shown below in this color coded
4 box, this is what the breakdown of the ARM jobs was
5 during the investigation. For graphics processing
6 tasks, which are in claim 19, those were fragment
7 shader jobs, vertex shader jobs, as well as geometry
8 shader jobs.

9 And then, secondly, you had general compute
10 tasks. That was the other category of claim 19. And
11 those were found to be compute shader jobs. This was
12 argued by AMD, and adopted by the ID and the
13 commission.

14 Now, these ARM jobs, once they come from
15 that blue box that we talked about the driver, they get
16 sent to the job manager, which is the green box. I've
17 copied that part of it down below you see at the bottom
18 with now the two slots. This is essentially a holding
19 area for the jobs.

20 Now, there's job slot zero. That could hold
21 fragment shading jobs. And there's job slot one that
22 holds essentially everything else, vertex, geometry and
23 compute. And these types of jobs and how they fit in
24 the job slots, that will come up again when we talk
25 about claims 19 and 20.

1 So what we have in terms of what was found
2 to infringe, and the main elements from the claims
3 compared to the accused products, the plurality of
4 engines term from claim 15, that was found to read on
5 the job slots in the job manager of the ARM Mali GPU.

6 And then the scheduling module, highlighted
7 in yellow, that was found to read on the Mali driver.
8 And that's what we'll talk about a lot today. That's
9 what we have modified, and that's what the patch is
10 applied to. Remember the driver's what configures the
11 GPU.

12 And then for claim 19, it was those
13 different ARM jobs we just discussed. The fact that
14 there are compute shader jobs compared to fragment and
15 vertex shading jobs, was those two types of tasks that
16 were found to satisfy claim 19.

17 And then finally for dependent claim 20, it
18 was simply the priority of compute jobs in job slot
19 two, over compute jobs in job slot one, that was the
20 sole reason for the ID's finding and the commission's
21 finding as it adopted for the priority of claim 20.

22 And these features at the bottom that were
23 found to infringe, this is exactly what we disabled to
24 prevent any accused products, and that's what we'll
25 spend our time discussing today.

1 But first a brief overview of what Realtek's
2 ruling request involved. There seems to be some
3 misapprehension either on purpose or confusion on the
4 part of AMD, so I want to walk through this quickly and
5 lay out what our ruling request is.

6 And simply put, our ruling request is based
7 on two representative ARM patches that implement a new
8 design that disables specifically the features found to
9 infringe for claims 19 and 20 in the accused products.
10 And these are the ARM patches labeled 04 and 05.

11 These patches remove the ability to disable
12 job slot two in the accused products, and they also
13 prevent the concurrent processing in the shader core of
14 compute shader jobs and fragment shading jobs. That's
15 specifically what was found to infringe claim 19. This
16 is the scope of our ruling request.

17 And what we don't really see in AMD's papers
18 is -- or expert declaration is really much of a
19 legitimate dispute that these ARM patches don't do what
20 we say they do. Now, we see in their expert report or
21 their declaration mention that job slot two is
22 disabled. It can't be used. And we see in there that
23 the patch as applied, there is no concurrent processing
24 of an ARM compute job versus an ARM fragment job.

25 As I mentioned at the very beginning, this

1 is why we think our ruling request is straightforward.
2 What was found to infringe, we have specifically removed
3 from the accused products, and there's not much of a
4 dispute that we didn't do that.

5 And what we've done with the patches which
6 are made to the driver, the driver code that configures
7 the GPU, to disable and prevent the specific features
8 found to infringe, this goes even further than some of
9 the recent decisions from the branch.

10 For example, in the decision this year
11 regarding the Apple watches, there was a software
12 modification made by Apple not to the watches, to the
13 Apple phones, and this was found to negate infringement
14 based on the hardware in the watches, because during
15 the pairing process, the phone would send something to
16 the watch, and it would change a value in the watch,
17 and that would prevent the watch from accessing that
18 feature that the hardware could do.

19 And based on that, the branch found that
20 this sufficiently designed around the claims, and it
21 was no longer subject to the exclusion order. Here, our
22 code that we changed is actually in the product. It's
23 flashed in as binary. It's there. It's what configures
24 the GPU. We don't need a separate device to come in
25 every time and change something in a pairing mode. So

1 we've gone steps further than what was found
2 permissible in the Apple case.

3 Similarly, there's a decision regarding
4 Diebold ATM machines. And here there was software that
5 was modified, and it was used to prevent certain
6 sensors in the ATM from performing a scanning function.

7 Now, the sensors in the ATM that were found
8 to include the infringing capability, those weren't
9 modified or removed. It was a software change that
10 prevented them from being used. Again, we went further
11 than this.

12 Now, AMD spent a large portion of its
13 opposition brief arguing that us modifying the driver
14 code was simply not enough, because we didn't
15 physically modify the RTL and the hardware within the
16 GPU. But we know from at least these two decisions,
17 plus several others that we cited in our briefing, AMD
18 is simply wrong.

19 So it came as no surprise when they moved
20 this section to the end of their sur-reply, and gave it
21 much less attention. And I'll have some slides later on
22 addressing this argument in more detail, including how
23 in order to make this argument AMD needs to backtrack
24 on arguments it made below. It needs to go against
25 what was found by the ID and in the commission.

1 But because of these decisions, and
2 regardless of what AMD argues later regarding some
3 scheduling module whether it's required, the
4 modification we made to the driver code, regardless of
5 any of those arguments, is sufficient to bring the
6 products outside the scope of the LEO because we
7 disabled what was found to infringe.

8 So what that leaves us with, and what we've
9 shown based on the two patches is that we've removed
10 the infringing features from the accused products, and
11 entry should be permitted for these products below.

12 First we have Realtek IC's that they don't
13 have any driver code or external memory. They're not in
14 a product. They simply can't operate, and they don't
15 have the infringing configuration, much less capability
16 based on what was found to infringe by the ID and the
17 commission.

18 Second, we have demo boards and other
19 products with Realtek ICs that have the patch driver
20 code. And these fall into several categories. First,
21 we have demo boards that Realtek brings to the US, and
22 maintains possession of that after importation.

23 And then we have demo boards that Realtek
24 may ship to the US, and delivered to a third party
25 customer.

1 And then finally we have any products that
2 incorporate a Realtek IC with dispatch driver code, as
3 well as the TCL televisions that were at issue here.
4 All of these with the patch driver code are outside the
5 scope of the LEO, because we have disabled specifically
6 what was found to infringe.

7 Based on that, and what we think is straight
8 forward, why do we see so much briefing from AMD, 200
9 some pages in this proceeding? It's because all it has
10 to argue here is essentially brand new case for --
11 again, there's no jurisdiction here for that case.

12 There are new infringement theories based on
13 features that existed in the accused products during
14 the investigation, and AMD did not raise. There are new
15 claim constructions that AMD could have raised, but
16 didn't. There are arguments that AMD is making now that
17 directly contradict what it argued below, what it's
18 arguing in an IPR that's pending for the '381 patent,
19 and what the commission found.

20 AMD spends a lot of its time arguing about
21 what we believe is irrelevant, and what you've seen is
22 called this performance enhancement patch, which again
23 is not part of our ruling request. But regardless, they
24 spend a lot of time complaining about this trying to
25 say that we concealed something. We provided it to them

1 on day one. It was on the source code computer. And
2 not just this release note they refer to. We gave them
3 the code. Then I'm sure we'll see the words secret and
4 concealed throughout their argument today, because it
5 was in their briefing. But they can call it whatever
6 they want. The fact of the matter is we provided it to
7 them on day one.

8 Now, this performance enhancement patch that
9 AMD focuses a lot of its argument, once again, it's not
10 in a ruling request. But we provided it to AMD as it
11 existed, and it may be something we may implement in
12 these redesigned products. But it doesn't affect or
13 remove the patches that were the subject of our ruling
14 request, thus that led AMD to make up these new
15 arguments. New infringement arguments, new claim
16 construction arguments. But we provided this patch on
17 day one. They've had it. It doesn't undue what's in the
18 patches. And it simply is not relevant.

19 So with that, I want to get to the two
20 claims here, and those were claims 19 and 20. And I'll
21 do this a little backwards. I'm going to start with
22 claim 20 first. And the reason for this is, as you see,
23 it's pretty simple. There's only a couple slides here,
24 and it directly relates to a feature that isn't
25 disputed that we removed from the accused products.

1 So to kind of set the stage here, let's
2 start with what AMD argued, and what the ID, and
3 eventually the commission adopted. And what's shown on
4 the slide here in slide 21, this is the demonstrative
5 used by AMD's expert, Dr. Reinman, during the
6 investigation. This is what they argued. This is what
7 they put up during the hearing to say this is what
8 infringes.

9 And AMD's position was there are three job
10 slots. And job slot two holds compute jobs. Job slot
11 one also holds compute jobs. There was something in
12 the products in the code that would always pick the
13 compute job in job slot two over job slot one. And
14 based on that, AMD argued that's a priority that meets
15 claim 20. And this was Dr. Reinman's testimony that
16 went along with this demonstrative. Again, testified
17 that it selects the highest number compute job slot
18 first. So, in other words, job slot two has priority
19 over job slot one. That was the basis for AMD's
20 argument.

21 And what did the ID do? The ID adopted
22 AMD's argument verbatim. The ID found that ARM jobs,
23 the claimed tasks located in job slot two have a
24 priority over those in job slot one. So we knew what
25 AMD argued, and that's exactly what the ID adopted. So

1 we literally took that and removed it from the accused
2 products. Job slot two no longer exists in the GPU.
3 It cannot be used. Nothing can be loaded from it.
4 Nothing can be loaded into it. It simply does not exist
5 any more in the GPU.

6 And because of that, there is no priority of
7 job slot two over job slot one. As job slot two doesn't
8 exist, we've removed the only basis for infringement in
9 the investigation.

10 And we have two main sources of support that
11 we produced in this case. The first was an ARM release
12 note. This is something that ARM provided that
13 accompanied the patches to the accused products.
14 Again, the driver code, and the GPU were provided by
15 third party ARM to Realtek. So the patches here were
16 done by ARM, provided to Realtek, along with this
17 release note. And what the release note says
18 unequivocally is job slot two can never be used. So
19 exactly what the ID found infringed, it takes it out of
20 the accused products.

21 And it's not just the release note that ARM
22 provided. We actually have the code. We provided the
23 code with our ruling request. We provided all of the
24 patches on the source code computer. And what the code
25 says is exactly what the release note says. Job slot

1 two is disabled. You can't schedule anything in job
2 slot two. You can't take anything from it.

3 So job slot two, which was found to
4 infringe, commissioned in the ID, priority of job slot
5 two over job slot one, what did we do with the patch in
6 the accused products? We disabled and removed the use
7 of job slot two.

8 So specifically what the ID found infringed
9 was adopted by the commission. We have removed it from
10 the accused products. This means there is no longer
11 infringement of claim 20 by the patched products, and
12 they're outside the scope of the LEO.

13 Now, with that said, there are some
14 arguments that AMD has raised, and we consider them new
15 arguments that they've raised in their briefing. And
16 there are at least 10 of these new arguments. They
17 could have been -- they could have been raised below.
18 In the legacy accused products, functionality existed,
19 but AMD did not raise them. And they're also just
20 wrong.

21 I have them later in the slides. And if
22 time permits in the opening, I may cover them quickly.
23 But I will save most of these for rebuttal, depending
24 on which of the 10 or 12 new arguments AMD may argue
25 today.

1 I'll pause there before moving onto claim 19
2 in case there are any questions. Otherwise, I'll move
3 ahead in the claim 19.

4 MR. TERRILL: I did have one question. I
5 know that Realtek has sophisticated consumers, and I
6 think there was some question of you have this new
7 driver code, but will Realtek's consumers use the
8 driver code? Do you have any like terms of agreement
9 for customers on, I guess, proved or unapproved code,
10 and how does Realtek kind of guarantee that this is the
11 code that's going to be on the devices?

12 MR. HIGGINS: Sure. I think there's a couple
13 layers to that. I think you used the terms consumers
14 and customers. I want to make sure we divide those
15 out. Realtek's customers would be manufacturers such as
16 TCL or other OEMs that are making televisions that
17 incorporate these products.

18 So Realtek provides it SOC, and the code to
19 those companies, and the code that Realtek provides is
20 what is supposed to be used in the product that's
21 what's intended to be used. That's what the customers
22 have to use.

23 So, and frankly, TCL, for example, Realtek
24 provides this code to TCL, and this patched code, which
25 let's assume it's found to be outside LEO, the

1 certification go along with that. So if TCL's importing
2 these, there would be a certification that says this is
3 the patch code. It prevents concurrent execution, and
4 there's no priority. So that would go along with it.

5 So there is -- there is no reason, no theory
6 behind AMD's argument that TCL or some other downstream
7 customer could just change this. You'd have the
8 certification, and that should be binding on them.

9 To go a step further, and with respect to
10 consumers, this driver code that is patched is provided
11 even to TCL. It's a binary. It's not something you
12 could go into and change.

13 The only argument AMD has potentially raised
14 is this -- maybe someone could go in and hack it. But
15 to us, that's not even possible. I mean, there may be
16 one or two people that could possibly hack this code
17 and redo all the binary. But they're not given the
18 source code. So the binary that's coming in, all these
19 bits and zeros and ones, it's not something that's
20 going to be easily changeable at all.

21 And even in light of that, and we saw that
22 from the recent Apple ruling case where I believe
23 Masimo was arguing that, well, you could go jailbreak a
24 phone and undo what Apple did in its software patch.
25 But the branch there found that -- that there wasn't

1 enough. When Apple provided it, the code was in there
2 as it intended and removed the infringement feature.
3 The fact that someone could go in, and jailbreak the
4 phone and try and undue it, that doesn't take the
5 product and import it outside the scope of the LEO.

6 And I think it's also worth noting, to jump
7 back to even the TCL issue, the initial determination
8 by the ALJ found in this investigation that TCL did not
9 modify the code that Realtek provided. It was put onto
10 the products. Realtek designs these. It puts it on
11 their -- TCL puts it on their TVs. So I think that
12 there's not much behind these speculative assertions
13 that AMD's making.

14 And then the last thing I would note, and we
15 did put this in our -- in our request that noted that
16 Realtek, you know, could go a step further and provide
17 this one time programmable boot key. Essentially, if
18 someone tried to hack it, the one in a million chance,
19 and they tried to load the old driver code back on,
20 this boot key would prevent the SOC from booting up.

21 And we even put that on a demo board and
22 provided it. AMD's expert tested that product, and it
23 couldn't boot it up at all. It's simply a useless
24 brick.

25 So there are multiple layers to this. We

1 don't think we need to get to that one time
2 programmable boot key, because there's simply no basis
3 that any kind of customer or consumer would take this
4 code and hack it and try and undue what's already been
5 done.

6 MR. TERRILL: Thank you for that
7 explanation.

8 MR. COOK: Before moving onto claim 19,
9 could I ask another question about claim 20 and what I
10 understand to be AMD's -- one of its infringement
11 arguments? Could you address AMD's argument that in
12 the redesign if job slot two is disabled or
13 nonexistent, the redesign is still infringing because
14 it prioritizes job slot zero over job slot one?

15 MR. HIGGINS: Sure. And let me jump ahead to
16 a slide I have on that. And you're right. This is one
17 of maybe the 10 or 12 arguments they make. And so on
18 the left here is the original job slots that existed,
19 zero, one and two.

20 On the right we have job slot zero and one.
21 And what they're arguing is, well, when you're loading
22 jobs into the job slots. So, again, that's starting
23 over here on the left. Just putting them in the job
24 slots, not sending them to the shader core. That the
25 new products start at job slot zero, so you're

1 prioritizing job slot zero. Well, the fact is the
2 legacy products started at job slot zero and stepped
3 through. The only difference is the legacy products
4 would consider slot two. But they started at the same
5 spot. The code, the algorithm was the same. AMD could
6 have raised this below and didn't.

7 So based on the branch's precedent,
8 arguments that could have been raised below that
9 weren't, and had been unchanged, should not be
10 considered.

11 Taking that a step further, even if the
12 branch were to consider this, as I mentioned, this is
13 not as claim 20 requires, deciding which job gets
14 received by the shader core based on a priority. So
15 say, for example, you have a fragment job loaded in the
16 job slot zero first, and then you have a vertex job
17 loaded in job slot one second. That doesn't tell you
18 which of those two jobs is going to be received by the
19 shader core first. All you've done is load it into the
20 job manager.

21 What claim 20 requires is over here on the
22 left in these GPU functional on the right, and the GPU
23 functional units, there's execution ending in the
24 shader core sitting in them. That's what has to receive
25 the tasks based on a priority.

1 So simply loading into the job manager
2 doesn't tell you what was found there. And, in fact,
3 this relates to another argument AMD made during the
4 investigation based on loading jobs into a job slot. It
5 made a similar argument that based on this peek prio
6 function that's highlighted in red here, that you're
7 going to look at priority of the jobs, and load them
8 into the job manager. Again, just into these slots, not
9 in the shader core. They argued that below, and the ID
10 rejected it. It adopted because that doesn't tell you
11 what is being received by the shader core.

12 Now, here while I'm on this, just to mention
13 because I used the word priority, they had argued this
14 below, and below there were priorities associated with
15 these jobs coming in from the driver. But even then
16 with the priority, the ID rejected that. What we've
17 done now and gone a step further is we've removed any
18 of that. All jobs come in with the same medium label
19 you see here on the right, and they all sit in one cue.
20 So this function just sits there, and just pulls them
21 out in a FIFO order essentially.

22 So it's a similar argument. That's why I
23 went here. This is all happening on the wrong side of
24 the shader core. What goes into the job manager doesn't
25 dictate what gets received by the shader core, which is

1 what their requisite claim element is in claim 20.

2 MR. TERRILL: Thank you.

3 MR. HIGGINS: All right. So let me jump back
4 to claim 19 then, and on slide 27. And as I mentioned
5 earlier, this section will be a bit longer than for
6 claim 20. But when it comes down to it, all what the
7 branch needs to consider is also very straightforward.

8 We took what the commission found to
9 infringe, and we removed it from the accused products.
10 And as you'll see here, much of the dispute revolves
11 around issues that AMD is raising that are new, or that
12 contradict what the commission found.

13 And just briefly just to kind of set this
14 up, I will walk through exactly what AMD argued, and
15 what the ID and the commission adopted for, in
16 particular. And I'm going to do this again using AMD's
17 demonstratives and the arguments they made at the
18 hearing. That way there can be no argument by AMD
19 that we're mischaracterizing anything. These are their
20 own slides.

21 And because we're dealing with a dependent
22 claim here, I'm going to start and quickly go through
23 the independent claim, and some of the more relevant
24 elements. And first what we see here is the plurality
25 of engines. These were found met by the job slots we

1 just discussed in the legacy products that was job
2 slots zero, one and two. But as we know from the patch,
3 there's only two of those job slots now.

4 Then the next element is the scheduling
5 module. The ID found that the top, based on what AMD
6 argued, that this claim scheduling module was the Mali
7 driver. The device driver, as I mentioned, that's what
8 configures the GPU. That's what sends it work. That's
9 what tells it what to do. The GPU is worthless without
10 this driver.

11 And accordingly, this element, and you can
12 see at the bottom AMD called it out separately. Claim
13 15D scheduling module, Mali driver, it is this required
14 element of the claim. This is what we modified. This
15 is what removes what was found to infringe.

16 And then briefly for the shader core, which
17 I was just discussing in response to Mr. Terrill's
18 question, this is down inside the GPU way past the job
19 manager. And what the execution of -- what the shader
20 core was found to be in this case were these execution
21 engines that sit down inside the shader core of the
22 GPU. And where the job manager would be would be
23 somewhere up here at the top outside of this shader
24 core.

25 This it where the tasks need to be received

1 based on a priority in claim 20 that we were discussing
2 about. This is also where the claim tasks need to be
3 executed concurrently as claims 15 and then 19 would
4 require.

5 So for claim 19, which is topic of our
6 discussion today, this adds that two tasks that have to
7 be received by the shader core and executed
8 concurrently. One has to be a graphics processing task.
9 The other has to be a general compute task. And what we
10 saw AMD and its expert argued during the investigation
11 was what corresponds to these claim tasks were the ARM
12 jobs, vertex shader jobs, geometry shader jobs,
13 fragment shader jobs, and the compute job.

14 And their expert, Dr. Reinman, was very
15 clear what he was pointing to. Graphics processing
16 tasks like vertex shader jobs, geometry shader jobs,
17 fragment shader jobs. That's one part of what's
18 required by claim 19. The other half, general compute
19 tasks, like the compute job. So Dr. Reinman and AMD
20 separated out what is processed in the ARM GPUs, and
21 that's what corresponded to claim 19.

22 And this is not just what AMD's expert said.
23 It wasn't a mistake he made in the hearing. This is
24 what AMD adopted in its post hearing brief. They
25 argued the exact same thing, ARM jobs and tasks for

1 fragment graphics. Compute ARM jobs and tasks, those
2 are the general compute tasks of the claims. There
3 was -- it was crystal clear what AMD was arguing, and
4 that's what the ID and the commission ended up
5 adopting.

6 And just to show this graphically, this was
7 also what was used by AMD during the investigation, and
8 what they were showing here is we're down in the
9 execution engine at this part right here. And in the
10 execution engine, you had to show that there were a
11 fragment, graphics processing tasks executing at the
12 same time as a general compute task.

13 So the red and the blue boxes shown here in
14 the execution engine at the same time, that's what AMD
15 argued. That's what the ID adopted. But this is based
16 on the old device driver. This is exactly what we took
17 out and precluded from the accused products.

18 So that was AMD's argument. What did the ID
19 do? This is what the ID -- this was their entire
20 infringement analysis for claim 19. What they did was
21 they adopted what AMD argued. Let's look at it in a bit
22 more detail and zoom in.

23 And what we see here on the slide is the ID
24 specifically relied on the fact that the ARM jobs were
25 the claim tasks. So the ARM jobs we just discussed,

1 graphics processing versus the compute general compute
2 task. And, again, that's what AMD argued. That's what
3 Dr. Reinman testified to. Was very clear.

4 And then in the ID, because claim 19
5 requires a graphics processing task, and a general
6 compute task to be processed at the same time, the ID
7 relied on fragment jobs for graphics processing tasks.
8 Remember those are in job slot zero. And then general
9 compute tasks were the ARM compute jobs. Those are in
10 job slot one. Because they're in separate job slots,
11 they could potentially be in the shader core at the
12 same time. That was enough to find the overlapping
13 execution that the ID required for infringement of
14 claim 19.

15 Again, and here's the two job slots I just
16 referenced, fragment in job slot two. Zero compute in
17 job slot one. That was the crux of AMD's argument.
18 That's what the ID relied on to infringe. This is
19 exactly what we removed in the accused products. You
20 can no longer submit a fragment job from job slot zero
21 at the same time you submit a general compute job from
22 job slot one. It has simply been taken out of the
23 products.

24 And I want to demonstrate what that means
25 what we did. And I'm going to use AMD's slide again

1 here, the demonstrative that its expert used to
2 describe what it found to infringe, and it used this to
3 show what we did, and show why it no longer infringes.
4 So as you see first here, there are fragment shader
5 requests. Fragment shaders down in the execution
6 engine for processing. Because of that, and the patch
7 code, it blocks any general compute jobs from being
8 sent to the execution engine.

9 So what does that mean? That removes the
10 general compute task and a graphics processing task
11 from executing at the same time. We've blocked it.
12 Quote, what if the opposite occurs? What if there's a
13 compute task in the execution engine?

14 Well, the corollary of this is the blue
15 boxes, those are the compute jobs in the shader core.
16 The code now in the patch products blocks job slot
17 zero, so no fragment graphics processing jobs can be
18 submitted to the execution engine. Exactly what the ID
19 found to infringe based on AMD's arguments, we have
20 prevented it from ever occurring in the accused
21 products. It simply cannot happen.

22 Now, as I mentioned, the relevant aspect
23 here is what's in the shader core. And, again, those
24 are over here in the GPU functional units. But just to
25 ensure that there was no argument AMD could make that

1 there was any possibility that a compute job could
2 exist in the GPU shader core at the same time as a
3 graphics processing job, went a step further. What that
4 means is, let's say a compute job is submitted to job
5 slot one. And then the next job available by the driver
6 is a fragment job. Well, it's blocked. You can't even
7 load the fragment job into slot zero if a compute job
8 is there. So this compute job can go down into the GPU
9 shader core. And once its done and gets cleared out,
10 then and only then can you submit the fragment job to
11 job slot zero.

12 And the opposite would be true, too. Because
13 this fragment job is in job slot zero, a compute job if
14 it came up would be blocked until this fragment job
15 completed execution.

16 So there is no possibility, let alone any
17 configuration for the coprocessing of general compute
18 tasks and graphics processing tasks. We have configured
19 the products to do exactly what the -- the opposite of
20 what the ID found to infringe. We've configured them
21 not to infringe.

22 Then, again, similar to what I walked
23 through for claim 20, our support for this is at least
24 twofold, ARM release note that was submitted with the
25 code and the patches. And it says very plainly if

1 there is a compute job in the chain, the kernel driver
2 will not allow the job to run in parallel with fragment
3 jobs, and will enforce serialization.

4 At the bottom it ensures that if you have a
5 compute job in the shader core, you cannot have
6 graphics processing job in there at the same time, and
7 the flip side of that as well. Specifically, what the
8 ID required and the commission found to infringe we
9 have disabled in the products.

10 And this was clear in the code as well. All
11 of the experts reviewed this code. No dispute in the
12 functionality of what the patch does in the code.
13 Again, it's the same. There's a compute job. It can't
14 be submitted if there's execution happening on the
15 other job slot, which would be fragment and graphics
16 processing. So they are now mutually exclusive, and
17 they cannot exist in the shader core at the same time.

18 So to summarize, again, the commission and
19 the ID based their infringement finding on the fact
20 that an ARM fragment job claimed graphics processing
21 task, and an ARM compute job, the claimed general
22 compute task, could exist in the shader core at the
23 same time.

24 So what did our patch do? We specifically
25 prevented that from ever happening. They can't exist at

1 the same time. They can't even be loaded into the job
2 slots at the same time. So we have prevented, and we
3 have designed around claim 19 so these patch products
4 are no longer subject to the limited exclusion order,
5 because we have removed the expressed feature found by
6 the commission to infringe.

7 So that brings us to an issue I'm sure the
8 branch has seen occupy a lot of AMD's briefing here.
9 And this is called a performance enhancement patch. So
10 rather than save this for rebuttal, I want to just hit
11 this head on right here at the front.

12 And I want to start by explaining what the
13 patch does not do. This patch does not undue anything
14 that the ARM patches 04 and 05 that we just discussed.
15 Absolutely not. It does not permit an ARM compute job
16 to run in parallel with ARM fragment jobs. It does not
17 prevent -- permit any priority to exist in how jobs are
18 submitted and received by the shader core. And it also
19 does not infringe under the ID and the commission's
20 rulings.

21 So what is AMD's deal with this patch? Why
22 did they spend so much time arguing on it? Well, they
23 have to come up with a new argument, because they have
24 nothing based on what they argued below. The change in
25 their infringement argument, and a modification of the

1 commission opinion is what AMD is seeking. Because now
2 they want to say that these vertex shader jobs down
3 here in the orange box, they want to say that those are
4 now compute jobs. So what was a clear delineation
5 before, they have to try and argue, well, vertex shader
6 jobs, those can also be compute jobs now.

7 But that contradicts what they argued below.
8 Contradicts what the ID and the commission found, and
9 it's not appropriate here for consideration as part of
10 our ruling request on patches four and five.

11 And before we get into some of those details
12 on the patch, just a little bit more overview. Now, we
13 produced this patch as part of our initial disclosures
14 on the source code computer as part of the request. Why
15 did we do that when we were only seeking a ruling on
16 patches four and five? It's because this is something
17 we've considered putting in the products. We put it on
18 a demo board and provided it to AMD so they could see
19 how it worked. But we didn't want to be accused of
20 hiding the ball or concealing anything that has been
21 developed. Well, AMD made the accusations anyway.

22 But we are not seeking a ruling request on
23 this patch, because it doesn't undue what the previous
24 two patches do, which is remove specifically what was
25 found to infringe by the ID and the commission. Once

1 we've removed what was found to infringe, there's
2 nothing to decide here based on this performance
3 enhancement patch. It doesn't undue any of that.

4 All this patch does is take an initial part
5 of a vertex job, and run it as what's called a pilot
6 shader. Uses a vertex job. It performs vertex
7 processing on a vertex. There's nothing compute about
8 this. It's not a general compute job. It's a vertex job
9 running a vertex shader.

10 So AMD tries to come up with a new
11 definition. And we'll get to that in a second. But
12 trying to now construe something different as to what a
13 general compute task means, but there are many, many
14 problems with this that we'll get to.

15 MR. TERRILL: So I have a question about the
16 previous slide.

17 MR. HIGGINS: Sure.

18 MR. TERRILL: So it's my understanding that
19 this performance enhancement patch designates, I guess,
20 the pilot shader program as a vertex job. But under
21 the legacy driver, the exact same pilot shader program
22 was designated as a general compute task. Was it
23 erroneously designated as a general compute task in the
24 legacy product such that that error was why it was
25 changed to it being a vertex task, or is it just a

1 situation where Realtek -- it's Realtek's position that
2 any program could be any task, and up to the role of
3 the engineer to decide?

4 MR. HIGGINS: More of the former than the
5 latter. And I think I can give you these slides, and
6 I'll address it, but let me answer your question up
7 front, and then I'll go through the slides in some
8 detail.

9 So what the initial pilot shader was doing
10 was being run as a compute job within the ARM GPU. But
11 what it was operating on was a vertex, multiple
12 vertices. So it was performing a calculation across
13 multiple vertices. The output of that was some position
14 or calculations then used by the remainder of the
15 vertex shader.

16 Why ARM was running that as a compute
17 shader, more for some kind of efficiency in the way
18 they had done it, it required more steps. But now
19 running it as a vertex shader, it is calling a vertex
20 job, calling the vertex shader program, and doing the
21 calculation on the same vertices producing what should
22 be the same output, so doing it a slightly different
23 way by using a vertex shader.

24 But that is the -- kind of the crux of what
25 was changed. And maybe now if I get into this, I think

1 I can provide more detail, but stop me at any time if
2 you have any more questions based on that. Because I
3 think what's important here is this was always a vertex
4 shader. The initial program written in OpenGL shader
5 language. And when that comes into the driver, it's
6 compiled. So you still have the original vertex shader
7 program.

8 What ARM initially did was when it broke it
9 into two parts, because sometimes you could have a
10 calculation that can be performed on maybe 100 vertices
11 at a time because it's common. They broke it out, and
12 they call it a compute job with some program that they
13 had designed, and ran a calculation on that. That
14 output would then be used by the vertex shader, so
15 you're still doing a graphics operation. It's providing
16 the graphics output for the vertex.

17 But what ARM has done now is changed that.
18 It no longer calls this compute shader. Whatever
19 efficiency they gained by that before, removed it so
20 it's now a vertex shader. So the two binaries that the
21 compiler will produce are two vertex shader programs.

22 And kind of it maybe it goes further to your
23 question that this wasn't just a name change that this
24 made. The change was made in the code. We produced this
25 code, and showed what was changed. So we're not just

1 calling a compute shader a vertex shader. There is a
2 separate code path for a vertex shader compared to the
3 compute shader.

4 Again, there's job descriptors that are
5 used. You call it vertex job now. So it's more than a
6 name change. We've changed what's been done, and it's
7 changing the actual program that's being run. And,
8 again, not just a name change.

9 The job type is set to a vertex job, and it
10 is run in a vertex shader. It does not run in a
11 compute shader any more. So it didn't just change the
12 name to compute job. It's no longer calling a compute
13 shader. It's calling a vertex shader. It's a
14 completely different type shader program from OpenGL.
15 It's still operating on a vertex which, you know, going
16 back to the original legacy code performing the
17 operation on a vertex.

18 All along, this was doing graphics
19 processing as part of the original vertex shader. Now
20 it's been made more express. It's using the vertex
21 shader. It's clearly a graphics operation; part of the
22 graphics pipeline.

23 And to give a little more detail behind
24 that, this is -- it's not just the document and the
25 code that ARM submitted now. This has been the position

1 for a long time. This is an ARM patent that AMD and its
2 expert, Dr. Reinman, cited to. And what we see in this
3 ARM patent that they contend describes pilot shaders, a
4 pilot shader is always part of the original shader
5 program. It's just an efficient way to do calculation
6 on maybe more than one vertex at a time.

7 And to put it another way, if a graphics
8 operation you do on one vertex is a graphic -- if an
9 operation done on one vertex is a graphics operation,
10 it can do the same operation on two vertices. It
11 doesn't undue the fact that it's still a graphics
12 operation. That's essentially what a pilot shader is.

13 And even here in the patent when they talk
14 about it, it's talking about graphics. There's no
15 dispute that it is a graphics operation. It is still
16 the original vertex shader program that it is being run
17 a part of.

18 And where there's still any doubt, if you
19 look at what AMD cited in the '381 patent, it talks
20 about certain preprocessing tasks that can occur for a
21 task. So if you look at this pilot shader as doing
22 some preprocessing for a vertex job, well, that's
23 allowable by a '381 patent. It doesn't transform it in
24 to something else. It's still a vertex job used in the
25 vertex pipeline.

1 And here the citation that AMD provided in
2 its post-trial brief identifies shader programs
3 associated with the graphics process. So here it's a
4 vertex shader program. It's always been the vertex
5 shader job that's associated with graphics processing.

6 And the '381 patent provides a bit more
7 description on that as well. Again, shader programs
8 such as geometry shader, pixel shader, vertex shader.
9 It's these shaders that handle graphics data. These are
10 graphics processing. Nothing about the performance
11 enhancement patch changes what that did. It takes some
12 flexibility away that ARM may have thought it had when
13 it did that before. But it correctly characterizes them
14 and uses a vertex shader job.

15 So based on that, and the fact that these
16 pilot shaders don't really change anything, AMD had to
17 come up with something new. And this is, again, just a
18 recap before I get into their new construction of what
19 was argued. Again, compute jobs or general compute
20 tasks.

21 Again, Dr. Reinman second has to be general
22 compute, the compute shader. And in these pilot jobs
23 they're running the vertex shader. So there's no
24 dispute that they are graphics.

25 So then -- and then, again, just to be clear

1 what the ID found, compute, claim general compute
2 tasks. So AMD argued it. The ID adopted it. So this is
3 what we see from AMD now. Their new construction, which
4 is a bit circular, that a general compute task is a
5 task that performs general computations.

6 The only source of this new found circular
7 definition was their expert, Dr. Reinman. So this was
8 written into his declaration, and then AMD copied it
9 into their brief. But when we got to Dr. Reinman's
10 deposition, he was specifically asked, Did you define
11 general compute tasks as the task that performs general
12 computations? And he didn't attempt to come up with a
13 definition. He would not agree that this is how you
14 define what a general compute task is.

15 But there was more to what Dr. Reinman tried
16 to testify to in AMD's argument. AMD tried to say that,
17 well, you have general computations, and you have
18 mathematical computations. You saw in the slide I had
19 before in the patent invention mathematical
20 computations was talking about graphics.

21 So we asked Dr. Reinman in his deposition,
22 what is the distinction between general computations in
23 this new definition, and mathematical computations?
24 Essentially there's nothing. They could mean the exact
25 same thing. The computation is a computation. When you

1 get to the ALU, that small block inside of the
2 execution engine, it's doing a computation at its most
3 basic level. Can add, multiply, something else. There's
4 no differentiation between general computation and
5 mathematical computations at that level. It's a
6 distinction without a difference.

7 AMD is trying to come up with something new.
8 But as they're doing this, they're contradicting
9 themselves and contradicting what the commission and ID
10 found.

11 And to be clear, it not only contradicts
12 what the ID and commission found, it contradicts what
13 AMD argued below to try and avoid invalidity
14 specifically to claim 19.

15 The only thing missing from Jiao on claim 19
16 was the fact that AMD argued it did not process or
17 disclose specifically general compute tasks. And it
18 argued this, even though Jiao had processing units that
19 had to perform general computation capabilities,
20 because it had ALUs in there.

21 But AMD's position when it had to argue for
22 invalidity -- for validity done in the investigation
23 was that general computations isn't enough. You need to
24 look to more. In other words, AMD was pointing to the
25 fact that Jiao didn't disclose using a compute shader.

1 It used a vertex shader, and a fragment shader.
2 Essentially what the performance enhancement patch
3 does. It's a vertex shader.

4 AMD can't have it both ways. And they are
5 still making the same argument today. There's an IPR
6 pending on the '381 patent based on Jiao and another
7 reference. They are again arguing there is no
8 disclosure of a general compute task.

9 And what do we see in Jiao? It has a vertex
10 shader. And in the vertex shader can do a basic
11 calculation of the transformation taking one value and
12 changing it. But apparently to AMD when it's talking
13 about validity, those types of general computations
14 don't count, nor does the fact that a vertex shader is
15 run as general computes, because it's not. It's
16 graphics. It doesn't work for invalidity. It can't work
17 for infringement.

18 And then I don't want to spend a lot of time
19 on this. I'm not sure if AMD is still going to push
20 this argument. But they made passive reference to the
21 fact that a vertex shader can be used to perform physic
22 simulations, and that's somehow not graphics
23 processing. But when you actually look at the document
24 that its expert cited to, and he didn't put any of this
25 into his declaration, the physics simulation, it's

1 operating on vertices. Position, velocity, vector,
2 colors. It's outputting these vertex shader optics
3 called bearings, which is what you have in a graphics
4 pipeline, to a fragment shader.

5 And here you have the code for position
6 velocity particle time. So their expert tried to call
7 it a particle instead of a vertex. It's a position on
8 a stream. It's a vertex running a vertex shader.
9 There's nothing here as part of the graphics pipeline.

10 So I'll pause there before I get into the
11 remedy, and if there were any additional questions on
12 claim 20.

13 MS. WU: Yes, hi. I have -- I have a quick
14 question. So how do you differentiate general compute
15 tasks from graphics processing tasks? So, for example,
16 according to page 42 of the commission opinion, it said
17 that other compute tasks include descriptors, pointers,
18 JASID, shader program start, attributes and shader
19 constants. So how would you differentiate those kind of
20 tasks from graphics processing tasks?

21 MR. HIGGINS: Before I -- so I didn't hear
22 what page you said from the commission opinion so I can
23 reference that.

24 MS. WU: Sure. It's page 42.

25 MR. HIGGINS: And maybe I didn't hear your

1 question correctly. You asked how do we differentiate
2 general compute tasks from other tasks, and I believe
3 you were referencing the descriptors, program counter
4 and pointers?

5 MS. WU: Yes. From my understanding, it
6 appears that those are considered compute tasks, which
7 are interchangeable, it seems, with general compute
8 tasks.

9 MR. HIGGINS: Okay. Yeah, to back up, I
10 think those are not tasks. So for each job in the ARM
11 architecture, there are certain -- there's certain
12 information that goes along with it. So a job will have
13 a job descriptor. It will have the JASID, a program
14 start location. Those aren't tasks themselves. Those
15 are what are used to identify a task.

16 So let me -- here back on slide 46. Do you
17 see where it says job descriptor? It may be hard to
18 see. It's right here.

19 MS. WU: Yes.

20 MR. HIGGINS: That's just one example. And
21 here there's others to the vertex pointer side. That's
22 information that can be used that's part of the job. So
23 when it's loaded into the job manager, that information
24 goes along with the job. So that's not a job itself.
25 But, for example, when it says, you know, the shader

1 program start. That's going to point for a general
2 compute task to a compute shader start location.

3 For a graphics processing task, that shader
4 program start will point to a vertex shader starter, or
5 a fragment shader start location. So while this
6 information is not a task, it can be used to identify
7 what the task is with the type of job by identifying
8 the shader program that is being run with, and the data
9 that it's being operated on. For example, a vertex or
10 a fragment.

11 MS. WU: I see. Thank you.

12 MR. HIGGINS: If there are no questions on
13 that, I'll jump quickly into the remedy portion. I just
14 want to address some of the arguments that AMD raises
15 on remedy. But the fact is once the branch will
16 determine that there are two patches that we have
17 applied that disable what was found to infringe, there
18 should be not much of an issue regarding the remedy.

19 AMD's arguments fall in the scope of a
20 hypothetical scenario where as, we discussed earlier,
21 one of Realtek's customers was trying to undue what was
22 in the patch code. As I mentioned, not only is it pure
23 hypothetical, it's just wrong.

24 And so Realtek SOC's and other products
25 without the software, again, they're not within the

1 scope of the LEO. They can't do anything. They're not
2 configured to infringe. They were not found to
3 infringe.

4 And then Realtek demo boards that we
5 maintain possession of, those would be covered by a
6 certification, and there's no dispute raised by AMD
7 that we're going to change the code once we carry it
8 into the US.

9 The Realtek products that have a GPU, the
10 driver code, and they're delivered to third parties,
11 such as a demo board, again, that driver's in binary
12 code format splashed onto the product. There's no
13 evidence it could be changed without hacking. So that'd
14 be outside the scope of the use of the products, and
15 there's no evidence that any customer of Realtek's
16 going to do that, or be permitted to do that.

17 And then finally you have the Realtek
18 products that are delivered to four entities other than
19 TCL. So TCL was the only respondent customer of
20 Realtek. This would not be covered by the LEO. But,
21 again, a certification provision could ensure that the
22 modified driver code is used regardless.

23 And then finally, as I mentioned before, you
24 have the program as a boot key. We don't think it's
25 needed here. It's an additional measure we could

1 implement if that became necessary. But that would just
2 prevent the booting of the entire chip if someone tried
3 to flash -- to reflash, delete, modify to the old
4 driver code. So it's just an additional measure we
5 don't think is necessary, but we could do it if we
6 needed to.

7 That takes us to the last section, and I'll
8 walk through these quickly. I know I have about a half
9 hour left, but I don't think I'll need to spend all
10 that time on this. But once the branch determines that
11 the two ARM patches disable what we did, all of these
12 arguments that AMD is raising in what we call this
13 kitchen sink, they are directly contrary to what they
14 argued below, or their arguments they could have raised
15 below and they didn't. Or they're now just directly
16 contradicting what the commission found. Either way,
17 the branch, in our view, does not need to consider
18 these arguments.

19 Based on its precedent, the CBP here and the
20 branch is not going to refuse entry on the basis of
21 features that were present on the accused devices
22 during the investigation that were not both
23 specifically accused of infringement, and found to
24 satisfy the limitations in the claims.

25 So, in other words, if AMD could have made

1 an argument before based on something in the accused
2 products that infringed, and it didn't, it can't make
3 that argument now. And the same goes for its argument
4 that it's trying to make that directly contradicted and
5 argued below, or argued to the commission. Those
6 aren't proper here. The branch's ruling found by the
7 findings of the commission, that's the relevant
8 inquiry. All of these new arguments that AMD's raising
9 are simply not relevant.

10 And there's two buckets here that I want to
11 talk about. The first one goes to the scheduling
12 module that I mentioned earlier. Based on the branch's
13 findings in the Apple and Diebold case, the
14 modification we made to the driver disables the
15 functionality that was found to infringe in the accused
16 products. So regardless of whether the scheduling
17 module is an element of the claim or not, it is. The
18 commission and ID specifically found that it was. It
19 doesn't matter. We've effectively disabled what's in
20 the accused products. This is nothing more than a red
21 herring raised by AMD.

22 And then I'll get to claim 20. We've
23 already covered two of these for Mr. Terrill's question
24 earlier, but there's just a lot of new argument that
25 AMD's raising. Because we've removed what was found to

infringe, and they're trying come up with something new.

So I'll start with the scheduling module.

And where I want to start here is the beginning of this claim element recites the plurality of engines configured to receive from a schedule module. Now, shown in the top box is in blue here, this is what the ruling was on the left from the commission and the ID. Configured to requires more than mere capability. Has to be a design, an intention, something to achieve the stated effects.

The precedent of the branch is configured to means in example program two. Not nearly capable performing a task. And actually if you look at AMD's IPR position that's still arguing now, it agrees with that. It's not met by mere capability.

But what does AMD try here? It has a new definition that it says requires the capability. That's not configured to. That's capable of. It contradicts what the commission and the ID found. It contradicts what AMD argued below. The only reason AMD is trying this now is because it doesn't have to worry about validity in this proceeding. That's not a reason that can go against what was argued below what was found by the commission. It's simply wrong.

1 So even if we get to the scheduling module
2 element on that basis alone, AMD's argument fails, even
3 if it's considered. But there are additional reasons
4 here it still fails. And that's based on AMD's new
5 argument now that the scheduling module is not limited.
6 But the ID found it was. Commission adopted that.

7 AMD is arguing in an IPR that it's required
8 limitation, and it used this as a required limitation
9 to argue over the prior art in the investigation. If
10 there were any doubt, the ID specifically stated AMD
11 acknowledges that from a scheduling module associated
12 with the second processing unit is limited. It can't
13 undue that now just because it doesn't like that
14 position any more, and because it doesn't have to worry
15 about validity.

16 That was the commission and the ID ruling,
17 it can't be contradicted. Then again, this is the slide
18 I showed earlier. They specifically pointed to the Mali
19 driver as the scheduling module. It was the claimed
20 scheduling module. Configures the GPU. It was
21 required to satisfy claim 15.

22 In here this is what the ID found with
23 respect to AMD's prior art argument. There was a prior
24 art reference Stuttard. And the only reason it didn't
25 anticipate claim 15, because AMD argued that it did not

1 have the scheduling module associated with the second
2 processing unit. So if it was required for validity,
3 and they use that to uphold the validity of their claim
4 here based on Stuttard, it definitely is required for
5 infringement, and that's why the ID found that it was.

6 And why did the ID find that? Because it
7 was specifically added during prosecution to overcome
8 the prior art. Exact limitations, scheduling module
9 associated with the second processing unit. There was
10 no way to undue this now.

11 And just as in I believe it may have been
12 the Apple decision, this is the exact element that we
13 modified in the products to disable and prevent what
14 was found to infringe. It was the element added to
15 overcome the prior art. They can't now try and argue
16 it's not required based on some new theories they're
17 coming up with because they don't have to worry about
18 validity.

19 And again, they're still arguing this in the
20 IPR, all claims or prior scheduling module. There's
21 nothing different about this proceeding that would
22 allow AMD to come up with a whole new argument that
23 contradicts everything else it's ever argued, and what
24 the commission has found. There's simply no merit to
25 this position.

1 I'll quickly step through claim 20 now with
2 the remaining time I have, and I'll address a lot of
3 this on rebuttal to the extent they argue it. We've
4 already covered the same value being set by the
5 priority. There is no priority any more. And what that
6 leads to is when all these jobs are received by or sent
7 by the driver, they come in in order.

8 So when you take away the priority and
9 different cues, you're left with FIFO buffer. And a
10 FIFO buffer is first in, first out. Jobs come in in
11 this order. They go out in that order. That's not a
12 priority. In fact, the '381 patent talks about a FIFO
13 buffer, and doesn't call it priority, because it's not.
14 It's simply sequential. You come in, you go out. There
15 is no priority based on a FIFO.

16 MR. TERRILL: Can I chime in for a second?

17 MR. HIGGINS: Sure.

18 MR. TERRILL: So I think one of the
19 arguments is that the redesign, because it prevents
20 a -- what I want to say is if there's like a -- it
21 prevents certain tasks from being in the job manager
22 while other tasks are in the job manager. Would --
23 let's say something that is being cued to go into the
24 job manager. I think the argument is that if, for
25 example, if you had a compute task in the job manager,

1 you would prioritize non compute tasks. You would
2 prioritize compute tasks, because the job manager is
3 incapable, or I guess job slot one isn't incapable of
4 having a compute, or I guess the job managers don't
5 have the ability to have two different jobs of two
6 different types, non compute and compute. And so in
7 your mind, does that create any priority or no and, if
8 so, why?

9 MR. HIGGINS: Absolutely not. And I believe
10 this is what you're referring to here. So, for example,
11 in the job manager now if you have a fragment job
12 loaded in job slot zero, you can't load a compute job
13 in job slot one. And then the corollary of that is
14 true. If you have a compute job loaded into job slot
15 one, you can't load the fragment job.

16 So AMD argued, well, this gives the compute
17 job in this example here exclusive access to the shader
18 core, and that means it has priority. But if you look
19 at the '381 patent, it talks about right here while
20 enabling one of the tasks to fully consume the
21 resources of the GPU.

22 Okay. Let's say that's what happens here.
23 But there are different ways that can happen based on
24 demand, priority, or preset limits. So you're not
25 giving priority here. The compute job came into the job

1 slot. More of a limit. And I guess the point here to
2 make is the mere fact that just the compute job is in
3 the shader core by itself, the '381 patent tells us
4 that does not mean that's based on a priority. There
5 are other factors that can go in.

6 And what we know based on the patched code
7 is the way it gets into the job manager, there is no
8 priority. We remove the priority. The compute job and
9 the fragment job are coming in with the same medium
10 priority settings. There is no priority in how they get
11 loaded. They come in, they get into the job slot. And
12 once they're there, and it gets loaded in the shader
13 core, the mere fact that you can't have a fragment job
14 in at the same time, that's not a priority. That's just
15 a limit that's set, or you're blocking the concurrent
16 access. But the '381 patent, and the way it describes
17 that it definitely is not a priority.

18 MR. TERRILL: Thank you for that.

19 MR. HIGGINS: I think that still leaves us
20 with about five or six new arguments that AMD's raised.
21 Let me quickly jump through these.

22 The other argument here, somewhat related to
23 the scenario you raised. It's a different argument.
24 They say there's increased throughput for fragment jobs
25 with the patch code. And that's because fragment jobs

1 sit in job slot zero by themselves. And job slot one
2 holds all of these other jobs. Well, as you can see in
3 the legacy products, job slot two held a bunch of other
4 jobs. Job slot one still held all these other jobs.
5 They could have raised this argument before. They
6 didn't. So they can't raise it now.

7 And to be honest, this still is not a
8 priority. You can only submit one job from a job slot
9 at a time. It's not giving priority to fragment jobs.
10 The way these jobs come into the GPU, it just
11 ping-pongs back and forth. There is no priority given
12 to fragment jobs. And also wouldn't make any sense,
13 because you can't do any fragment shading until you've
14 done vertex shading.

15 So giving fragments some kind of increased
16 throughput at the start, it just doesn't make any
17 sense, and it's another argument that -- it was in the
18 old products. They didn't raise it. Can't be raised
19 now.

20 Another argument they raised, and this is
21 based on application level requests. So do you recall
22 back to our diagram, we have the driver code here, and
23 then you have the job manager. Shader core's down
24 here. We're talking about the application program
25 sitting way up here. Further removed. So their argument

1 is, well, if a programmer says we need to do this
2 vertex job, or fragment job first, give it priority in
3 the market. Well, it doesn't work, because the patch
4 and the driver ignores any of that. It takes what is
5 set in the patch, and it marks everything coming in
6 with that same medium priority.

7 And then specifically I jump back to slide
8 72. Right here we see it's in the release note. All
9 items have the same priority value. Any request from
10 applications for a different priority will be ignored.
11 So specifically what AMD is arguing, it just doesn't
12 work.

13 The other aspect of this, and Dr. Reinman
14 recognized this, he could have done this in the
15 investigation. All he did was write his own program and
16 try to force this. He didn't do it. He didn't confirm
17 whether this was a priority. It wasn't a priority.

18 They didn't argue it in the investigation.
19 It's not something they can argue now. Again, this is
20 just adding to their long laundry list of arguments
21 that are completely irrelevant here because they didn't
22 raise them before.

23 I think this one is one of the worst. They
24 argued that because the job manager can soft stop jobs,
25 and that means if a job is running in the GPU, the job

1 manager can stop it and say, okay. We need to send
2 another job in there. And they're arguing this based on
3 the fact that, and this is a quote from Dr. Reinman's
4 declaration, You can let higher priority jobs run, and
5 then it repeated that. But that's what the patch code
6 gets rid of. There are no high priority jobs.
7 Everything is the same. This is gone.

8 And you see the rest of this, the other
9 option is to give each application a fair share. Giving
10 two things or three things a fair share is the exact
11 opposite of a priority. If you're trying to give equal
12 access, that is not giving anything a priority. And the
13 other fact of the matter is this was in the legacy
14 products. They didn't argue it, so they can't argue it
15 now. Another argument that falls into that same bucket.

16 And then the last one here, this one clearly
17 falls into the grouping of something they could have
18 argued below, and did not. And they even say these
19 three things that they reference here that they say are
20 priority, they were maintained in the unmodified
21 hardware of the Mali GPUs. Otherwise, they were there
22 during the investigation. They didn't argue it. And why
23 didn't they argue them? Because there's no priority.

24 They're all about ensuring fairness. For
25 example, the end point allocation, it's a round robin

1 selection. That's a fair selection process. The [REDACTED]
2 [REDACTED], even Dr. Reinman characterized it as
3 maintaining fairness. In the [REDACTED], ping-pong.
4 It's just going back and forth. It's all fair
5 selection.

6 It's not a priority. That's why they didn't
7 argue it below. They shouldn't be able to argue it now
8 either. This is just adding to their laundry list of
9 new infringement arguments that the branch can just
10 summarily deny and focus on what was argued below, and
11 how our patches remove exactly what the ID found to
12 infringe.

13 Unless there are any further questions, I'll
14 save the rest of what I have for rebuttal, but I thank
15 you for the time.

16 MR. COOK: Thank you for that. So at this
17 time we'll take a 10 minute break.

18 MR. TERRILL: Aaron, just before the break,
19 I just have a question or two.

20 MR. COOK: Please go ahead.

21 MR. TERRILL: Thank you for that. As I
22 understand it, AMD is making an argument that the claim
23 19, claim 20 as they depend on claim 15 are directed to
24 the hardware rather than the software, and there've
25 been no modifications to the hardware. I think you

1 alluded to this argument earlier, and might have said
2 that in AMD's sur-reply it's an argument that they
3 didn't focus on. But I'm not sure I saw a Realtek in
4 its presentation address those arguments. So I wanted
5 to give you an opportunity to do so now.

6 MR. HIGGINS: Oh, sure. And there were two,
7 two places in which I address this argument. The first
8 was back on slide 15. And the point here was that
9 regardless of what AMD's position is on that,
10 regardless of whether the scheduling module, which is
11 what they're arguing.

12 They're saying it wasn't required, when it
13 was directed to hardware. They read that on the
14 driver. The point is that doesn't matter, because if
15 you look back at even the Apple decisions, that Diebold
16 decision, we have taken the driver that's on the
17 products, and modified that to prelude the hardware
18 from ever accessing those features. It can't be done.

19 We have effectively disabled and removed
20 what was found to infringe by the ID and the commission
21 by modifying the driver code. And we did that in a way
22 that went further than what was done in Apple. I mean,
23 in the Apple case, change wasn't made, you know, to the
24 hardware of the watch. It was a setting made on the
25 phone that when you paired the watch with the phone it

1 would change a bit in the watch, and you couldn't
2 access that hardware feature any more.

3 So we've gone a step further than that. And
4 it's similar to the Diebold case.

5 And the other part of where I address
6 this --

7 MR. TERRILL: Before you go on to the second
8 part, because I see -- I understand there's an argument
9 you're making about the configured to construction at
10 the commission, and whether that requires something to
11 be programmed to perform, or designed to perform a
12 certain task, or whether it just requires capability.
13 But I'm not sure if I understand the connection between
14 the claim construction argument about configure to, and
15 whether the other claim limitations require -- whether
16 the other claim limitations read on to features of the
17 legacy products that were hardware components or
18 software components. I guess my question is could you
19 explain a little bit more about the connection between
20 those two aspects, and as that impacts on your non
21 infringement argument?

22 MR. HIGGINS: Sure. And that would take us
23 back to this section of the presentation. And this
24 is -- this relates to the scheduling module. So the
25 claim recites a scheduling module associated with a

1 second processing unit. And essentially what AMD is
2 arguing now is it's not limiting. It's in claim 20.
3 Plurality of engines configured to receive from a
4 scheduling module associated with a second processing
5 unit. And that's where this all comes into play.
6 Because this scheduling module that AMD acknowledged
7 was limiting below, and they argued this to overcome
8 the prior art, this is the element that was read on the
9 driver.

10 So the Mali driver was -- it corresponded to
11 the claim scheduling module that was required by claim
12 15. So it was this element that -- the driver is not
13 hardware. It's not in the RTL. So this is where that
14 comes into play. And this is what they read on the
15 driver, the scheduling module. It sends the tasks to
16 the plurality of engines which then gets the shader
17 core. So it was required to be part of the claim.

18 What they're arguing now is it's not
19 required, and the claim only directed the hardware.
20 But that contradicts the arguments they made to get
21 over the prior art where they said this had to be shown
22 in the prior art.

23 This Stuttard reference did not have a
24 scheduling module. And the reason the ID ruled that way
25 that it was limiting was because this limitation was

1 added to the claim to overcome prior art. What AMD is
2 trying to say now is, well, you can get rid of this.
3 Just look at the memory bus shader core first
4 processing it. That's all you need.

5 MR. TERRILL: You left out one particular
6 part. And I think it is the focus of AMD's arguments,
7 which is the plurality of engines that's configured to.
8 And I guess as a starting point, does Realtek agree
9 with AMD that at least as of the various claim
10 limitations, at least a plurality of engines does read
11 onto a hardware feature?

12 MR. HIGGINS: So the way that issue came out
13 in the case was during the investigation it was ruled
14 that it is not required to be either hardware or
15 software. What they are reading it on in this case are
16 the job slots, and the job manager, which is in the
17 GPU.

18 What the claim construction was in the case,
19 it wasn't required to be one or the other. There was an
20 argument made that they had to be virtual engines, and
21 that was software only. And the ALJ rejected that, and
22 said there's no requirement it has to be either one.
23 But that fact alone does not change that the scheduling
24 module associated with second processing unit, this was
25 read on the device driver, which is not in the

1 hardware. So I think as they're trying to couch this in
2 terms of plurality of engines, the relevant limitation
3 here is this, which they read on the Mali driver.

4 MR. TERRILL: Thank you.

5 MR. COOK: Are there any other questions
6 from the EOE branch?

7 MS. WU: Not at this time. Thank you.

8 MR. TERRILL: My apologies. One last
9 question from me. And this is -- this question is
10 about the application of the EOE branch precedent. And
11 I don't have the slide in front of me, but I believe
12 you have a slide that references a number of the
13 rulings about when a feature -- that's the exact one,
14 slide 61 that I'm looking at. An aspect of the
15 application of these rulings is that the -- other than
16 the removal of the legacy feature found to infringe, or
17 found to satisfy a particular claim limitation, that
18 the redesign device operates in the identical fashion
19 other than, of course, for the removal of that
20 particular feature.

21 Is there evidence in the record that you can
22 point to that the Realtek articles at issue in this
23 ruling request perform identically to the legacy
24 products but for, of course, the removal of the
25 particular feature at issue, such that this EOE branch

1 precedent should be applied?

2 MR. HIGGINS: Sure. Yeah, and I think it
3 would probably take it on -- probably goes under
4 argument by argument basis, and they've made a lot of
5 them. For example, if you go to this argument they're
6 raising on the priority. These three components in the
7 RTL are unchanged. What they do in the RTL has not
8 changed. They get an input, and it either ping-pongs
9 back and forth, or does a round robin selection, and it
10 sends it on. There was no modification made here.
11 Again, we made the modification at the driver that
12 disabled the use of job slot two, and also removed the
13 priority that was done.

14 Another example would be here. This is the
15 slide we talked about earlier when I prefaced one of
16 your earlier questions. This function here, this peek
17 prio, that existed in the legacy products. It operates
18 the exact same way as it did before. This was rejected
19 by the ID, even when there were priority levels.

20 All we did now was remove the priority.
21 They're all medium. So even though this function is
22 going through and looking at, okay, is there anything
23 in the low cue, anything in the medium cue, anything in
24 the high cue. There is never anything in the lower and
25 high cues. It's still functioning the same. It's just

1 everything is the same medium value, and there's no
2 priority.

3 MR. TERRILL: Thank you.

4 MR. HIGGINS: And I could walk through -- I
5 could walk through more of these, or I can address them
6 on rebuttal, but the story is the same. The only
7 argument they have is that we prevented this concurrent
8 processing or remove priority. But all these functions
9 are pointing to those weren't changed. They operated
10 the same way. And if there was any argument to make
11 before, they could have made it before, and that
12 argument would have changed.

13 MR. TERRILL: Thank you. Those are my only
14 questions at this time.

15 MR. COOK: Thank you. So at this time, I'm
16 going to go ahead and take a 10 minute break. It is
17 2:33. So come back at 2:45.

18 (Break at 2:34 p.m.)

19 (Back on the record at 2:45 p.m.)

20 MR. COOK: So it looks like it's 2:45. Is
21 counsel for Realtek back on the call?

22 MR. HIGGINS: Yes.

23 MR. COOK: Thank you for that. Is counsel
24 for AMD back on the call?

25 MR. RIZK: Yes, we are.

1 MR. COOK: Thank you for that. So at this
2 time, AMD will have one hour and 30 minutes to present.
3 So AMDm please proceed.

4 ORAL DISCUSSION BY MR. RIZK:

5 MR. RIZK: Thank you, Mr. Cook. Adam Rizk
6 again on behalf of AMD. Before jumping in, and just to
7 provide a brief overview of AMD, and some background
8 about the patented technology that's at issue here.

9 AMD, I think as many of us know, is an
10 American innovator in computing and graphics
11 processing. Has a global footprint in 31 countries
12 with a significant concentration in the United States
13 where it has its headquarters in Santa Clara,
14 California.

15 It has a long history of innovation in the
16 CPU and the GPU industry. Central processing unit,
17 graphics processing industry. In the CPU space, it was
18 the first company to break the one gigahertz barrier
19 with its Athlon multiport CPU.

20 In 2005, Microsoft selected AMD, and its
21 novel unified shader based GPU designs, and powers Xbox
22 gaming console that eventually became the standard that
23 all graphics processing unit companies around the world
24 have adopted, AMD's unified shader design.

25 In 2009, AMD invented its asynchronous

1 compute GPU architecture, which is also sometimes
2 referred to as asynchronous task dispatch. That's the
3 phrase that's used in the patent. And that
4 asynchronous compute architecture is what ultimately
5 led to the '381 patent.

6 Turning to the '381 patent, as I mentioned,
7 it's directed towards a GPU architecture that enables
8 asynchronous task dispatch. That's at least the title
9 of the patent. And as you can see from the title, it
10 is the processing unit, the actual physical hardware
11 that performs the enabling. We'll get into details of
12 how the patent discloses that the hardware within the
13 processing unit enables the asynchronous task dispatch
14 based upon its unique architecture.

15 But among other things at a high level, this
16 asynchronous task dispatch technology architecture,
17 allows CPUs to offload general compute tasks to the
18 GPU, the concurrent execution with graphics processing
19 tasks, which generally was not possible prior to the
20 '381 patent and invention.

21 So what ended up happening is that by --
22 back to the timeline slot, by 2013, every major gaming
23 console eventually adopted AMD's GPU design. And that
24 wasn't coincidental with the timeline for the
25 development of the '381 patent technology. The GPU

1 architecture that was covered by the '381 patent and
2 its configuration to enable asynchronous compute was
3 one of the key architectural features that led to the
4 wide scale adoption of AMD GPUs, and just about every
5 major gaming console in 2013, as you can see on this
6 slide.

7 What you can see on the left is an article
8 about an individual by the name of Mr. Mark Cerny who
9 was the chief architect of Sony's PlayStation 4. And
10 he provided a talk regarding AMD's new GPU
11 architecture, and he specifically credited AMD's
12 asynchronous compute technology, and its novel
13 architecture related to what was called its
14 asynchronous compute engines, the engines, the
15 plurality of engines that we see in the claim, as being
16 the key, one of the key enablers for that underlying
17 technology. He was talking about the hardware. Ability
18 of the hardware to then allow the software to utilize
19 its functionality.

20 So when we get into the details of what was
21 the key features of the claims of the claim in the
22 patented invention that enabled its success in the
23 industry, and the adoption as the gold standard within
24 the GPU industry, claim 15 says it's all.

25 There's a memory, first processing unit, and

1 a bus coupled to the memory in the first processing
2 unit. And that first processing unit is what's referred
3 to as a GPU. It's more generic in this claim. But as
4 it's described in the patent, the first processing unit
5 is the GPU. And the recitation, there's a separate
6 second processing unit which would be a CPU.

7 And there are two basic components that are
8 part of the first process. It says wherein the first
9 processing unit comprises. It is a plurality of
10 engines, and a shader core. Those are the only two
11 elements that need to be contained within the first
12 processing unit in order for there to be infringement.
13 For there to be direct infringement.

14 The first processing unit need not contain a
15 scheduling module. It need not contain the second
16 process. In fact, the computing system as a whole,
17 which is only recited in the preamble, need not contain
18 either the scheduling module, nor does it need to
19 contain the second processing. It only needs to contain
20 a memory, a berth processing unit, and a bus.

21 And we'll go into more details of the ITC's
22 construction of these issues and the, you know, some of
23 the aspersions that were made by counsel for Realtek
24 concerning whether AMD's taking the position that
25 scheduling module is limiting.

1 Just to provide a brief preview, AMD's
2 always maintained that it is limited. The phrase a
3 scheduling module as used in claim 15 is limited. But
4 it is limiting on the configuration of a plurality of
5 engines, meaning the plurality of engines has to be
6 configured to receive from a scheduling module the
7 plurality of tasks. But it does not require a
8 scheduling module to be present in the first processing
9 unit, in the second processing unit, or in the
10 computing system in order for infringement to lie, and
11 that has always been AMD's position, and it's not
12 changed.

13 MR. TERRILL: I'm sorry to interrupt you so
14 early in the presentation, but I think this is an
15 important point that I don't think I understood until
16 now. Under -- I think having heard AMD's position on
17 this, is this something that requires claim
18 construction? Does the EOE branch -- first off, has
19 the commission already construed the relevant claim
20 limitation to confirm the interpretation that you just
21 provided? Or would the branch have to interpret the
22 scheduling module to decide whether under the claim
23 language that's correct that the scheduling module does
24 not have to be part of the plurality of engines and, in
25 fact, doesn't have to be -- as long as it's just part

1 of the computing system, it could be outside of any of
2 the other limitations that are recited here?

3 MR. RIZK: The ITC did squarely address this
4 issue at our review. Before the ITC, the same issue
5 came up between the parties where Realtek argued that
6 the claim that the alleged engines couldn't meet, which
7 are the job slots, I think, by Mr. Higgins' own
8 admission, job slots in the Mali GPU. That they
9 couldn't satisfy the plurality of engines requirement
10 for two reasons.

11 Number one, because they were hardware only.
12 Because the alleged plurality of engines were hardware
13 only, and the ITC squarely rejected that argument. And
14 I will -- I promise you that I will, put that quote
15 before the EOE branch today.

16 The second issue that came up is Realtek
17 also tried to argue that to infringe the Realtek chip,
18 for example, would have to have the scheduling module.
19 And they allege that that scheduling module would need
20 to be part of the claimed computing system.

21 And the ALJ dismissed that argument because
22 all of the parties agreed that the computing system is
23 in the preamble, and none of the parties alleged that
24 it was limited. And it's not. There is no evidence that
25 the computing system breathes life into the claim. And

1 had Realtek wanted to raise that claim construction
2 argument during the ITC proceeding, it should have done
3 that.

4 The last issue was in connection with the
5 domestic industry products. During the ITC proceeding
6 for the domestic industry products, Realtek alleged
7 that there was no evidence that a scheduling module was
8 present on the domestic industry products.

9 And the ALJ also dismissed those arguments,
10 because it is clear that the hardware configuration
11 itself is set up so that way it could be fielded in an
12 intended operating environment where it receives a
13 plurality of tasks from a scheduling module. That's
14 what the hardware's designed to do. That's what the
15 Mali GPU hardware's designed to do, and we'll go into
16 the exact design details that provide the required
17 configuration irrespective of the new Mali driver.

18 MR. TERRILL: Thank you.

19 MR. RIZK: You're welcome. So the bottom
20 line is that as the claim is written, the claim does
21 not -- only requires the plurality engine to be
22 configured to receive. That means that they need to
23 have the necessary hardware componentry to perform the
24 receiving step of the plurality of tasks.

25 They need to be need capable or configured

1 to interact with an external scheduling module that is
2 actually executing on an external CPU. That's the key
3 feature of this. Can't be a scheduling module. As
4 part of the GPU that it's talking to, it has to have
5 the necessary componentry to communicate or to receive
6 a plurality of tasks from an external scheduling
7 module. And we'll see shortly that that is precisely
8 the configuration of the accused products.

9 Okay. So turning to claim 19, that further
10 limits claim 15 requiring the first task to comprise a
11 graphics processing task, and a second task to comprise
12 a general compute task so the shader core has to
13 execute those two types of tasks concurrently.

14 And then claim 20 further limits the shader
15 core to be configured to receive the plurality of tasks
16 based upon a priority. Again, this required
17 configuration of the -- the shader core is how the
18 shader core is configured, and we'll see when we get
19 into the details that the Mali shader core is
20 configured to execute those different types of tasks
21 based upon an internal component called the [REDACTED]
22 [REDACTED]. That's what was submitted as evidence a
23 before the ITC, and that's what the ID discusses. It
24 discusses the [REDACTED] configuration in enabling
25 the required configuration of the shader core. And

1 there are simpler hardware configuration features that
2 get in connection with the priority.

3 So to prevent -- the entire industry ended
4 up adopting ultimately AMD's asynchronous computer
5 architecture for it to be very critical to performance,
6 and to prevent further injury with regards to companies
7 such as Realtek, or TCL that were using AMD's patented
8 technology without AMD's authorization.

9 AMD brought this ITC investigation roughly
10 two years ago now. It invested years of time. There was
11 a five day final evidentiary hearing. There was
12 extensive briefing before the ALJ and the commission.
13 And then finally, as we all know, the commission agreed
14 with AMD that Realtek was in violation of Section 337
15 based upon its infringement of claim 19 and 20 of the
16 '381 patent which go to the heart of AMD's asynchronous
17 task dispatch technology.

18 As a result of that, the commission issued a
19 limited exclusion order both against TCL and Realtek
20 with respect to their digital televisions, as well as
21 integrated circuits containing GPUs, and printed
22 circuit board assemblies containing the same. So those
23 are the three product categories.

24 In this case, TCL has since settled, so it
25 is no longer subject to the remedial orders of its

1 importations into the United States are authorized.
2 Realtek has not, so it remains subject to the LEO in
3 this case. Technically, Realtek's -- the Realtek
4 products that would be the subject of the LEO would
5 include your legacy standalone ICs. That's the
6 integrated circuits containing GPUs, and printed
7 circuit board assemblies containing the same, which
8 corresponds to the legacy development boards containing
9 ICs, which also happen to contain Mali drivers.

10 But the key point here, and take away is
11 that unlike the legacy development boards which contain
12 the drivers at the time of importation, the legacy
13 stand-alone ICs that contain GPUs, do not contain the
14 Mali GPU driver, but were nonetheless found to directly
15 infringe.

16 Before we get into the details of that, just
17 a little bit of background in terms of the -- some of
18 the technical documents that we're going to see here.
19 There are two distinct types of source code. There's
20 something called software source code, and then RTL
21 source code.

22 Software source code would be source code
23 that would be used to design the function of software
24 such as a driver, or an application, or an operating
25 system. That would then get compiled by the -- usually

1 the Realtek end customer would compile that software,
2 that source code, into something called a binary or
3 executable file.

4 Unlike source code, a binary executable file
5 is not human readable, and would not be considered
6 source code. That's going to be important when we start
7 talking about the applications that are on the
8 development board, and Mr. Terrill's earlier question
9 to Mr. Higgins regarding whether there has been any
10 comparison between the legacy product and the new
11 Realtek development boards that would allow Realtek to
12 avail itself of prior customs rulings regarding the
13 disablement.

14 RTL source code is written in different
15 programming languages than software source code will be
16 written using something called RTL. So if you hear
17 RTL, that connotes hardware source code. Would also be
18 written in VHDL or virilog. It would also be human
19 readable, like seeing if it had some distinctions.

20 Ultimately, that hardware source code in the
21 form of RTL would be input into something called a
22 synthesizer, which is the equivalent of a compiler for
23 software. And that synthesizer would output the
24 physical design of the logical design that was defined
25 by the RTL.

1 So the RTL fully specifies the functions,
2 and logical design of the circuit. Input into a
3 synthesizer, which would output the physical
4 implementation. Realtek would then take that physical
5 implementation design and transfer it to a boundary
6 like TSMC or UMC, who would transfer the physical
7 design, the physical implementation of the RTL into a
8 silicon chip, which is called an integrated circuit. So
9 that is the component. That is the integrated circuit
10 containing the GPU that is imported into the United
11 States.

12 So we've already touched upon this briefly.
13 This will be my last background slide. There are three
14 important layers in the supply chain at issue here.
15 Realtek is at the center of it. Their job is to -- they
16 don't design their own GPU technology. They obtain
17 predesigned GPU designs in the form of RTL source code
18 from a company called ARM.

19 They go through that process that I just
20 described of synthesizing it into a physical
21 implementation, shipping that physical implementation
22 to a bag that burns the physical implementation of the
23 circuitry into a silicon chip, which you can see on the
24 left hand side, the silver object that's shown on the
25 left hand side.

1 They will sell those stand-alone, those
2 integrated circuits as stand-alone chips. They also
3 supply as a value add to their very sophisticated
4 customers something called a demo board or development
5 board. These are printed circuit board assemblies that
6 incorporate the Realtek IC, and a number of other third
7 party ICs, as well as software that they will put to
8 work with -- to work with the IC.

9 They will sell both the development boards
10 in low volume, as well as the Realtek integrated
11 circuits to end customers. The Realtek integrated
12 circuits when sold don't include the Mali GPU driver.
13 As they're shipped across the country, there's no Mali
14 GPU driver on it.

15 ARM Realtek customers are given the option
16 to download the Mali GPU driver, but they have
17 unrestricted ability to make whatever changes that they
18 want to the source code. It is provided to them in
19 source code form.

20 And Realtek imposes no restrictions
21 whatsoever on what their customers do with that driver.
22 And that's based upon testimony from Realtek witnesses
23 in the prior 1318 proceeding.

24 So what do Realtek's customers do with those
25 products? For the development boards, we'll get into

1 this in more detail. They use it to develop their
2 product using Realtek's chip. So the typical use
3 application is they want to test the customer
4 proprietary software application using the Realtek IC.
5 That's their typical use case of a development board.

6 And necessary to that is to make all kinds
7 of modifications. Think of a development board like a
8 professional PC where it has an intel processor built
9 in, but the end user, which in this case are
10 sophisticated multinational technology companies,
11 that's Realtek's customer, to make whatever changes
12 they need to that in order to develop their
13 application. That's the purpose of the development
14 board, and why it's called the development board.

15 With respect to chips, again, you know,
16 those are ultimately if the customer chooses to award
17 Realtek with a design win, those -- they will integrate
18 their chip into a completely custom customer
19 proprietary board, not into the Realtek development
20 board, with all of their customer proprietary
21 applications communicating with the chip.

22 All right. So let's jump into the petition.
23 So there are two big issues that we're going to be
24 talking about today. The first big issue is what I
25 think there was some discussion about before, which is

1 that for these categories of products that Realtek has,
2 the Realtek integrated circuit, and the Realtek
3 development board, their addition of their -- their new
4 Mali driver, you don't even need to get into the
5 technical details. It doesn't disable anything. In the
6 end use application, it does not meet the standard for
7 disablement with how they set it up.

8 And we're going to start with the
9 stand-alone IC first. So for the stand-alone IC, as I
10 mentioned earlier, that contains the complete ARM Mali
11 GPU. There's no part of the ARM Mali GPU that is not
12 part of the IC.

13 See the picture of the left of the IC. The
14 entire block diagram on the right, while I admittedly
15 borrowed it from a different built-in chip, the RTD
16 1312, the block diagrams are going to look more or less
17 the same. But all of these modules, including the whole
18 ARM Mali GPU will be baked into the silicon of the
19 Realtek chip.

20 And they're sold as standalone components
21 that are then imported into the United States in high
22 volume in these tape and reel type packages. Sometimes
23 it could be a cassette. Multiple different forms of
24 package, volume packaging they would use to ship these
25 standalone chips to their customers.

1 But in all cases when they ship the chips to
2 their customers when it's crossing the border, there is
3 no Mali driver that is incorporated into the chip.
4 There's nothing that is modified by the new Mali driver
5 on this chip whatsoever.

6 Mr. Higgins had a number of comments that
7 we'll get into later about the job slot two being
8 removed. It's completely there. There is nothing
9 changed in here in this chip whatsoever.

10 Now, Realtek incorrectly suggested that the
11 commission's decision is limited to Mali driver in
12 combination with standalone ICUs. That wasn't the
13 commission's decision. They have no basis to say that.
14 The commission's decision, as we can see on this slide,
15 excluded our integrated circuits that incorporate one
16 or more graphic processing units. It doesn't say and a
17 Mali driver.

18 Commission was clearly aware of the Mali
19 driver. The issue was squarely before the commission.
20 Realtek argued that, and they bailed. Commission
21 rejected the argument. Integrated circuits that do not
22 contain a Mali driver are excluded from entry into the
23 United States because they directly infringe.

24 Now, this is on the scheduling module being
25 limiting. I promised I would put up some quotes. Here

1 are a few quotes. Again, this issue of whether or not
2 AMD's contending that they're limiting, was an issue
3 that was disputed before. And the issue that the ALJ
4 addressed, this is the ID at page 129 is that
5 respondents, Realtek, acknowledge that the preamble
6 computer system comprising is not limited.

7 So whether the scheduling module's inside or
8 outside of the computing system is not required by the
9 claims. Claims are expressly silent on that point.

10 And, again, ID at 130, the ALJ resolved that
11 a person of ordinary skill in the art would understand
12 that a scheduling module and second processing unit are
13 outside of the GPU. Don't need to be part of the GPU
14 in order to infringe. So that's been AMD's position all
15 along that GPU by itself, that the Realtek integrated
16 circuit by itself directly infringes. And Realtek
17 hasn't shown that the changes that it has made to that,
18 to the external driver that isn't even part of the
19 Realtek IC, disables the GPU functionality.

20 Certainly doesn't do that in the stand-alone
21 Realtek IC configuration, and we'll talk about why for
22 the Realtek development boards their petition fails as
23 well.

24 At best, I would say that the federal
25 circuit precedent strongly supports AMD's position. The

1 scheduling module is just like the operating system in
2 the Silicon Graphics case. The fact that while if you
3 don't connect it to a driver, you can't use it, well,
4 sure. Yeah, if you -- you take the processor outside of
5 your computer, and you, you know, completely disconnect
6 it from anything else, it'll be really hard to use it.
7 But that doesn't mean that there is nothing inside.
8 That it's an empty brick, as Mr. Higgins puts it.

9 And we'll see. We'll walk through in detail
10 every element all of the functionality of the detailed
11 architecture that enables asynchronous task dispatch is
12 all implemented within that chip.

13 So I think we're more like Silicon Graphics
14 here where the scheduling module's part of the external
15 operating environment, and not required to support a
16 finding of direct infringement which has been AMD's
17 position all along.

18 So now we're into the detailed plans. Let's
19 go over what was actually presented to the commission.
20 Plurality of engines, we all agree, it was the job
21 slots. Job slots are in the GPU. Mr. Higgins mentioned
22 several time, well, the new Mali driver removes job
23 slot two.

24 Let's be very clear. Job slot two
25 corresponds to a very specific piece of hardware in the

1 Mali GPU. There is literally a circuit that implements
2 the functionality of job slot two with a very complex
3 state machine that is implemented in digital logic
4 corresponding to the features that AMD alleges
5 infringed.

6 Realtek has made zero changes to that job
7 slot. It has not removed it. It is alive and well, and
8 still kicking in the Realtek ICs that are crossing the
9 border. And even when they're put into the Realtek
10 development board, those job slots persist, all three
11 of them, inside of the Realtek IC.

12 So earlier I said that there's a complex
13 state machine. This is the state machine that was
14 presented to the commission. So inside of the Mali GPU
15 itself there is a circuit that is labeled or called
16 dvalin_jm_job_slot. And it implements a very complex
17 state machine that provides the necessary configuration
18 to receive from a scheduling module associated with a
19 CPU. Associated with a different processing unit other
20 than the GPU.

21 That is the configuration that is required,
22 and that is the evidence that was presented. It was the
23 detailed hardware RTL source code, and the
24 functionality of the job slot state machine that
25 enabled the tasks, the plurality of tasks to be

1 received. None of that has changed.

2 Next when we go to what Dr. Reinman
3 testified about, he clearly testified and relied upon
4 the configuration of the engines, of the jobs as
5 providing the necessary configuration. In fact, he
6 corrected the questioning attorney, that I'm relying on
7 the hardware source code, the RTL source code that is
8 implemented in the silicon to show that they're
9 configured to receive from a scheduling module. None of
10 that has changed.

11 Now, I earlier alluded to some issues with
12 the Realtek's allegations about whether or not software
13 was required. This is a quote from the ID at page 112
14 where respondents actually asserted during the ITC
15 investigation that this limitation of configured to
16 receive from a scheduling module, plurality of engines
17 configured to receive from a scheduling module not met
18 because there -- doesn't have the required scheduling
19 module doesn't have software.

20 And the ID and the commission ultimately
21 rejected that argument. The ID, in fact, expressly
22 acknowledged that the alleged engines are hardware
23 only. And they acknowledge, Realtek acknowledged that
24 the alleged engines are hardware only. Nothing has
25 changed in this, because they didn't modify the

1 engines.

2 The second element was the shader core
3 configured to receive. So let's look and see what did
4 AMD rely upon, and what did the ALJ and the commission
5 cite to for providing the required configuration. Well,
6 what AMD relied upon, as I alluded to earlier, was a
7 module called a [REDACTED]. That again, the [REDACTED]
8 [REDACTED] and all of its functionality corresponds to a
9 circuit inside of the Mali GPU hardware. It is baked
10 into the silicon, baked into the Realtek IC. Realtek
11 hasn't changed it.

12 The [REDACTED], [REDACTED] that
13 are shown on the slide were also part of the GPU.
14 They're not part of the Mali GPU driver. So all of
15 this entire block diagram is all in the Mali GPU.

16 And what AMD argued is that the
17 configuration to receive and to execute was based upon
18 the execution engine's physical implementation of
19 [REDACTED] that are configured to receive multiple tasks
20 at the same time, and to execute those multiple tasks
21 at the same time based upon [REDACTED]

[REDACTED] [REDACTED]. And the configuration of the [REDACTED]
[REDACTED]
[REDACTED]
[REDACTED].

1 All of that functionality, again, unchanged.
2 It still performs all of those functions because it's
3 in the hardware. [REDACTED] will continue to operate
4 the way it has always operated, and Realtek can't argue
5 otherwise.

6 We already covered this, but this is the
7 internal micro pipeline architecture of the internal
8 chip. And that's what the ID credited. So this is
9 actually the ID at 116. And what they, in fact, I
10 think this is actually the final determination of the
11 full commission at page 52. Citing to the ID at 116,
12 and crediting the ID's finding that the configuration
13 of the shader core was found based upon the [REDACTED]
14 [REDACTED], which is inside of the Mali GPU hardware, and
15 its deliberate [REDACTED]
16 [REDACTED]. That was the cited
17 configuration.

18 So there's been some discussion or
19 suggestion by Realtek's counsel that the term
20 configured to must be construed so that way it has to
21 happen all the time. That can't likely happen, or
22 can't, you know, has to be consistent with the show
23 that it always happens.

24 But the ITC considered and rejected that
25 same exact argument. So during the pending ITC case,

1 Realtek argued that AMD did not prove, even though the
2 GPU had that [REDACTED]

3 [REDACTED], Realtek argued, well, AMD
4 didn't show that there are applications that are
5 constantly used, like what the use case is of that --
6 of the [REDACTED]. How often is that functionality
7 exercised.

8 And what the ID and the commission found is
9 that the configured to language does not require the
10 event to occur all of the time. So Realtek's arguments
11 that it makes to the EOE branch today about
12 prioritization, and that this configured to language
13 requires AMD to actually show that there's a
14 significant number of applications that will use the
15 GPU in an infringing manner is not necessary, and
16 inconsistent with how the commission came down with the
17 same exact language in the pending ITC proceeding.

18 So now we're onto the Realtek development
19 board. So, again, the stand-alone ICs it's
20 straightforward. All the configurations in the
21 hardware are found to directly infringe. They should
22 not -- Realtek is not entitled to a ruling with respect
23 to that category of products.

24 With respect to the development boards, this
25 isn't a system like the Apple watch that is a closed

1 system, or a TCL TV that is a closed system, right, or
2 any other consumer product that is not intended to be
3 modified. This is a board that is intended to be used
4 for engineers to take their custom software
5 applications, move them onto the development board, and
6 test them out with Realtek's chip where they have full
7 access to Realtek -- to the ARM source code. They have
8 full access to alternative source code, and they have
9 unrestricted access to do whatever they want.

10 So it's a very different type of product.
11 But this is a picture of it shown on the slide. You
12 can see a printed circuit board assembly, which is the
13 red rectangle. The Realtek IC is obscured by the black
14 heat sync, but it would be soldered onto the board.

15 And then there will be another chip,
16 typically a memory chip which is circled in blue that
17 would incorporate potentially a software operating
18 environment that is provided by Realtek, but subject to
19 potential modification, just like a general piece. And
20 that would include an operating system, potentially the
21 Mali driver, application software.

22 But all of those as implemented in the board
23 itself wouldn't be in source code form. That would all
24 be in binary form as it's put on the board. So the
25 operating system, Mali driver application software,

1 those will all be binaries that are stored on the chip.

2 Realtek would separately provide the source
3 code for the Mali driver to its customers, and then its
4 customers could do whatever they want, or change, you
5 know, replace the Mali driver that's on there with a
6 different Mali driver. Maybe they got a Mali driver
7 from somebody else. But the default that's put on there
8 would probably be whatever Realtek puts on there.

9 So, again, these are devices, though, that
10 are intended to be reprogrammed and modified. Realtek
11 customers use the Realtek development boards to test
12 integration of Realtek ICs with their own workloads.
13 They don't care if it works with the stock Android
14 version that Realtek put on the development board.

15 They want to make sure that it works with
16 the Roku operating system. They want to make sure that
17 it works with the Amazon operating system and driver
18 configuration. They don't care about what Realtek did.

19 Now, Realtek originally tried to withhold
20 documents that relate to what changes Realtek would
21 make. We asked them for contract agreements, which I
22 think Mr. Terrill perhaps asked about. They didn't
23 provide us with these contractual agreements. We asked
24 them for instructions that they provide to their
25 customers, and how to use the board which, of course,

1 we provided. There's no way that they're going to
2 provide a product with a development board like that
3 and say, hey, figure out how to connect to it on your
4 own. They initially said they didn't have it. They
5 didn't have any instructions that they provided to
6 their customers.

7 Ultimately, AMD's technical expert,
8 Dr. Reinman found that there were documents on the
9 source code computer that were improperly made
10 available. It's unclear whether Realtek intended for
11 those documents to be there. But what these documents
12 tell us is what I've been saying all along is that
13 these types of products are designed to be modified by
14 the end customers.

15 These aren't products that are closed just
16 like the Apple watch. So this is, you know, cited.
17 It's a document cited in Dr. Reinman's declaration, but
18 the document which is RTK-177-00024, states
19 specifically the instructions on how to modify the
20 contents of the board. How to replace the binary files
21 that are in that memory with, you know, with whatever
22 binary files the customer wants to use.

23 Now, we asked Realtek to provide its
24 30(b)(6) witness that can provide some testimony,
25 perhaps, on the -- you know, the operation of the board

1 and how, how its customers use these types of boards.
2 Realtek refused to provide witnesses in this
3 proceeding. What we do know is that from prior
4 proceeding from the 1318 investigation, when we had the
5 processes available to compel the deposition testimony
6 of Realtek, Realtek witnesses and corporate reps
7 testified that their customers can do whatever they
8 want.

9 So you have Mr. Chen testifying at 136,
10 lines one through nine testifying TCL can modify
11 anything it wants to. We give it the source code. It
12 does what it wants. We don't know. We don't ask. We
13 don't know. That's the basic gist of the testimony. So
14 they exercise zero control on their customers, at least
15 based upon the evidence of record.

16 So where does that leave it? Well, let's
17 leave aside the boot lock option, which Mr. Higgins has
18 alluded to several -- at several points. They don't
19 meet the required standard for disablement.

20 They haven't been able to show that at least
21 as to the boards, board number one, which doesn't have
22 the boot lock code, that the features are unreasonably
23 inaccessible as required by the applicable legal
24 standards. So they're not able to show disablement of
25 the GPU. These are boards that are intended for

professionals to develop and to modify.

As to the boards, I think it's board number two that had a boot lock. Board number three completely unattached. So they're not really seeking a ruling on that at all.

But for board number two, they actually withdrew their request on board number two. The only board that had this supposed boot lock feature, and they did it for good reason. They did it because when Dr. Reinman analyzed the boot lock, he found that it doesn't work. And they provided no expert rebuttal in response to Dr. Reinman's analysis of the boot.

In fact, when we asked their expert about this boot lock, Dr. Bister. He couldn't even identify the one time programmable boot key in the produced code that Mr. Higgins alluded to.

So here's what Realtek says. This is in its reply at 76. Realtek is not seeking a ruling on boards two and three. So respectfully at this point, given that it hasn't produced a working device that has boot lock, and it's effectively withdrawn its request, the only board at issue, board number one, is -- it doesn't meet the standard for disablement.

And, therefore, we don't even really need to get into the more detailed nuance of their new Mali

1 driver, and what it does. But that should provide the
2 EOE branch with sufficient basis to deny their request
3 just based upon the types of products that are at issue
4 here, and what they've done.

5 So but let's take the arguments that they
6 made head on. And I've heard today for the first time
7 that the performance enhancement patch is not part of
8 this ruling request. Today is the first time I'm
9 hearing that, and I was really surprised.

10 Seems like this is a moving target, because
11 we specifically went, AMD specifically went to the EOE
12 branch to avoid any ambiguity about what is the exact
13 and precise configuration of the product for which
14 they're seeking a ruling.

15 And we asked the EOE branch to compel
16 Realtek to produce a physical sample of the device that
17 they want to import into the United States. So AMD's
18 experts can test it, and they did that.

19 This is -- I'm looking at an email from Mr.
20 Terrill that is dated Thursday, February 22nd, 2024,
21 1:51 p.m. And in that email, Realtek was directed to
22 provide to AMD by no later than close of business
23 tomorrow, which was 2/23 at the time, a sample of the
24 relevant integrated circuits that incorporate the
25 relevant chip used in the driver device. And they

1 provided that consistent with the EOE branch's guidance
2 in the form of board number one.

3 Now, board number one undisputedly contains
4 the patched -- sorry, the performance enhancement patch
5 as part of. It's built into the new Mali driver and
6 configuration of that board. That's what AMD's experts
7 analyzed, the board that we were given pursuant to
8 AMD's request, and the EOE branch's guidance.

9 Now, board number two and three are out
10 based upon Realtek's statements. But board number two,
11 which was the only other board that had the new Mali
12 driver also incorporated, the performance enhanced
13 patch as part of it.

14 Now, the performance enhancement patch, Mr.
15 Higgins tries to pretend that it's just like we're
16 building up a hamburger where if you receive the
17 hamburger, and it has pickles, but you didn't want
18 pickles, you could just take out pickles, and we're all
19 good.

20 That's not how it works. These are really
21 complex systems where all of these things work
22 together. And the configuration that AMD's experts
23 tested and considered and relied upon was rightly the
24 configuration that was present on that development
25 board, board number one, as well as the source code,

1 the full complete source code.

2 That was another issue that we went to the
3 EOE branch for. We had asked for -- they'd given us the
4 source code in piecemeal what they call the patches.
5 We asked them please give us the exact configuration.
6 Put all these patches together. We want to see the
7 actual source code when it's all built together, what
8 does it look like, and had the performance enhancement
9 patch code.

10 Again, that's what AMD's experts considered.
11 We didn't consider this other configuration. And at
12 this point in the proceeding, for the proceeding to
13 conclude at a reasonable -- in a reasonable time, we
14 don't have the time to go back and do a redo, because
15 now Realtek wants to take out the pickles.

16 There just isn't enough time. It would take
17 months for AMD's experts to consider a completely
18 different configuration for which we were not given
19 Realtek development board physical samples as we asked
20 the EOE branch to help us obtain. We were not given
21 the source code. And we were, you know, if we're given
22 the time, we would provide an opinion.

23 We haven't been given the time. We were
24 providing a submission based upon what was given to
25 AMD. And the fact of the matter is Realtek tries to

1 make some hey about, well, we didn't really conceal.
2 Well, let's be -- you know, just to be fair, they
3 didn't put it in their petition. It is clearly
4 something that they developed. They decided that they
5 were going to put it on the product that AMD moved it
6 to -- essentially moved to compel to obtain. But they
7 neglected to include it in your petition, and your
8 experts neglected to analyze it.

9 Now, if it is taken out, that would
10 certainly -- even in a subsequent proceeding, let's
11 just presume for a moment that the recourse here would
12 be for Realtek, which I hope will be to properly file a
13 new one, instead of proceeding. And then to, you know,
14 provide a physical sample that doesn't include that
15 functionality. If they were given the opportunity, what
16 I would suspect is that there would still be some real
17 problems with Realtek's proposed solution to the
18 performance enhancement patch.

19 One issue would be how do you certify around
20 that? They're asking for, you know, the solution is we
21 want to be able to certify that it includes patches one
22 through four. And if it includes patches one through
23 four, we want permission to enter the device into the
24 United States.

25 Well, if it includes patches one through

1 four, but also includes the performance enhancement
2 patch, would that be encompassed? Would they have to
3 certify that it includes patches one through four, but
4 then also doesn't include the performance enhancement
5 patch, or anything like it? So it creates some real
6 challenges and problems, even if the commission -- if
7 the EOE branch were to consider that type of approach.

8 But let's just take the performance
9 enhancement patch out. So none of Realtek's experts
10 considered the performance enhancement patch.
11 Dr. Reinman is the only expert that considered it. So
12 when Mr. Higgins says we don't believe that the
13 performance enhancement patch -- the new Mali driver,
14 that this applied to AMD, infringes. That's Mr.
15 Higgins' opinion. That's not an expert opinion. Mr.
16 Higgins, well, like me might have a double E degree, he
17 is not a technical expert, and hasn't been subject to
18 the perils of cross-examination, or providing a written
19 opinion in this case.

20 Second issue is Realtek's experts, to the
21 extent they offered any opinion, was completely
22 untethered to any claim construction or interpretation
23 of the claims. When asked at their depositions did you
24 read the patent, the answer was, no, I didn't read the
25 patent. It almost feels like between the lack of

1 review of the entire source code, as well as the lack
2 of reading of the patent, that this was written by Mr.
3 Higgins, and the truth of it is that the experts merely
4 just rubber stamped his work.

5 Well, let's take the issues head on. So
6 starting with the presumption that it is built in,
7 which is the presumption that work load was given to
8 AMD in this proceeding based upon the development
9 board.

10 So what do we have? We have Dr. Reinman's
11 un rebutted testimony that it was part of the
12 configuration that he was given. It was part of the
13 demonstration boards. And it was part of the source
14 code. That's from Dr. Reinman's declaration.

15 Next what do we have about the patches? So
16 when we asked ARM and Realtek for information, more
17 information about these vertex shader programs, what we
18 were told at the deposition is we're not -- we're going
19 to instruct the witness not to answer. We're not going
20 to say why we did not disclose this information in the
21 original petition.

22 So there's no discovery that we were able to
23 get based upon these improper privileged instructions
24 of why Realtek chose to conceal this patch from its
25 initial petition, and to provide a way in with it.

1 What we do know again is that the new Mali
2 driver contains the patches. The physical sample
3 contains the performance enhancement patch. And that's
4 confirmed by Dr. Reinman. What it does is it actually
5 modifies what was known as legacy vertex jobs in a
6 product that was before the ITC.

7 So it is true that AMD in the ITC
8 proceeding, it did not accuse what vertex jobs in that
9 proceeding, which were the legacy vertex jobs, of
10 containing compute tasks.

11 But we're not talking about the legacy
12 vertex jobs any more based upon the change that they've
13 made with this performance enhancement patch. They've
14 modified vertex jobs to import a new type of task. And
15 that new type of task is the vertex pilot shader
16 program that was formerly in the legacy, part of the
17 legacy compute job, and that the commission found to
18 infringe.

19 And you can see that right in the
20 Dr. Reinman's demonstrative code on this slide where
21 basically all they did was shift the same exact vertex
22 pilot shader program. They jammed it into a compute
23 job without making any changes whatsoever to the
24 underlying vertex pilot shader program.

25 So all we have to go off of about really

1 that is, if it's anything, is unfortunately this
2 release note that aren't provided. And I say it
3 unfortunately because this release note was actually
4 written by their litigation team.

5 The individual who authored it, Guy Larri,
6 he reports into the head of litigation at ARM. And his
7 role is more or less a professional testifier on behalf
8 of their litigation team. So he authored this release
9 note to, you know, explain his view on the -- you know,
10 what the vertex pilot shader program is.

11 We'll get into that we tried to get some
12 more discovery on whether or not ARM considers this to
13 truly be a mistake, or a mere error or not. But we were
14 -- why ran into a number of roadblocks, to say the
15 least.

16 But even if you take the most favorable
17 document on its face value, what this document says is
18 that these -- these are essentially general
19 calculations that operate on uniforms that will be
20 wasteful to run as what one would normally consider to
21 by a vertex shader board. And so this, as Dr. Reinman,
22 who is the only who's weighed in on it, would be an
23 example of a general purpose compute task about this
24 required by the patent.

25 And his testimony that this is a general

1 purpose compute task, and if the changes they've made
2 are unsubstantial to infringement, it's unrebutted by
3 any other technical expert.

4 So just to sum up, what they've done here is
5 this is a demonstrative showing the ID on the left hand
6 side. And what they had is that AMD accused the -- or
7 compute tasks, AMD had accused tasks that were included
8 within compute jobs. That necessarily encompassed these
9 vertex pilot shader programs.

10 So Realtek has brought up the issue of
11 waiver several times. Realtek never argued to the ITC
12 that there was a mistake, and these compute jobs
13 contained these vertex pilot shader tasks. And those
14 aren't compute tasks, therefore, there's no
15 infringement.

16 So to the extent that there's any waiver
17 here, it's on Realtek, because they never made that
18 argument to the ITC. They shouldn't be allowed to argue
19 now that these -- that a subset of a compute job,
20 without making any change to the underlying programs
21 that they run, and only shifting them into the list of
22 vertex tasks, somehow transforms it from a compute task
23 to a graphics process task. They've already conceded
24 effectively that a vertex pilot shader program is a
25 general compute task.

1 And as you'll see, Mr. Larri, who is ARM's
2 30(b)(6) conceded that there was no change made to the
3 vertex pilot shader program task. It's the same exact
4 program. Mr. Higgins mentioned several times that, oh,
5 it's different. We made changes to the program. That's
6 not what Mr. Larri said. And Mr. Larri is actually
7 the, you know, he's the ARM corporate representative.
8 Mr. Higgins has no basis for his statement that the
9 program has changed.

10 So in sum, with these vertex pilot shader
11 programs, again, we're going back to the hardware.
12 When that hardware receives that vertex shader program
13 through the same channel, which will job slot zero,
14 it's going to do exactly what it was doing with those
15 programs before. It's going to alternate between that
16 and a fragment shader request, and execute them
17 simultaneously.

18 So in terms of the hardware configuration,
19 with the vertex program shader programs, that there is
20 no change whatsoever. Still running those two types of
21 tasks simultaneously. So this is Mr. Larri juxtaposing
22 it against Mr. Higgins. And what Mr. Larri said is
23 whether it's submitted as a compute job, or a vertex
24 job, it will run the same exact program. Doesn't make
25 any difference. And the program will perform the same

1 exact computations. That's Larri's transcript in this
2 proceeding, lines -- page 94 lines one through six.

3 So to resolve this dispute, we did ask for
4 additional discovery on the vertex pilot shader
5 programs. Realtek refused. We asked on the record for
6 ARM's internal engineering technical documents that
7 talk about the vertex pilot shader programs, and why
8 they classified them as compute tasks, which we believe
9 was the proper classification. Realtek failed to
10 produce those documents at the ITC.

11 When similar disputes arose, AMD moved to
12 compel ARM to produce internal engineering documents.
13 Realtek didn't do that. They have a common legal
14 interest agreement. They probably wouldn't need to do
15 that. They wanted the documents based upon their
16 common legal interests in ARM. But they didn't produce
17 those documents.

18 We also asked for testing. Apparently there
19 was testing that was performed by Realtek and by ARM.
20 The ARM witness was instructed not to reveal any
21 information about the testing that was performed. And
22 we were unable. We requested that information from
23 Realtek and ARM on the record, and they refused to
24 provide it. So now we're onto claim 20, which is the
25 prioritization.

1 So there was a lot made about, you know,
2 there are 10 different forms of prioritization. That's
3 true. There are 10 different forms of prioritization.
4 Again, all of these are on the -- many of them are on
5 the hardware. They're baked in. Many of them are in
6 the software, which is just part of the external
7 operating environment.

8 And, therefore, we don't -- we believe that,
9 you know, prior customs rulings that relate to
10 preclusion don't involve issues where the functionality
11 was found only outside the actual product that was
12 found to directly infringe, and only in the intended
13 operating environment. And cited for the purpose of
14 demonstrating the use case.

15 But let's just focus on three specific types
16 of prioritization that were created just based upon the
17 new Mali driver itself. And I've listed those out as
18 task specific prioritization. Prioritization for the
19 tasks in job slot one. Prioritization of fast based
20 Mali sequence number.

21 And it's worth mentioning just right off the
22 bat very first bullet point on the slide, which is, by
23 the way, Realtek's expert agrees with AMD that Realtek
24 has not eliminated all forms of prioritization as it
25 claims. That's not what the new Mali driver is. And at

1 a minimum, it's rebutted. It's at least unrebutted by
2 Realtek's experts, AMD's position.

3 They never considered the new prioritization
4 that gets created, because they never considered the
5 complete new Mali source code. They were only given
6 bits and pieces that Realtek's counsel wanted them to
7 see.

8 And when they applied their opinions to what
9 little prioritization they applied their opinions to,
10 it was using some weird lens that isn't supported by
11 any legal precedent related to claim construction. They
12 essentially considered something to be prioritization
13 if it -- if the ARM document said that it was
14 prioritization, but they didn't actually review the
15 patent, and they didn't consider whether or not there
16 was prioritization based upon that.

17 The ARM, sorry, Realtek came up with a bunch
18 of attorney arguments primarily Mr. Higgins that are
19 based upon a number of tortured constructions where
20 Realtek imposes four different requirements. Most of
21 those have already been addressed by the commission.
22 They could be dismissed easily. But we'll go through
23 those.

24 So first let's start with Realtek's own
25 expert, Dr. Pfister. At his deposition he was asked do

1 you agree with Realtek's statement that there's no
2 longer any priority anywhere in the Mali GPU? That is
3 literally what Realtek said page 49 of its request
4 letter. His answer, No, I can't say that. I think that
5 about sums it up, particularly given that it is
6 Realtek's burden to show that prioritization has been
7 eliminated. That by itself should be sufficient to
8 deny Realtek's request.

9 Well, let's take the three new forms of
10 prioritization head on. So first task specific
11 prioritization, and this is based upon the behavioral
12 change. This form of prioritization allows one type of
13 task. Let's say a compute, attached compute job that
14 fully consume the execution resources temporarily while
15 other non compute job tasks are sitting in a cue and
16 are prevented from getting access to the GPU.

17 Let's be clear, this type of prioritization
18 was actually directly discussed in the patent itself.
19 Sorry. And we'll go through the specific citations,
20 but the patent specifically discloses embodiments of
21 this type of prioritization. Because Realtek withheld
22 code from its own experts, they didn't really weigh in
23 on this. So the only testimony that is on the record
24 is Dr. Reinman.

25 And Mr. Larri, ARM's own technical witness,

1 maybe the only witness on behalf of Realtek that
2 weighed in on this, he agreed that this type of a
3 prioritization could result in one type of job
4 finishing faster, but suggested it would require
5 testing.

6 So this is Mr. Larri's transcript at page
7 195, line 15, through 196 line 13 he was asked -- he
8 answered, that in this scenario you describe the
9 fragment job might finish faster because it's not
10 sharing resources on the shader core with another type
11 of job. Now, he said that it might require testing. We
12 asked for testing but, again, Realtek and ARM chose to
13 withhold that testing. And given that it is Realtek's
14 burden to demonstrate noninfringement of its redesign,
15 we think that that should be their failure to produce
16 testing that they clearly had should be held against
17 them with respect to their ability to meet their
18 burden.

19 They never produced any data on workloads to
20 support these arguments regarding duration of tests.
21 So, again, Realtek's only argument with respect to
22 testing that Dr. Reinman has performed to confirm Mr.
23 Larri is that well, we, you know, we at Realtek,
24 Realtek's outside counsel don't agree with
25 Dr. Reinman's testing. It's unrealistic. All of that

1 is just attorney argument.

2 But at bottom, Dr. Reinman confirmed what
3 Mr. Larri said, which is that this type of task
4 specific prioritization does allow one task to finish
5 faster than it would have without the prioritization.
6 And that's all that's really required for
7 prioritization under the claims.

8 Again, Realtek's experts never considered
9 the new prioritization mechanisms. And, therefore, they
10 haven't weighed in. Dr. Reinman's testimony is
11 essentially un rebutted on this type of prioritization.

12 Their testimony that they provided has only
13 been based upon the term priority literally appearing
14 in ARM's documentation. Since ARM didn't produce any
15 engineering documentation related to this
16 prioritization, that would explain why they were not
17 able to render an opinion and confirm that this
18 specific type of algorithm would result in
19 prioritization would infringe as claim 20.

20 And to the extent that Realtek makes a
21 number of claim construction arguments, one of them is
22 that the claim requires that the priority be part of
23 the task itself, or tied to the task itself. They tried
24 to make this argument with respect to the
25 prioritization by job select two. The commission

1 rejected it.

2 Specifically in the claim construction order
3 number 31, the ALJ held that the term task is construed
4 as junior processing board, and does not need to
5 include priority. So this notion that the priority has
6 to be associated with the task itself is a -- that is
7 an argument that they tried to push with the ALJ, and
8 the ALJ rejected in order number 31.

9 So let's look to see what the spec actually
10 says. And Mr. Higgins was focused on the sentence that
11 is, you know, immediately before the underlined
12 sentence in the top most passage. I'm reading 3.5,
13 column five lines 14 through 25, and I have underlined
14 the last sentence. He was focused on the sentence
15 before it. Sorry, he was focused on a couple sentences
16 earlier. But he didn't focus on this sentence that
17 says, To achieve temporal partitioning, the first and
18 second tasks share percentage of the time the
19 processing elements. And what's notable about that is
20 that's exactly how this type of prioritization is
21 implemented.

22 The patent further column 10, lines 47
23 through 51, which is where I think Mr. Higgins was
24 focused, talked about temporarily enabling any one task
25 substantially to fully consume the resources at GPU.

1 Mr. Higgins cut off his emphasis right after the preset
2 limits and ignored the last portion of the sentence
3 that's underlined that says by temporarily enabling any
4 one task to fully consume the resources of the GPU. So
5 the only expert that has reviewed this passage,
6 Dr. Reinman, has applied the lens with a POSITA, as
7 testified that he would understand that this passage
8 discloses to a POSITA that one way to generate the
9 priority is to temporarily enable one task to fully
10 consume the resource of the GPU, while the other task
11 doesn't have access. And that's exactly what Realtek
12 is doing with patent prioritization number one.

13 There were a number of other claim
14 construction arguments that were raised. I think we
15 talked about this already. They throw in the must
16 happen all the time. The commission has squarely
17 rejected that the configuration language of the shader
18 board requires this prioritization to happen all the
19 time. All that is required is that the configuration
20 has to exist.

21 And if Realtek is correct that this new Mali
22 driver software somehow affects the configuration of
23 the Realtek integrated circuit, which it doesn't, the
24 configuration is what it is, the same would be true
25 with respect to this new prioritization that they built

1 into the new Mali driver. So this new Mali driver would
2 then configure under Realtek's lens the shader core
3 specially programmed to cause this result to happen.
4 And that makes it configured.

5 We're not saying that this is -- it's not in
6 your capability. This is an actual configuration in the
7 driver code itself to cause when those two types of
8 tasks come in to cause them to -- the patched vertex
9 tasks at the patched vertex jobs, and the patched
10 compute jobs were those two types of jobs to be
11 serialized according to Realtek.

12 So next let's move on to prioritization
13 number two, job slot zero prioritization. One would
14 think this should be easy, because the commission's
15 already found job slot two favoring to be sufficient to
16 meet the claims.

17 Job slot zero is exactly the same. Mr.
18 Higgins suggests in his oral argument that, well, this
19 isn't the same, because really the FFS function
20 controls what goes into the job slots, not what comes
21 out. That's actually, again, part of the problem here
22 that we're going to see over and over is that Realtek's
23 experts didn't provide an opinion here. That's Mr.
24 Higgins' opinion. It's unsupported by any expert, and
25 that's not how it works. It actually -- FFS function

1 controls what comes out of the job slots, and not what
2 goes into the job slots.

3 Mr. Higgins also alluded to this FFS
4 function being a ping-pong. That's Mr. Higgins' term.
5 None of his experts have said that it's ping-pong. He
6 has tried to put that on Dr. Reinman's testimony, but
7 Mr. Higgins expressly asked Dr. Reinman is the FFS
8 function a ping-pong, and Dr. Reinman testified, no, it
9 is not ping-pong. He explained what it is, but it's not
10 a ping-pong. It prefers job slot zero over job slot one
11 in the same way that the job slot two was preferred
12 over the other job slots.

13 Having said that, this is a new type of
14 prioritization. Before there was only job slot -- take
15 Realtek at its word, the job slot two is actually
16 disabled, which it's not. But if we take Realtek at its
17 word that the job -- that the driver provides that
18 configuration, now you have a new type of
19 configuration. It's no longer per Mr. Higgins' own
20 admission, it is no longer a selection between job slot
21 zero, job slot one, and job slot two. Job slot zero is
22 getting selected more frequently, and it is getting
23 prioritized.

24 Mr. Larri, ARM's own corporate
25 representative agreed at his deposition and this

1 proceeding that FFS function preferring job slot zero
2 would be a solution. That's citing to his transcript
3 at 231 on lines 19. Line 19 to 232, line 12.

4 Last prioritization is age based, and
5 sequence number. You know, I think that Realtek's full
6 position on this, they have two sort of positions. One
7 is that this is functionality that was in the legacy
8 accused product. Again, the answer is no, it wasn't.
9 The legacy accused product, the selection by the driver
10 of what tasks would go next is principally based upon
11 the high, medium, low priority set. It's no longer
12 based upon this change that they made based upon it.
13 It's now principally driven by a different factor,
14 which is the age and sequence numbers associated with
15 the task.

16 Now, the only expert that has really weighed
17 in on this is Dr. Reinman, and he's provided credible
18 evidence that prioritization is predominantly based
19 upon the age on the assigned sequence number.

20 Dr. Pfister has also provided an opinion,
21 but his opinion is generally, and we asked him what is
22 prioritization. And he said, well, it would be, you
23 know, putting a numerical value on something that asked
24 for a high numerical value. We have a high priority,
25 and that would be an example of priority. That's

1 exactly what age and sequence number is. Age and
2 sequence number, if the age and sequence number is low,
3 it goes first. If the age and sequence number is high,
4 it goes last. There are other ways to determine what
5 gets selected next. But that is now in the new Mali
6 driver the way that Realtek is using to select what
7 jobs should go next to the GPU and, therefore, it
8 focuses.

9 Okay. So I'm going to, in the interest of
10 time, move on to the last issue here, which is -- and
11 this is something that applies to all claims, and
12 something that I think Mr. Terrill related to earlier,
13 which is what has Realtek done to compare the legacy
14 accused products -- to compare the legacy accused
15 products to the accused products that are now in this
16 case mainly demo board number one. And the answer is
17 nothing. There is no analysis or assessment of how the
18 applications that are on Realtek development board, new
19 Realtek development board number one in this
20 proceeding, how those applications are the same or
21 different from the applications that were present
22 before the ITC.

23 So while it is true that in the ITC AMD
24 requested the source code, and didn't see any evidence
25 in the source code of the development boards or TCL

1 products that were issued there, that the applications
2 were either using -- were using either patch, sorry,
3 legacy vertex jobs to submit compute, what were
4 fundamentally compute tasks, or to use applications to
5 cause the shader core to receive tasks based upon a
6 priority as required by claim 20. That does not mean
7 that the boards that -- that Realtek is seeking a
8 ruling on necessarily have that same functionality.

9 It is incumbent on Realtek in this
10 proceeding at a minimum to demonstrate that the
11 applications that are on the Realtek board, the source
12 code is the same. That they operate the same. Had it
13 done that, then we wouldn't be having this discussion
14 about, well, you know, do these applications for which
15 Realtek hasn't produced, do these applications
16 actually, you know, cause claim 20 to be practiced,
17 claim 19 to be practiced independent of what the new
18 Mali is practiced. But they haven't done that
19 analysis. And they haven't produced the source code
20 for AMD or its experts to perform that analysis either.

21 So even if the EOE branch were to accept all
22 of Realtek's arguments about the operation of the
23 patches, and the new Mali driver with respect to
24 serializing compute and non compute tasks, as well as
25 the removal of the alleged prioritization, and that

1 there is no new prioritization that is created by the
2 changes that they put into the new Mali driver, and
3 further that all of this disables the Mali GPU marker,
4 which it doesn't, if we accept all of that, Realtek has
5 still failed to meet its burden, because the undisputed
6 evidence shows that depending on the applications that
7 are unknown that are on this Realtek development, those
8 applications could just bypass those features.

9 By the way, shouldn't come as a surprise.
10 They would use the very same techniques that Realtek
11 used in its secret patch that it used, the performance
12 enhancement patch to simply go around whatever else
13 they put in software for the driver.

14 So let's start from the top. So at top, you
15 know, at the very beginning of this proceeding, AMD
16 flagged the need for the app, for the source code or
17 the applications that are installed on the development
18 board. We explained why they were relevant, and why
19 Realtek should produce them to satisfy its burden in
20 this proceeding.

21 Realtek did not produce the source code for
22 the applications that are installed on the development
23 board that it is seeking to report in this proceeding.
24 And so that is the current state of play. There is no
25 record evidence now of what those applications are, and

1 how their source code compares to the source code in
2 the legacy applications that were part of the legacy
3 products that were before the commission.

4 The fact, what what we have is, you know,
5 the exact opposite of what Realtek is saying. Realtek
6 was saying doesn't matter. That's not what Mr. Larri
7 said, who's the only expert outside of AMD's expert
8 who's actually really weighed in on this issue.

9 Mr. Larri said that it does happen. At his
10 deposition, he was asked using vertex jobs and compute
11 jobs can an application -- sorry, this was actually in
12 the 1318 investigation. And he attributed in terms of
13 the importance of applications, he attributed the
14 applications that are on the device as being the
15 component of the system that is responsible for
16 submitting general compute tasks to the GPU. So
17 Realtek's notion that it wasn't necessary for them to
18 produce applications because they're irrelevant, is
19 just simply wrong. In fact, in the ID which I have
20 quoted, page 114 of the ID, the ALJ explicitly noted
21 that it is the overall application's running on the CPU
22 that request different types of work to be done.

23 So, of course, it would be incumbent on
24 Realtek to show what are those applications on the
25 device that it is seeking. And I apologize. I got -- I

1 got confused with between the Larri transcripts. This
2 is the Larri transcript I had in mind. So this is in
3 this proceeding, Larri transcript 3/19/24, page 70,
4 lines 13 through page 71 line six, Mr. Larri was asked,
5 Can an application be constructed hypothetically to
6 submit compute and non compute jobs for simultaneous
7 execution independent of what all of these patches do
8 or don't do. And his answer was yes.

9 And, again, Mr. Larri was asked on the
10 prioritization, does the driver prevent applications
11 from prioritizing submission of shader programs to the
12 GPU's execution engine, and his answer was no, it
13 doesn't. Correct. Application programmers are free to
14 write application programs as they wish as long as they
15 conform with the Kronos standard APIs. Larri in this
16 proceeding, page 199, line 7.

17 Now, Dr. Reinman confirmed it. We had
18 limited information because Realtek refused to produce
19 the application source code, or any detailed analysis
20 of how the application source code on the development
21 boards compares to the legacy products. But Dr. Reiman
22 confirmed that, in fact, Mr. Larri was correct that
23 application developers could write applications, put
24 them on the development board, and access the
25 infringing configuration of the GPU.

1 And he confirmed this both with respect to
2 the prioritization claim 20, as well as the execution
3 of simultaneous computing non compute jobs. He was able
4 to confirm that the applications could submit those
5 types of tasks.

6 So looking at slide 77, this is a summary of
7 Dr. Reinman's tests. He constructed an application
8 that showed the compute tasks in the form of
9 simulations executed simultaneously with fragment jobs.
10 In this case, Mr. Higgins was saying again it's not
11 supported by any expert, but Mr. Higgins was saying,
12 well, these really operate on vertex.

13 Mr. Higgins asked Dr. Reinman repeatedly
14 whether or not they were operating on vertex, vertices.
15 Dr. Reinman repeatedly testified they were operating on
16 particles, not on vertices. So Mr. Higgins' comment
17 that it's really operating on vertex, not on and,
18 therefore, it's not a compute -- it's not a compute
19 shader, is not supported by any evidence.

20 Mr. Higgins' criticisms of Dr. Reinman about
21 the amount of time that it took, also not supported by
22 any evidence. Dr. Reinman's expert opinion is
23 unrebutted by any other expert.

24 Dr. Reinman also testified prioritization,
25 and also found that the prioritization works if, you

1 know, application developer writes an application and
2 can -- the patched new Mali driver has no effect on the
3 prioritization configuration of the shader board.

4 So where does that leave us? Actually, just
5 one more thing before I summarize. I'm going to share a
6 different screen. So this is the '381 patent. Mr.
7 Higgins mentioned that, you know, because the patch
8 vertex shader program that is a compute task, it is now
9 part of -- because it's really part of a vertex shader.
10 We shouldn't consider it to be a compute task. Even if
11 the EOE branch were to consider Realtek's arguments in
12 this regard, and that they have not been waved because
13 Realtek failed to make those arguments in the ITC
14 proceeding.

15 Mr. Higgins' opinion that for something to
16 be a compute task it needs to be part of a label's
17 compute shader, is completely divorced from the patent
18 itself. The patent itself actually makes no mention
19 whatsoever of a compute shader program. It only
20 references three types of shader programs. A vertex
21 shader, a pixel shader, and a geometry shader. Those
22 are the only three types of shader programs.

23 Advent of having a separate labeling, a
24 separate type of program, and calling that a compute
25 shader is something that came after the '381 patent.

1 It's not something that is discussed in the '381
2 patent. In fact, in the '381 patent, the way in which
3 it is distinguished whether or not the task is a
4 compute task is there's a special circuit inside of the
5 device called a vertex analyzer.

6 This is a vertex analyzer 208. And the way
7 it works is that the shader programs come in, the three
8 types of shader programs that are identified. They go
9 through this MUX, and then the vertex analyzer will
10 determine based upon its examination and analysis of
11 the shader program, is it a compute shader? Is it more
12 of a compute task, or is it more of a graphics
13 processing task? And depending on that, the output of
14 that decision, it will either send the task to the scan
15 converter if it's a graphics processing unit. Or, if
16 not, it'll send it on to the shader board. And that's
17 described in the highlighted section that I have here
18 of the '381 patent.

19 So I'm on column one. This is line 49
20 through 50, 53. This is the first time the term vertex
21 shader is actually mentioned. And here it says to
22 perform a single graphics processing task, GPU may
23 execute plurality of different threads. Each thread
24 may comprise the shader program such as the geometry
25 shader, pixel shader, or vertex shader, or the like.

1 There is no mention of a compute shader.

2 The only other mention of vertex shader in
3 the patents or any type of shader, specific type of
4 shader program in the patents is column eight, line 34
5 through 38. And it says, for example, processing
6 elements 220 may implement plurality of different
7 shader programs, e.g., geometry shader, vertex shader,
8 pixel shader, tessellation shader, or the like, to
9 complete a single graphics processing task by the OS
10 scheduler.

11 So this construct of compute shaders just
12 simply, you know, that construct wasn't necessary at
13 the time. The vertex analyzer, which is actually
14 described in column seven, line 50 through 57, performs
15 much of the same function that Realtek's compiler, it's
16 internal compiler performs in that it examines the
17 shader program that comes in, determines what types of
18 tasks are in that shader program, and then splits them
19 out and sends them to the right place. That's basically
20 what the vertex analyzers doing here.

21 So the vertex analyzer 208 sends graphics
22 processing, in fact, across the -- I'm sorry. I
23 apologize. Oh, I see. It's just not highlighting now.
24 So vertex analyzer sends graphics processing tasks to
25 the scan converter, then sends general compute tasks to

1 the arbitrating logic scan converter. I apologize. So I
2 do have the wrong site.

3 So we should be looking at column seven,
4 line 41 through line 46. And it states vertex analyzer
5 identifies shader programs associated with the graphics
6 processing and/or general compute tasks and schedules
7 in shader program could be launched in a shader core
8 based upon input and output data.

9 So what happens is that it comes into the
10 vertex analyzer. It makes the decision of what type of
11 task it is, and then sends it to the right place. It
12 doesn't require a special compute shader program to be
13 a vertex shader that has compute tasks in it. And
14 that's a decision that's made by the person analyzing
15 it.

16 So with that, I'll just go back to providing
17 a quick summary, and then I will open it up for any
18 questions that you might have.

19 So where does this leave us? So starting
20 back up at the top, the EOE branch can dismiss
21 Realtek's petition based on two very simple grounds.
22 The chips, there is no new Mali driver. For the
23 Realtek development boards, they've abandoned the only
24 two boards where they could even potentially have a
25 shot of arguing disablement based upon the boot lock

1 code, and that's for good reasons, because it didn't
2 work. And then as to board number one, which has no
3 boot lock code, they haven't shown that the
4 functionality of the Mali GPU, which hasn't changed at
5 all, is reasonably inaccessible such that it should be
6 considered disabled under the applicable legal
7 precedent.

8 So just based on those two issues alone, the
9 EOE branch can dismiss, and not issue a ruling request
10 as requested by Realtek. But even if the EOE branch
11 were to consider the arguments that Realtek presented,
12 the new Mali driver just doesn't operate as described.
13 And there's a fundamental failure of proof because
14 Realtek's experts failed to consider the entirety of
15 the code of the new Mali driver in the performance
16 enhancement patch. And they have failed to provide an
17 adequate claim construction.

18 And Dr. Reinman provided and analyzed the
19 actual complete code of the new Mali driver and
20 performed the assessment. He found that it does
21 continue to infringe based upon functionality that was
22 not present in the legacy products for any of the items
23 that are listed on this slide, summary slide number
24 eight.

25 So in sum for those reasons, and we

1 respectfully request the EOE branch to deny Realtek's
2 request.

3 MR. TERRILL: Thank you for that. This is
4 Dax Terrill. Just one question. I guess this is sort
5 of a summation question. In looking back at your
6 presentation, I'm trying to get AMD's primary response
7 and theory of infringement in reply to Realtek's
8 argument that the product -- the articles at issue are
9 not infringing because they don't meet claim
10 limitation. They don't meet the limitation claim 19.
11 But the first task comprises a graphics processing
12 task. The second task comprises a general compute
13 task.

14 I mean, perhaps the answer is that there are
15 multiple theories of infringement and responsive
16 arguments. But I am trying to get a sense of what is
17 the primary argument, and let me know if I'm incorrect
18 about this. But it seems to me the primary argument as
19 found on your slides, it seems that it's slide
20 number -- I would say it starts at really slide 21, and
21 continues on to slide 23. Is that the heart of your
22 argument in response and, if so, would you lay out the
23 position once more to confirm my understanding?

24 And if I'm missing where the heart of your
25 argument is, could you point me to it? And the slide

1 numbers again were 21 to 23.

2 MR. RIZK: I apologize. I muted myself. Are
3 you able to see my screen, Mr. Terrill?

4 MR. TERRILL: Yes.

5 MR. RIZK: Yeah, so this is the principal
6 argument for the Realtek integrated circuit, and the
7 Realtek development board which is that the
8 functionality is baked into the hardware. And they
9 haven't shown disable -- the new Mali driver's not
10 present in the Realtek integrated circuit, and it is
11 not disabled with respect to the Realtek development
12 board, because it is not a closed system. It's an open,
13 an open system that is intended for modification by end
14 customers. That is the principal argument.

15 But if we're to consider the merits of the,
16 you know, their argument that the patch actually
17 modifies this configuration, which we disagreed with,
18 and that it disables the configuration, then we believe
19 that the performance enhancement patch needs to be
20 considered in conjunction, because that's what Realtek
21 furnished to AMD on the -- on the physical sample that
22 was ordered to be produced in this proceeding. On the
23 article that was ordered to be produced in this
24 proceeding. And that included the performance
25 enhancement patch.

1 And when AMD's experts tested that, that
2 article that was furnished in response to AMD's request
3 for article that has the exact configuration that
4 they're seeking to input, what AMD found is that it the
5 new Mali driver continued to allow compute shader
6 requests in the form of these vertex pilot shader jobs
7 to come in and to be executed concurrently with
8 fragment share requests. And that's found on slides 42
9 through -- 42 through 48.

10 So that's what this slide is saying is that
11 if we take -- we ignore the fact that the hardware
12 configuration is not disabled in the Realtek IC, and
13 it's not disabled in the Realtek development board, and
14 we just take on the merits of their argument that the
15 new Mali driver could conceptually disable it, what we
16 found is that our experts provided unrebutted testimony
17 that they just shifted vertex pilot shader jobs, which
18 is a type of -- it was formally part of legacy compute
19 jobs.

20 So when the commission found the compute
21 jobs contained compute tasks, included in that were
22 these vertex pilot shader jobs. Without making any
23 change for the vertex pilot shader jobs, in the new
24 Mali driver, they moved them to become part of vertex
25 jobs. So now vertex jobs instead of just merely

1 containing vertex graphics processing tasks, they also
2 contain compute tasks in the form of these vertex pilot
3 shader jobs.

4 And we go back to our diagram that we were
5 looking at, these -- even if we, again, accept that the
6 new Mali driver actually modifies the configuration,
7 the new Mali driver that they put on their physical
8 sample that they gave to AMD modifies the
9 configuration, that configuration still -- it actually,
10 the driver actually forces vertex pilot shader programs
11 to be executed simultaneously with these graphics
12 processing tasks in the form of fragment shader
13 programs. It forces them.

14 So our position is that even with the
15 driver, it's still -- even if it was hypothetically
16 modifying the configuration, which it isn't, we believe
17 that it still infringes.

18 And then finally there's a failure of proof
19 issue. And I apologize. There are three principal
20 answers in response to your question, Mr. Terrill. But
21 there's a failure of proof issue for claim 19 in that
22 Dr. or Mr. Larri, rather, testified that even, even
23 using -- even if we say that the -- that the
24 performance enhancement patch is not part of the new
25 Mali driver configuration, which is what they gave us

1 on the physical sample, even if we were to accept that,
2 then where we would end up is they still haven't shown
3 what are the applications, and what is the source code
4 for the applications that are included inside of the
5 new Mali driver, excluding the performance enhancement
6 patch.

7 And as a result of that, there is no
8 evidence of record as to whether or not, even though
9 the legacy products may very well have -- the
10 applications may not have had applications that
11 implement general purpose compute within vertex jobs,
12 that doesn't mean that the applications in this case
13 are the same. The Realtek hasn't shown that they're the
14 same.

15 And what we have is we have Mr. Larri, who
16 testified in this proceeding. I don't have the exact
17 cite. But Mr. Larri testified in this proceeding
18 consistent with Dr. Reinman that applications that are
19 on the device could, in fact, submit general compute
20 tasks as vertex jobs.

21 So without the source code, which Realtek
22 hasn't produced, and without any explanation of how
23 these applications are the same or different from the
24 applications in the ITC, they haven't met their burden
25 of proof.

1 MR. TERRILL: Thank you.

2 MR. COOK: Can I chime in there? I guess is
3 AMD presenting a specific claim construction for, I
4 guess, a general compute task as it's claimed in claim
5 19? Because it seems like one of the disagreements
6 between the parties is that Realtek doesn't think it's
7 a general compute task, but AMD does under their
8 construction. So I was just wondering what that might
9 be? Is it a task not being a part of the general
10 graphics pipeline? Like could you please respond to
11 that?

12 MR. RIZK: Well, our first position, Mr.
13 Cook, is that there's a waiver issue here. So at the
14 ITC, we're looking at the ID page 124. And the ALJ
15 noted that what AMD pointed to as the compute tasks,
16 the alleged general compute tasks were compute jobs.
17 I'm going to read the quote and the parenthetical,
18 which include tasks that are not specific to graphics
19 processing.

20 And it is undisputed that vertex pilot
21 shader programs were at least a subset of those tasks
22 that were encompassed within the ID's decision. Realtek
23 did not come out and argue, oh, well, those, you know,
24 those types of compute job tasks don't qualify as
25 general compute tasks. And, therefore, there's no

1 infringement. They didn't make that argument. If they
2 wanted to make that argument that those are not compute
3 tasks and, therefore, don't satisfy the claim, they
4 should have raised that at the ITC. Their remedy at
5 this stage wouldn't be for to get a claim construction
6 as to whether or not they qualify. But the remedy would
7 be to ask the ITC to initiate some sort of a
8 modification proceeding.

9 But we believe that at a minimum, the
10 construction should encompass computations that such as
11 what were adjudicated to infringe at the ITC which
12 wouldn't necessarily include the vertex pilot shader
13 programs. Whether general compute tasks can encompass
14 other things, I think that there's significant guidance
15 and examples provided in the specification of the '381
16 patent.

17 The '381 patent specifically identifies
18 physic simulations. As an example, like the physic
19 simulation that Dr. Reinman ran as an example of a
20 general compute task. But generally, Dr. Reinman in his
21 expert report has offered an opinion as to how a
22 placebo would apply. It has a plain and ordinary
23 meaning. We don't think there is a special definition.
24 And we think plain and ordinary meaning is appropriate
25 in this circumstance, and we're not opposed to a

1 specific instruction. But it should encompass at least
2 two examples. One example of what the ITC found to
3 infringe, vertex shader programs, as well as what is
4 specifically listed as examples in the patent such as
5 physic simulations.

6 MS. WU: Yes, I'd like to just add, add onto
7 that if that's all right, Aaron, or did you have
8 anything -- did you want to add something first?

9 MR. COOK: Please go, Jess.

10 MS. WU: Oh, thank you. So I understand that
11 that is -- you know, based on the construction that AMD
12 would take regarding general compute tasks, you know,
13 this would involve, as you were saying, the plain and
14 ordinary meaning of the task, performs general
15 computation, mathematical algorithms and physics
16 simulations, and those are cited in the '381 patent.
17 However, in the claim language it says, you know, it
18 specifically denotes that these are two distinct
19 categories of tasks.

20 So my question will be where would you draw
21 the line? For example, between the graphics processing
22 task that -- or a general compute task that performs,
23 you know, physic simulations, but is, you know,
24 somewhat related to the graphics processing pipeline,
25 like how far removed would it have to be from being a

1 graphics processing task versus just, you know, a
2 computation that's entirely divorced from the graphics
3 processing process, if that makes sense?

4 MR. RIZK: Yeah, that's a great question. I
5 don't know that there really is a bright line, Ms. Wu,
6 and so in some ways we have to go off of examples. I
7 will say that the types of program, like vertex pilot
8 shader programs, if you read them. And, unfortunately,
9 we tried to get more discovery from ARM on what they
10 actually are, and we weren't able to get that
11 discovery. So we think that has to be part of the
12 process.

13 But what I would say is that we understand
14 that those -- from the code, that those types of
15 programs, what the code will do is it'll peel out
16 computations that are not performed on what's called a
17 per vertex basis, but rather let's say if you have a
18 program that is doing graphics processing specific
19 calculations, but then somewhere in the code it says
20 multiply X by Y. And X is equal to four, and Y is
21 equal to six. All right. And it just has those two
22 constants, and we just need to perform a general
23 computation like with a calculator of what is the
24 answer to this general computation.

25 They call those -- in the patent they're

1 called uniforms or constants, so you're multiplying two
2 constants together. That's generally not considered.

3 MR. COOK: Okay. I missed that first part
4 of it. Could you just repeat that first part before
5 constant.

6 MR. RIZK: Yeah. Did I say uniforms?

7 MS. WU: Okay, yeah.

8 MR. RIZK: So they're referred to kind of
9 interchangeably in the industry as uniforms or
10 constants. It's more or less the same thing whether
11 it's a uniform or a constant. But if it's a calculation
12 like that where you're just working with uniforms or
13 constants, something that you could probably do on a
14 calculator, right? That would generally be considered
15 a general purpose computation, even if it's going to be
16 used down the road to, you know, put into some process,
17 later actual graphics processing tasks.

18 But to answer your question about what is
19 the specific type of thing that you could point to and
20 say, well, that would be a graphics processing task.

21 So a vertex shader, if you look at the
22 Kronos specification for what a vertex shader helps
23 define, it has a very special type of input
24 requirements, and a very special type of output
25 requirements. So a vertex shader generally performs a

1 special type of graphics processing computation called
2 a per vertex operation. Fragment shaders or pixel
3 shaders, by the way, have their own language which is
4 called per fragment operations including a process
5 called scan conversion.

6 But what I'd say is if a program only has
7 the required calculations of the Kronos vertex shader,
8 and it's not doing general calculations in addition,
9 then -- then those required calculations will be
10 considered graphics processing tasks. That's the way I
11 would see it.

12 But that's not what these vertex pilot
13 shader programs do. They do general computations. They
14 don't do it on a per vertex basis. They don't use any
15 the fixed function hardware. And instead, just very
16 much like the vertex analyzer that we saw in the
17 patent. There's a decision that's made. In this case,
18 the decision is, well, this is just performing, you
19 know, computations on uniforms and constants. So we're
20 going to -- this is really a general purpose
21 computation. We're just going to send this to the
22 floor. And that's what the GPU does, and the driver
23 does, regardless of what we call it a legacy compute
24 job, or a patch vertex job. Does that answer your
25 question?

1 MS. WU: Yeah, I think that certainly
2 clarifies, clarifies it. Thank you. I also had another
3 question. So regarding what you were discussing before
4 about temporarily, you know, priorities, so you
5 mentioned that temporarily enabling any one job, such
6 as the fragment job to substantially fully consume the
7 resources of the GPU, is an example of a priority. But
8 how does this tie together with the fact that based on
9 the claim language of claim 20, the shader core's
10 configured to receive the plurality of tasks based on a
11 priority rather than providing a priority to a task
12 during execution of that task?

13 MR. RIZK: It's an excellent question,
14 Ms. Wu. And I admittedly did not go over it in detail.
15 The reasoning is set forth in detail in Dr. Reinman's
16 report. But let me try to do a better job of walking
17 through that particular component of it. So let me try
18 to pull up a version of the claim here, if you will
19 give me a moment.

20 Here we go. Okay. So you're correct. The
21 shader core has to be configured to receive the
22 plurality of past and at least one of the plurality of
23 engines, and to execute the first task from the
24 plurality of tasks while executing a second task from
25 the plurality of tasks based on the respective

1 statement here.

2 So if let's say, for example, compute jobs
3 are being executed. Actually, let me go to the
4 demonstrative slides. Let's say, for example, compute
5 jobs are being executed. Give me a moment to talk. If
6 compute jobs are being executed, then while those
7 compute jobs, number one, not all of them are
8 necessarily in the cue. Not all of the compute jobs are
9 in the cue at the same time. Some of those compute jobs
10 can be held in earlier stages of the pipeline, and not
11 received by the shader core, while others are actually
12 in the shader core execution by one.

13 So the only types of jobs that will be
14 received by the shader core in that circumstance would
15 be compute jobs, and so entire, what they call -- it's
16 called a job chain. The entire chain of jobs associated
17 with that with those compute jobs are actually the
18 complete -- the execution. And then only after that
19 point would the fragment jobs, attached fragment jobs
20 that are held up by the driver be released into the --
21 into the GPU, and into the shader.

22 And as a result of that, the shader core is
23 going to be configured to receive, if we can consider
24 the driver as being something that actually configures
25 the shader core. The shader core is configured to

1 receive those tasks based upon a priority. I apologize.
2 Taking a little while to get this demonstrative which I
3 think will be helpful.

4 Yes, this is what I want to show. So you can
5 see here in this demonstrative we have represented with
6 the orange dot the fragment jobs. Represented with a
7 blue dot is the blue cross is the patched compute jobs.
8 And what we could see is that the fragment jobs are
9 flowing through, right? And because of this, the
10 function of the driver, these pluses, the patched
11 compute jobs are being held up, and the shader core is
12 receiving the tasks based upon the priority associated
13 with we're going to allocate this resource just to
14 fragment jobs at this moment in time. Does that make
15 sense, Ms. Wu?

16 These fragment jobs are going to become
17 again serially until into the shader core, and
18 therefore receive based upon the priority of the
19 exclusive access to fragment until fragment jobs have
20 completed, executed. And, therefore, the receive based
21 upon priority.

22 Similarly when the shader core eventually
23 receives the patched compute job, it will receive that
24 based upon the priority of the compute jobs were held
25 up and delayed. And they only receive it at that moment

1 of time, you know, end plus 20 later, because of the
2 serialization functionality of the new Mali driver.

3 MS. WU: Okay. Yes, I think I understand.
4 Thank you very much for answering my questions.

5 MR. COOK: Are there any other questions
6 from the EOE branch?

7 MR. TERRILL: Nothing else me.

8 MR. COOK: So seeing as there are none, we
9 will take a 10 minute break, so please plan to be back
10 at 4:45. Thank you.

11 MR. TERRILL: And our questions, thank you
12 to counsel for AMD for response to the questions. I
13 think we took about, by my count, about 15 minutes of
14 extra time, so we'll give 15 minutes in addition for
15 Realtek's rebuttal. Thank you.

16 (Break at 4:35 p.m.)

17 (Back on the record at 4:45 p.m.)

18 MR. COOK: It is 4:45. Is Realtek back?

19 MR. HIGGINS: I am back, yes.

20 MR. COOK: Thank you. Is counsel for AMD
21 present?

22 MR. RIZK: Yes, I'm here. Thank you, Mr.
23 Cook.

24 MR. COOK: Thank you. So at this point,
25 Realtek will have 45 minutes for rebuttal, and we'll go

1 ahead and give it over to you.

2 ORAL REBUTTAL DISCUSSION BY MR. HIGGINS:

3 MR. HIGGINS: Thank you. And I'm really
4 hoping I don't need all that 45 minutes. I'll try to
5 spare everyone the time. But I do want to hit on a few
6 points that Mr. Rizk covered. And the first point I
7 want to start with, and maybe I can share my screen
8 again to make this go a little easier.

9 So I want to start with this argument on the
10 scheduling module, and whether this claim can be read
11 solely on hardware just looking at a GPU. And Mr. Rizk
12 went through the patent invention. The hardware is the
13 sole, the point of invention here. That's really what
14 matters.

15 But if the hardware, just the hardware alone
16 was the main invention of this patent, and the sole
17 point of novelty, then why did AMD have to amend the
18 claim during prosecution to add the scheduling module
19 and the second processing unit? If it didn't add that,
20 the claim was invalid over the prior art.

21 It's not the hardware here we're talking
22 about. It's the scheduling model. That's the key part
23 of this claim that was added during prosecution. And
24 that's what we modified. That's what the commission
25 and the ID found was required as part of this claim.

1 The claim doesn't say engines configured to receive
2 tasks. They specify it has to be received from the
3 scheduling module specific thing associated with a
4 second processing unit. Something else.

5 The ID found these are limiting. These have
6 to be met to show infringement. They were met and
7 argued by AMD to be met by the Mali driver code. And
8 simply because the ID in its ruling said that the
9 computing system was not limiting, that doesn't change
10 this fact.

11 The ID said computing system is a
12 limitation. We don't need to further determine whether
13 this scheduling module is within the second processing
14 unit, or actually where they are. All that matters is
15 they are there and present. They don't have to be in
16 this computing system specifically. They don't need to
17 be within the first processing unit. That doesn't mean
18 they aren't required, and AMD's actions during
19 prosecution from what it argued to get over the prior
20 art means that it is required, and it must be shown.
21 And our modification of this disables the configuration
22 that this claim is directed to.

23 And Mr. Rizk made a point about the DI
24 portion of the case, and how AMD didn't have its driver
25 code to show whether it met the claims. And that means

1 that the ID read it solely on the hardware. That's
2 absolutely not correct. What the ID relied on, and this
3 was on page 131 of the ID, was that there was other
4 evidence in the case. There were AMD documents, AMD
5 testimony. And it walked through those documents
6 showing that there was a scheduling module that was
7 associated with the second processing unit.

8 There was quote from the ID that says
9 another document shows the DRM scheduler code is within
10 AMD's driver. It's on page 132 of the ID. The ID
11 referenced that because that met the claim scheduling
12 order. The ID also said nor is there any assertion that
13 they, the AMD GPUs can operate without a CPU, second
14 processing unit, under any circumstances.

15 The ID didn't say these weren't required or
16 didn't have to be shown. It specifically walked through
17 the elements for both DI and infringement, and said
18 this is a requirement. This has to be shown.

19 Mr. Rizk pointed to a few cases. I believe
20 he mentioned Silicon Graphics. Silicon Graphics doesn't
21 use the language configured to. Neither did the INVT
22 case that it cited to. But interestingly about the
23 INVT federal circuit case is that that claim was
24 directed to capability of a hardware processor. But
25 that case held that there must be a showing that the

1 hardware is able to perform the function. Designed to
2 perform the function in some circumstance.

3 Here there's absolutely no showing. It's not
4 disputed the GPU can do nothing without the driver.
5 The driver is a part of that configuration of the GPU.
6 And, frankly, if it was enough for a complainant to go
7 through an ITC investigation, have a claim, point to
8 certain device drivers, software as being required, get
9 infringement, and then come to the branch and argue
10 none of that's required. Doesn't matter what you do
11 with those required claim elements. It only requires
12 what's in the hardware, that would have far reaching
13 effects. It would also require that the Apple and
14 Diebold decisions wouldn't have been correct. But
15 that's simply not the case. AMD is just wrong on this.

16 And were that not enough, AMD is arguing in
17 its IPR, or in our IPR we filed against the '381
18 patent, that configure to means, Actually programmed or
19 equipped with software to perform the recited function.
20 And we've quoted this on page 23 of our reply brief.
21 The section that AMD had in its IPR quoting several
22 federal circuit cases, including Mitek versus TIS
23 America and Electro Ventures, several cases, that this
24 is what configured to requires. Well, if you don't have
25 the driver code, you haven't programmed anything the

1 GPU isn't capable.

2 And, again, this is all going to the point
3 that you have to get past the Apple and Diebold
4 settings. We have disabled what was found to infringe.
5 If you buy into AMD's argument that, well, we need to
6 get into the scheduling module. Well, they lose still
7 there, because it's required as part of the claim.

8 So however you kind of chop this up, AMD
9 should lose on multiple fronts. But the branch doesn't
10 need to get to this claim construction, frankly, the ID
11 and commission already did. Scheduling module is
12 required, because we have disabled the features that
13 were found to infringe. It's been approved by the
14 branch before, at least in the Apple case and Diebold,
15 and several others we cited to. That's all the further
16 we need to get.

17 And no part of Mr. Rizk's presentation that
18 argued that the patches as we applied them didn't do
19 what we said they did. They disabled the specific
20 features found to infringe, and that's what's relevant.

21 And Mr. Rizk spent some time talking about
22 operating system source code and how they apparently
23 needed the operating system source code to make some
24 kind of a determination here as to whether these
25 patches work. That's all irrelevant. It wasn't required

1 below. They didn't cite any of it. They didn't analyze
2 any of it.

3 But as I stated, and I believe this is on
4 slide 72 here. It doesn't matter. Whatever is in the
5 application software, by virtue of the patch that's
6 applied in the Mali GPUs, it will ignore any requests
7 for priority that comes in. A programmer can write in
8 whatever it wants, and the operating system and
9 application the patch driver's going to ignore it.
10 Everything is set to the exact same value.

11 But the fact that Mr. Rizk spent so much
12 time talking about the operating system source code,
13 application level, software, it kind of begs the
14 question why if they actually believed that just the IC
15 standing alone with no driver code or anything can
16 infringe, why are they complaining that they needed all
17 of this operating system code, application level code.

18 Their expert went through many of these new
19 arguments, including, I believe, the one on at least
20 here on slide 77, if not others, was directed to the
21 application level code. It's because the GPU standing
22 alone does not and cannot infringe. It's not subject to
23 the LEO. All of the analysis, and the ID the
24 commission, pointed to the Mali driver being a
25 necessary claim element, corresponding to a claim

1 element, because it configures the GPU.

2 All of the claims here recite configured to.
3 Federal circuit precedent is clear. That requires
4 something more than capability. You can't just point to
5 bare hardware and say, oh, it's capable, but that's
6 what they're trying to do here.

7 Figure two requires actual programming that
8 can perform the desired function. And we've never
9 argued it has to perform at each and every time. It
10 can't do anything else. That's not what the commission
11 held. It has to at least do it some of the time. But
12 that still means that that driver code meets beyond
13 there, because otherwise it can't do it any of the
14 time.

15 The next topic that I want to address is
16 claim 20. And there were a few things that Mr. Rizk
17 mentioned here. I'll stop sharing my screen. We don't
18 need that any more. But he put up a quote from our
19 expert, Dr. Pfister. I believe it was on slide 53 of
20 his presentation. I believe I can -- maybe I can bring
21 that up here so I have it.

22 I believe you should be able to see the PDF
23 we have of their presentation on slide 53. He pointed
24 to this quote from Dr. Pfister and said, doesn't
25 support our request. You know, you can't say there's

1 no priority. Well, if you look at the question, as
2 confusing as it was, he was asked whether he could say
3 there's no longer any priority anywhere in the Mali
4 GPU, anywhere in the RTL code, driver code, anywhere.
5 And he said I can't say that.

6 Dr. Pfister is very honest. He's a very
7 honest expert. His job during the investigation was to
8 rebut Dr. Reinman's opinions. He reviewed what Dr.
9 Reinman cited to. He based his analysis on our source
10 code expert. Our expert's job was not to sift through
11 the entirety of the ARM source code and RTL code
12 millions and millions of lines, and somehow try to say
13 there's no other priority anywhere.

14 Dr. Pfister was honest. He said I can't say
15 that. What he didn't testify to, and what his opinion
16 was in his declaration and throughout his deposition
17 was that everything that AMD and Dr. Reiman pointed to
18 for claim 20 is absolutely not a priority. And.

19 Dr. Pfister specifically went through the
20 patches, and said these eliminate the priority. This
21 one off quote for kind of a strange question, it
22 doesn't change the substance of what Dr. Pfister
23 actually testified to during the trial. Frankly, this
24 lacks credibility. It matters nothing to what the
25 actual opinions are in this case.

1 There are also a few cites to testimony from
2 Mr. Larri. I won't go through all of these. As you
3 could tell from the slides when Mr. Rizk was going
4 through them, they weren't complete quotes. They were
5 kind of partial quotes, partial answers. And they were
6 hypotheticals posited to Mr. Larri which he said, I'm
7 not sure. He would require testing to show that.
8 That's not a definitive answer if something is or isn't
9 in there. Mr. Larri was responding. He was given a
10 hypothetical, and he couldn't give a definitive answer
11 to it.

12 But, again, nothing in the '381 patent
13 differentiates or says that just because you have sole
14 access to some resource means that it's a priority. And
15 I believe it was on slide 74. I may have the slide.
16 Here we go. The slide isn't important. There was a
17 point in Mr. Rizk's presentation where he accused me of
18 leaving off the full quote. And that was with respect
19 to a quote from the '381 patent that talked about
20 exclusive access to the shader core. And I believe that
21 was on my slide 74, so I will share the screen again
22 and bring this back up.

23 All right. Mr. Rizk put up a slide that has
24 this similar quote in there, and he accused me of
25 leaving off the rest of the quote that shows up here

1 which starts, While temporarily enabling one of the
2 tasks to fully consume the resources of the GPU. I
3 went back and looked at the transcript. I specifically
4 quoted that part, and there was a good reason for that,
5 because the point I was making here, and the point the
6 '381 patent makes is sure, there may be circumstances
7 where one task in here, I had compute on the screen,
8 and I can flip it to fragment.

9 There are times where the '381 patent says
10 you can consume all resources of the GPU. But that does
11 not necessarily mean that that is based on a priority.
12 And the point I made was the '381 patent lists three
13 different things. Demand, maybe what jobs or what type
14 of tasks it's asking for. Priority, and/or preset
15 limits. Just because compute is the only job in here,
16 and that's what goes to the shader core, that's not a
17 priority. It could be a limit or a demand. Simply
18 because something is taking up the full, or most of the
19 resources of the GPU, that does not mean it is a
20 priority, and that is clear from the '381 patent.

21 And frankly here on the slide, it's either
22 compute or fragment. It's one or the other. Neither one
23 is being prioritized. Fragment is in job slot zero.
24 It's going to go into the GPU. But if a compute is in
25 job slot one, it will go to the GPU. There is no

1 prioritization here, because there's only one to pick
2 from. And how they get into the job slots, there's no
3 prioritization there either. There's a block, if one is
4 in there. But, again, that's not prioritizing. That's
5 something that's already been chosen.

6 There was another argument, and this goes to
7 one of the buckets of new arguments that AMD made. And
8 I addressed this in response I believe to one of Mr.
9 Terrill's questions at the beginning of my
10 presentation. And this is about the argument that you
11 start a job slot zero. And Mr. Rizk mentioned the FFS
12 function, and other modules that Dr. Reinman had cited
13 and to. And Mr. Rizk made the point that this is not
14 dictating what goes into the job slot. This is
15 dictating what comes out of it. That's just wrong, and
16 that's not what Dr. Reinman said.

17 His declaration at paragraph 154, he
18 specifically refers to this function as filling the job
19 slot. Not choosing what goes to the shader part. This
20 is just putting jobs in the job slots.

21 He further defines the Mali driver for this
22 function, determining which jobs to schedule to the job
23 slot. This has absolutely nothing to do with what gets
24 down to here in these functional units. Simply putting
25 two things in the job slot, what gets chosen after that

1 has nothing to do with the order in which they were put
2 in, and frankly there's no priority associated with
3 them. It's all about loading into the job slots. And
4 regardless, that walk-through, this works the same way
5 as it did before the legacy products. Nothing has
6 changed except when it gets the job slot two, there's
7 nothing there, because job slot two has been disabled.
8 The same functionality works the exact same way that it
9 did in the legacy products. There is nothing new here
10 for the branch to consider.

11 The next topic -- excuse me for a second. I
12 have to get a drink of water. Was the issue of general
13 compute tasks. And the issue here, and what Mr. Rizk
14 says, he was asked the question I believe by, Ms. Wu,
15 is there any bright line, you know, to draw. Is there
16 any line to draw between what is a general compute
17 task, and a graphics processing task. And didn't hear
18 one. I think he said there is no line.

19 But what he pointed to as an example of what
20 can tell you as to whether you may have a compute,
21 general compute task versus a graphics processing task,
22 he specifically mentioned constant. The constants are
23 general non graphics data. And that can tell you
24 whether you have a general compute task. And he made
25 reference to the pilot shaders, I believe. But there's

1 a big problem with that.

2 If you go to slide 56, and this was a slide
3 I had, and I've increased this quote to include more
4 from Jiao than what I had before. This is from Jiao.
5 It now goes to line 26. But the problem with Mr. Rizk
6 saying constants tell you it's general compute is this
7 is the Jiao prior art reference. That the only
8 argument AMD has that it doesn't disclose the claims is
9 that it does not disclose general compute tasks.

10 So if what Mr. Rizk says is AMD's position,
11 then they've admitted the claim is invalid because Jiao
12 processes constants with its vertex share. So I'm
13 guessing if it went back to Mr. Rizk and AMD, they'd
14 probably withdraw that statement. And the fact of the
15 matter is they have no answer as to how to
16 differentiate between general compute tasks and
17 graphics tasks.

18 And what the '381 patent says, and what we
19 walked through in my earlier presentation, it talks
20 about the shader programs. There are API level
21 graphics, graphics level API programs that were
22 standard at the time DirectX, OpenGL. Graphics
23 programs run graphic shaders. They operate on graphics
24 data, a vertex. And it doesn't matter one vertex or
25 100 vertices. It's operating on a vertex. That's a

1 piece of graphics data. It doesn't matter if you call
2 it a particle. It's still a vertex, because it's
3 positioned on the screen. The screen could have
4 millions of vertices. Each vertex is a position on the
5 screen. It has maybe lighting, some transformations
6 done. It may change the position a little bit. That's
7 all that's needed for the fragment process. You have
8 encolor what's on the screen.

9 This is standard graphics operations, and
10 there may be many or millions of sub operations in
11 there. Maybe you need to add two numbers, but it's
12 still being done in the graphics pipeline via vertex
13 shader.

14 Mr. Rizk also stated that compute shaders
15 weren't around at the time. There was no compute
16 processing. Compute shaders were, and compute
17 processing was around long before the '381 patent.
18 OpenCL, for example, which is all about doing compute
19 work on a GPU, was officially released earlier in 2009.
20 It was in development at least four or five years
21 before that. It was around long before the '381 patent.

22 And when it talks about general compute
23 tasks, the reason it's not mentioning the shaders is
24 because OpenCL was there. It was another general
25 computing thing that was understood, and it's going to

1 send in compute to the GPU.

2 Everything is consistent with what we said,
3 and with their arguments for Jiao. Vertex shader, it's
4 graphics. General compute, you're doing actual compute
5 work that's being given to the GPU.

6 And there was also some mention that with
7 respect to this, with the pilot shaders, and the
8 performance enhancement patch. And we knew this coming
9 in. We saw it in AMD's briefing that their argument's
10 going to be we didn't change anything. We just changed
11 the name.

12 And I told you that in my opening
13 presentation that that's what they were going to say,
14 and that's what we heard. But the problem is I showed
15 you the code. It wasn't just the name change. And we
16 walked through this. I believe it was in response to
17 another one of Ms. Wu's questions. There are job
18 descriptors, job handles, vertex job pointers. There
19 are other things that you need to pass along. This is
20 not just -- here, a draw called descriptor for the
21 vertex job. I think that tells you something there.
22 What is being run here in the performance enhancement
23 patch relates to the parent vertex job.

24 And that was true even in the legacy accused
25 profits. Even they were running something that did

1 these calculations, has a compute job, for whatever
2 reason ARM chose that was efficient at that time. It
3 still related to the parent vertex job. It was doing an
4 operation on vertex. And I believe it was mentioned
5 that somehow we had waived this argument that this was
6 a graphics processing task. It was never accused. Or
7 I'm not sure how we could have waived it.

8 What AMD had pointed to, the ID commission
9 adopted. Compute jobs running compute shaders were the
10 general compute tasks. And vertex jobs running vertex
11 shaders with the graphics task. The reason this didn't
12 come up is because the vertex jobs and compute jobs,
13 they shared the same job slot. So whether there's a
14 compute job anywhere in the chain other than in the
15 compute shader, wasn't relevant, because they were in
16 two separate job slots that could be submitted
17 concurrently. It didn't matter where it was. That
18 wasn't a disputed issue.

19 And I think the other thing that was telling
20 when the branch asked Mr. Rizk for what is AMD's
21 construction of a general compute task? He didn't
22 agree with their own expert, Dr. Reinman. When Mr. Rizk
23 said that Dr. Reinman offered unrebutted testimony at a
24 few points, but I think it's pretty telling when the
25 direct question is asked for an opinion that was

1 provided in the expert declaration, and there's no
2 agreement with their own expert. And it's probably
3 because he walked away with -- away from it in his
4 deposition, and he couldn't agree with it.

5 I think that's telling that all of these
6 different ways they're trying to finagle this around
7 and say, oh, this pilot shader is still a compute task.
8 It doesn't make any sense. They're walking way from the
9 patent, and they're walking away from everything else
10 that was said below in the investigation.

11 And the last thing that I want to cover has
12 to do with the scope of our ruling request, and what
13 was produced, and what we're asking for. And I'll stop
14 sharing my screen here. And I think to begin, we never
15 asked for a ruling on the demo board. That's to say
16 this specific demo board is outside the scope of the
17 LEO. The demo boards didn't even exist when we filed
18 our ruling request. Our ruling request was very clear.
19 It's directed to ruling on a design. The same as Apple
20 did with its Apple watches.

21 And our request was directed to termination,
22 products using a modified device driver that, one,
23 prevents the concurrent processing of general compute
24 tasks, and graphic processing tasks in the shader core
25 of the GPU. And, two, prevents the shader core of the

1 GPU from receiving tasks based on a priority associated
2 with each of the tasks.

3 That was our ruling request. That was based
4 on the two ARM patches that specifically disabled those
5 two elements that were found to infringe from the
6 claims. We didn't ask for ruling on a demo board. We
7 didn't ask for a specific ruling on a stand-alone
8 Realtek chip. The Realtek chip doesn't have the driver
9 code. It can't do what the functionality was that was
10 claimed. It's not configured to do it. It's not capable
11 of doing it. That's why a ruling request is directed to
12 the functionality and the design as implemented by the
13 patch driver code.

14 And there was a lot of discussion, and more
15 that I thought I'd hear today about pickles. And I'm
16 not sure I followed all of what Mr. Rizk was saying
17 about the pickles. But it all seemed to relate to the
18 performance enhancement patch. It's all just a smoke
19 screen. I mean, whatever word he's trying to use for
20 it, and with these demo boards, okay, we produced a
21 demo board that had all of the patches on it. And if we
22 weren't ordered to do it, it was he said we should do
23 it, and it said we provide a sample of relevant
24 integrated circuits that incorporate one or more of the
25 device drivers that we were talking about. So we put

1 them up. But the fact of the matter is AMD didn't have
2 these demo boards during the investigation, yet they
3 made out its infringement case. They had them here.
4 They could test them, and they did. And they even
5 tested demo board three, which I'm not aware of
6 withdrawing anything, that had the boot, the one time
7 programmable boot key on it. They couldn't boot up the
8 demo board. It worked as intended.

9 I'm not sure what was wrong with us
10 producing that with our boot key, and how we'd
11 withdrawn it, but it didn't work. It worked exactly as
12 intended. We produced them so they could test them, as
13 they complained about. We maintained our position they
14 weren't relevant. We produced the patch code.

15 All of the experts could review the patch
16 code. It specifically tells you the lines of code it
17 applies to. That's what their experts made out their
18 infringement case on below. That's all that was needed
19 here.

20 And the last thing I want to point out, and
21 this is on the demo boards, there is absolutely zero
22 evidence that any customer, let alone TCL in this case,
23 could take its own device driver and just replace what
24 Realtek has. There's literally nothing in the record.
25 In fact, the ID specifically found that TCL or any

1 other customer that was noted, did not make any
2 changes. Do not make any changes to the driver. This
3 is all just pure speculation. It's not the fact that
4 they're allowed to. They don't do it. They don't modify
5 any of this. And Mr. Rizk put up one slide. I believe
6 it was their slide 30, and said here's a quote from a
7 Realtek employee saying that TCL can modify anything it
8 wants to. It specifically said TCL software provided by
9 TCL in the TV. It wasn't anything provided by Realtek.
10 It was a misquote.

11 So I think when you look at all of this in
12 sum, and what our ruling request was directed to, it's
13 the design that is implemented in these two patches.
14 That's what we want the ruling on. Sure, the
15 performance enhancement patch was there. That's
16 separate. If the branch decides that it needs to rule
17 on that, it's a separate ruling from this, you know,
18 parts one, two and three. But I think at minimum, and
19 what we see, and what we don't see from AMD's
20 responses, there's no dispute that our patches disable
21 specifically what was found to infringe in the accused
22 products. That's what our ruling request is directed
23 to, and that's what we asked the branch to grant in our
24 favor.

25 With that I'll stop if there's any

1 questions.

2 MR. TERRILL: No questions for me. Thank
3 you.

4 MS. WU: I have a quick question. So you're
5 saying that that would -- SO counsel for AMD previously
6 said that the Mali driver source code is available for
7 download. That directly seems to contradict what
8 Realtek says about only providing binary to its
9 customers. So is that correct?

10 MR. HIGGINS: So there are public versions
11 of a portion of the driver code. There's a kernel space
12 and user space driver. There's a kernel space driver
13 that's available online. That does not have any of the
14 settings that are in there. So when Realtek receives
15 this code, it may have to input some settings that go
16 into their products so it works with what they're
17 putting it on. So a user can't just go download and
18 flash something on there.

19 And the second part of that is let's say a
20 user has a demo board or a TCL television. The code
21 they have is in binary form. It's not like they can go
22 in there and easily -- or easily, if at all, swap out
23 the code. The code that's available online, you know,
24 you can go in there. You can review what the code is,
25 maybe write -- use it to write your shader programs or

1 something. But it's not something they can take and
2 compile, and put into the accused products. It simply
3 can't be done.

4 MS. WU: Thank you.

5 MR. COOK: So I did have a question. You
6 mentioned that there was a version of a demo board that
7 had the boot lock on it that was provided to AMD. Do
8 you have any details on when that was done?

9 MR. HIGGINS: That was done at the same time
10 as -- there were three demo boards. One, and then two
11 and three. I believe it was late February or early
12 March. Their expert tested all three of them at the
13 same time in the setup we provided. I believe it was
14 Los Angeles. And we provided -- the source code for
15 that was also on the source code computer.

16 MR. COOK: Thank you for that. Are there any
17 other questions from the EOE branch?

18 Okay. Well, thank you so much for attending
19 the oral discussion today. We thank the parties. And
20 we look forward to seeing the parties post oral
21 discussion submissions on May 10th. Are there any
22 additional questions for the EOE branch at this time?

23 MR. HIGGINS: Just one from Realtek. We
24 were wondering if the EOE branch was interested in
25 posing any page limits on this further briefing given

1 amount of briefing it's already received?

2 MR. COOK: I'll defer to Dax on this
3 question.

4 MR. TERRILL: As a general matter, we don't
5 impose page limits on the parties. Obviously, if an
6 argument can be made in 50 pages as opposed to 150,
7 we're open to that, but we're not going to impose a
8 limit on the parties in this proceeding. But thank you
9 for the question.

10 MR. RIZK: Adam Rizk on behalf of the AMD.
11 I did have two quick questions. So, first of all, I
12 believe, according to my records, that the deadline is
13 May 13th, not May 10th. So I just wanted to clarify
14 that if I'm mistaken. And then I have another follow-up
15 question.

16 MR. TERRILL: That's consistent with our
17 thinking as well, May 13th.

18 MR. COOK: There may have been an adjustment
19 in Mr. Terrill's April 18th email to the schedule, so
20 if that helps the branch confirm that date.

21 MR. TERRILL: I believe that's correct. The
22 date in question, I believe, is a Monday.

23 MR. RIZK: That's correct.

24 MR. TERRILL: Yes. That's my understanding
25 as well.

1 MR. RIZK: Okay. All right. Thank you for
2 that clarification.

3 And then the second question that I wanted
4 to pose to the EOE branch was as to whether there were
5 any particular areas that would be helpful for the EOE
6 branch parties to focus on in their brief?

7 MR. TERRILL: It's another good question,
8 but we generally defer to the parties to decide what
9 they feel are the most salient arguments to make,
10 positions to flesh out, and arguments to raise. Well,
11 at this point, arguments to reflect that have already
12 been raised previously. So but I think the oral
13 discussion was very helpful to the branch's
14 understanding. Seeing that memorialized in the
15 submissions, I think, would be helpful. But other than
16 that, nothing, no guidance from the branch on what
17 particular arguments or aspects the parties should
18 focus on.

19 MR. RIZK: Thank you, Mr. Terrill. And
20 thank you to the EOE branch as a whole for all of the
21 time and attention that the EOE branch has given to
22 this matter. We really appreciate it.

23 MR. TERRILL: Certainly. One question for
24 me, just a general matter, is it possible to obtain a
25 copy of the transcript prior to the post oral

1 discussion submissions being submitted to the branch?
2 If not, that's not a problem, but if it's an option I
3 think the branch would be happy to receive it
4 beforehand.

5 MR. HIGGINS: We can definitely get a copy
6 of the transcript and send it along.

7 MR. TERRILL: Thank you very much.

8 MR. COOK: Are there any other questions?

9 MR. HIGGINS: None Mr. Realtek.

10 MR. RIZK: Not for AMD. Thank you, Mr. Cook.

11 MR. COOK: Well, thank you again. And this
12 concludes today's discussion.

13 MR. HIGGINS: Thank you.

14 MR. RIZK: Thank you very much to the
15 parties.

16 (The Oral Discussion was concluded at
17 5:21 p.m.)

18

19

20

21

22

23

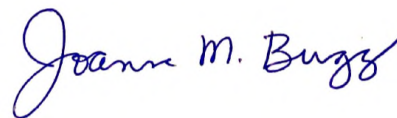
24

25

CERTIFICATE OF NOTARY

STATE OF MICHIGAN)
) SS
COUNTY OF WAYNE)

I, JOANNE MARIE BUGG, certify that this
oral discussion was taken before me on the date
hereinbefore set forth; that the foregoing statements
were reported by me stenographically and reduced to
computer transcription; that this is a true, full and
correct transcript of my stenographic notes so taken;
and that I am not related to, nor of counsel to, either
party nor interested in the event of this cause.



JOANNE MARIE BUGG, CSR-2592

Notary Public

Wayne County, Michigan

My Commission expires: 2-26-2025

&	136 98:9	1:03 1:18 5:3	2100 2:15
& 2:10,14 5:9 6:22	13th 169:13,17	1:51 100:21	220 129:6
0	14 116:13	2	22nd 100:20
02111 2:6	147 4:7	2-26-2025	23 132:21
04 15:10 38:14	15 12:12 14:4	172:24	133:1 150:20
05 15:10 38:14	32:3 56:21,25	2/23 100:23	231 120:3
1	64:23 65:8	20 11:15 12:6	232 120:3
1	67:12 74:24	13:25 14:17,21	25 116:13
1 1:19 5:2	76:3 79:10	15:9 20:20,22	2592 1:21
10 7:25 8:2	114:7 146:13	21:15 23:11	172:21
23:16,24 27:17	146:14	27:9 28:13,21	26 159:5
64:17 71:16	150 169:6	30:1,6 32:1	26742 172:19
111:2,3 116:22	154 157:17	36:23 49:12	27 30:4
146:9	15d 31:13	54:22 58:1	2:33 71:17
100 42:10	177 6:16	64:23 67:2	2:34 71:18
159:25	177-00024	79:14 80:15	2:45 71:17,19
10th 168:21	97:18	110:24 115:19	71:20
169:13	18th 169:19	122:6,16 126:2	3
112 91:13	19 6:16 11:14	143:9 146:1	3.5 116:12
114 124:20	12:6,9,12 13:6	153:16 154:18	3/19/24 125:3
116 93:9,11	13:10,25 14:12	200 19:8	30 7:25 8:1,3
12 23:24 27:17	14:16 15:9,15	20037 2:16	72:2 97:24
120:3	20:20 24:1,3	2005 72:20	109:2 166:6
124 137:14	27:8 30:4 32:3	2009 72:25	31 72:11 116:3
129 88:4	32:5,18,21	160:19	116:8
13 114:7 125:4	33:20 34:4,14	2013 73:22	337 1:8 6:18
130 88:10	38:3 47:14,15	74:5	80:14
131 149:3	64:23 79:9	202.339.8400	34 129:4
1312 86:16	80:15 120:3,3	2:17	38 129:5
1318 1:8 6:18	122:17 132:10	2024 1:19 5:2	381 11:14
84:23 98:4	135:21 137:5	100:20	19:18 44:19,23
124:12	195 114:7	208 128:6	45:6 48:6
132 149:10	196 114:7	129:21	58:12 59:19
	199 125:16	21 21:4 132:20	60:3,16 73:5,6
		133:1	73:20,25 74:1

80:16 127:6,25 128:1,2,18 138:15,17 139:16 150:17 155:12,19 156:6,9,12,20 159:18 160:17 160:21	5:21 171:17 6 6 97:24 109:2 61 69:14 617.542.6000 2:7 628 11:12 6676614 1:25	153:22 absolutely 38:15 59:9 149:2 150:3 154:18 157:23 165:21 accept 122:21 123:4 135:5 136:1 access 59:17 60:16 63:12 66:2 95:7,8,9 113:16 117:11 125:24 145:19 155:14,20 accessing 16:17 65:18 accompanied 22:13 accusations 39:21 accuse 106:8 accused 9:16 11:19 12:4,5 12:24 14:3,24 15:9,12 16:3 18:10 19:13 20:25 22:1,13 22:20 23:6,10 23:18 30:9 33:17 34:19 35:20 39:19 53:21,23 54:1 54:15,20 79:8 108:6,7 120:8	120:9 121:14 121:14,15 155:17,24 161:24 162:6 166:21 168:2 achieve 55:10 116:17 acknowledge 88:5 91:23 acknowledged 67:6 91:22,23 acknowledges 56:11 actions 148:18 actual 9:21 43:7 73:10 102:7 111:11 118:6 131:19 141:17 153:7 154:25 161:4 actually 16:22 22:22 48:23 55:14 79:2 89:19 91:14 93:9,10 94:13 99:6 106:4 107:3 109:6 112:14 113:18 116:9 118:21 118:25 119:15 122:16 124:8 124:11 127:4 127:18 128:21 129:13 133:16 135:6,9,10
4	7		
4 74:9 40 11:9 41 130:4 42 49:16,24 134:8,9 45 146:25 147:4 46 50:16 130:4 47 116:22 48 134:9 49 113:3 128:19 4:35 146:16 4:45 146:10,17 146:18	7 125:16 70 125:3 71 125:4 72 4:6 62:8 152:4 74 155:15,21 76 99:18 77 126:6 152:20		
	9		
	9 4:5 94 110:2		
	a		
5	aaron 3:3 6:24 64:18 139:7 abandoned 130:23 ability 15:11 59:5 74:17 84:17 114:17 able 64:7 98:20 98:24 103:21 105:22 115:17 126:3 133:3 140:10 150:1		
50 128:20 129:14 169:6 51 116:23 52 93:11 53 128:20 153:19,23 56 159:2 57 129:14			

140:10 144:3 144:11,17,24 148:14 150:18 152:14 154:23 adam 2:3 5:21 7:20 72:5 169:10 add 47:3 84:3 139:6,6,8 147:18,19 160:11 added 57:7,14 68:1 147:23 adding 62:20 64:8 addition 12:13 86:3 142:8 146:14 additional 49:11 52:25 53:4 56:3 110:4 168:22 additionally 8:12 address 10:11 27:11 41:6 51:14 58:2 65:4,7 66:5 71:5 77:3 153:15 addressed 88:4 112:21 157:8 addressing 10:14 17:22	adds 32:6 adequate 131:17 adjudicated 138:11 adjustment 169:18 administrative 6:14 admission 77:8 119:20 admitted 159:11 admittedly 86:14 143:14 adopted 13:2 13:12 14:21 21:3,21,25 23:9 29:10 30:15 32:24 33:15,21 46:2 56:6 72:24 73:23 162:9 adopting 33:5 80:4 adoption 74:4 74:23 advanced 2:10 5:22 7:4 advent 127:23 advisor 6:23,24 affect 20:12 affects 117:22 afternoon 5:12 5:20 7:15 9:5	age 120:4,14,19 121:1,1,2,3 ago 80:10 agree 46:13 68:8 89:20 113:1 114:24 162:22 163:4 agreed 77:22 80:13 114:2 119:25 agreement 6:25 24:8 110:14 163:2 agreements 96:21,23 agrees 55:15 111:23 ahead 6:7,12 24:3 27:15 64:20 71:16 147:1 algorithm 28:5 115:18 algorithms 139:15 alive 90:7 alj 11:12 26:8 68:21 77:21 78:9 80:12 88:3,10 92:4 116:3,7,8 124:20 137:14 allegations 10:17 91:12	allege 77:19 alleged 77:6,12 77:23 78:6 91:22,24 122:25 137:16 alleges 90:4 allocate 145:13 allocated 8:8 allocation 63:25 allotted 8:7,7 allow 37:2 57:22 74:18 82:11 115:4 134:5 allowable 44:23 allowed 108:18 166:4 allows 73:17 113:12 alluded 65:1 91:11 92:6 98:18 99:16 119:3 alternate 92:23 109:15 alternated 94:2 alternating 93:15 alternative 95:8 alu 47:1 alus 47:20
---	--	---	--

amazon 96:17	72:9,20,25	68:6 72:24	142:16
ambiguity	74:4 80:9,14	73:23 74:10,11	analyzers
100:12	90:4 92:4,6,16	75:24 76:1,11	129:20
amd 2:9 5:18	94:1,3,13	76:16 80:4,7,8	analyzing
5:19,25 6:5	100:11,22	80:16 88:2,14	130:14
7:19,21 8:1,22	102:25 103:5	88:25 89:16	android 96:13
9:24 10:7,8,14	104:14 105:8	97:7 100:17	angeles 168:14
11:10,12 13:1	106:7 108:6,7	101:6,8,22	answer 41:6
13:12 15:4	110:11 111:23	102:10,17	104:24 105:19
17:12,17,23	121:23 122:20	112:2 124:7	113:4 120:8
18:2 19:8,14	123:15 133:21	132:6 134:1,2	121:16 125:8
19:15,16,20	134:4 135:8	148:18 149:10	125:12 132:14
20:9,10,14	137:3,7,15	151:5 159:10	140:24 141:18
21:2,14,25	139:11 146:12	161:9 162:20	142:24 155:8
23:14,19,24	146:20 147:17	166:19	155:10 159:15
25:13 28:5	148:7,24 149:4	amdm 72:3	answered
29:3 30:11,14	149:4,13	amend 147:17	114:8
30:18 31:5,12	150:15,16,21	america 150:23	answering
32:10,19,24	151:8 154:17	american 72:10	146:4
33:3,7,14,21	157:7 159:8,13	amount 8:6	answers 135:20
34:2 35:25	162:8 165:1	126:21 169:1	155:5
39:1,18,21	167:5 168:7	amy 3:10 6:2	anticipate
40:10 44:1,19	169:10 171:10	analysis 33:20	56:25
45:1,16 46:2,3	amd's 10:4,16	99:12 121:17	anyway 39:21
46:8,16 47:7	15:17 21:5,9	122:19,20	api 159:20,21
47:13,16,24	21:19,22 25:6	125:19 128:10	apis 125:15
48:4,12,19	26:13,22 27:10	152:23 154:9	apologies 69:8
51:14 52:6	27:11 30:16	analyze 103:8	apologize
53:12,25 54:21	32:22 33:18	152:1	124:25 129:23
55:17,21,21	34:17,25 35:19	analyzed 99:10	130:1 133:2
56:7,10,25	38:8,21 46:16	101:7 131:18	135:19 145:1
57:22 59:16	47:21 51:19	analyzer 128:5	app 123:16
62:11 64:22	54:8,25 55:14	128:6,9 129:13	apparently
67:1,6 68:1,9	56:2,4,23	129:21,24	48:12 110:18
71:24 72:2,6,7	60:20 65:2,9	130:4,10	151:22

appearances 2:1	122:11,14,15 123:6,8,17,22	architect 74:9 architectural 74:3	92:16 94:1,3 108:11 148:7 148:19 151:18 153:9
appearing 2:10 2:19 5:19 115:13	123:25 124:2 124:13,14,18 124:24 125:10 125:23 126:4	architecture 50:11 73:1,4,7 73:14,16 74:1 74:11,13 80:5 89:11 92:22 93:7	argues 18:2 arguing 17:13 19:18,20 25:23 27:21 33:3 38:22 48:7 55:15 56:7 57:19 62:11 63:2 65:11 67:2,18 130:25 150:16
appears 50:6 apple 16:11,12 16:13 17:2 25:22,24 26:1 54:13 57:12 65:15,22,23 94:25 97:16 150:13 151:3 151:14 163:19 163:20	136:3,4,10,10 136:12,18,23 136:24 applied 12:4 14:10 15:23 51:17 70:1 104:14 112:8,9 117:6 151:18 152:6	area 13:19 areas 170:5 argue 19:10 23:24 39:5 47:21 56:9 57:15 58:3 62:18,19 63:14 63:14,22,23 64:7,7 77:17 93:4 108:18 137:23 150:9	argument 8:16 13:2 17:22,23 20:4,9 21:20 21:22 25:6,13 27:11 29:3,5 29:22 30:18 33:18 34:17 35:25 38:23,25 46:16 48:5,20 54:1,3,3,24 56:2,5,23 57:22 58:24 60:22,23 61:5 61:17,20,25 63:15 64:22 65:1,2,7 66:8 66:14,21 68:20 70:4,4,5 71:7 71:10,12 77:13 77:21 78:2 87:21 91:21
applicable 98:23 131:6 application 61:21,24 63:9 69:10,15 81:24 85:3,4,13 86:6 95:21,25 124:11 125:5 125:13,14,19 125:20,23 126:7 127:1,1 152:5,9,13,17 152:21	applies 121:11 165:17 apply 138:22 appreciate 170:22 approach 104:7 appropriate 39:9 138:24 approved 151:13 april 169:19 arbiter 64:3 79:22 92:7,8 92:22 93:3,14 94:2,6 arbiter's 79:24 arbitrating 130:1	argued 13:1,12 19:17 21:2,6 21:14,25 29:9 29:13 30:14 31:6 32:10,25 33:15,21 34:2 38:24 39:7 45:19 46:2 47:13,16,18 53:14 54:5,5 55:21,24 56:25 57:23 59:16 62:24 63:18 64:10 67:7 77:5 87:20	
application's 124:21 applications 62:10 82:7 85:21 94:4,14 95:5 121:18,20 121:21 122:1,4			

93:25 108:18 114:21 115:1 115:24 116:7 118:18 132:8 132:17,18,22 132:25 133:6 133:14,16 134:14 138:1,2 147:9 151:5 157:6,10 159:8 162:5 169:6 argument's 161:9 arguments 10:4,12 17:24 18:5 19:16 20:15,15,16 23:14,15,16,24 27:11,17 28:8 30:17 35:19 51:14,19 53:12 53:14,18 54:8 58:19 60:20 62:20 64:9 65:4 67:20 68:6 78:9 94:10 100:5 112:18 114:20 115:21 117:14 122:22 127:11 127:13 131:11 132:16 152:19 157:7 161:3 170:9,10,11,17	arizk 2:8 arm 10:24 12:17,23 13:4 13:14 14:5,13 15:7,10,19,24 15:24 21:22 22:11,12,15,16 22:21 32:11,20 32:25 33:1,24 33:25 34:9 36:24 37:20,21 38:14,15,16 41:10,16 42:8 42:17 43:25 44:1,3 45:12 50:10 53:11 83:18 84:15 86:10,11,18 95:7 105:16 107:6,12 109:7 110:12,16,19 110:20,23 112:13,17 114:12 115:14 140:9 154:11 162:2 164:4 arm's 109:1 110:6 113:25 115:14 119:24 arose 110:11 art 56:9,23,24 57:8,15 67:8 67:21,22 68:1 88:11 147:20 148:20 159:7	article 74:7 133:23 134:2,3 articles 69:22 132:8 aside 98:17 asked 46:10,21 50:1 96:21,22 96:23 97:23 99:13 100:15 102:3,5,19 104:23 105:16 110:5,18 112:25 114:7 114:12 119:7 120:21,23 124:10 125:4,9 126:13 154:2 158:14 162:20 162:25 163:15 166:23 asking 103:20 156:14 163:13 aspect 35:22 62:13 69:14 aspects 66:20 170:17 aspersions 75:23 assemblies 80:22 81:7 84:5 assembly 95:12 asserted 11:9 91:14	assertion 149:12 assertions 26:12 assessment 121:17 131:20 assigned 120:19 associated 29:14 45:3,5 56:11 57:1,9 66:25 67:4 68:24 90:18,19 116:6 120:14 130:5 144:16 145:12 148:3 149:7 158:2 164:1 assume 24:25 asynchronous 72:25 73:2,4,8 73:13,16 74:2 74:12,14 80:4 80:16 89:11 athlon 72:19 ati 2:11 5:22 6:5 7:4 atm 17:4,6,7 attached 113:13 144:19 attempt 8:10 46:12 attendance 3:1 attending 5:6 168:18
---	--	--	---

attention 17:21 170:21	73:22 102:14 109:11 130:16	40:2 42:2 45:15 48:6	53:20 56:2 70:4 87:13
attorney 6:23 6:24 91:6 112:18 115:1	130:20 132:5 135:4 146:9,17 146:18,19	53:19 54:1,12 56:4 57:4,16 58:15 59:23	100:2 109:8 140:17 142:14
attorneys 5:24	155:22 156:3	60:4,6 61:21	bat 111:22
attributed 124:12,13	159:13	63:2 72:21	bearings 49:3
attributes 49:18	background 72:7 81:17 83:13	73:14 79:16,21 80:15 84:22 92:17,21 93:13	because 57:6
authored 107:5 107:8	backtrack 17:23	98:15 100:3 101:10 102:24	beginning 15:25 55:4 123:15 157:9
authorization 80:8	backwards 20:21	105:8,23 106:12 110:15	begs 152:13
authorized 81:1	bag 83:22	111:16,19 112:16,19	behalf 2:10,19 5:7,9,12,19,25 6:5 7:13,17,19 7:21 9:6 72:6 107:7 114:1 169:10
avail 82:12	baik 3:15 5:13 5:13	113:11 115:13 120:4,10,12,12	behavioral 113:11
available 36:5 97:10 98:5 167:6,13,23	bailed 87:20	120:18 122:5 128:10 130:8	believe 19:21 25:22 50:2 57:11 59:9 69:11 104:12 110:8 111:8 133:18 135:16 138:9 149:19 152:3,19 153:19,20,22 155:15,20 157:8 158:14 158:25 161:16 162:4 166:5 168:11,13 169:12,21,22
avenue 2:15	baked 86:18 92:9,10 111:5 133:8	130:21,25 131:8,21 139:11 143:8 143:10,25 145:1,12,18,20 145:24 154:9 156:11 164:1,3	
avoid 10:13 47:13 100:12	ball 39:20	baseless 10:17	
award 85:16	barbara 3:12 6:4	basic 47:3 48:10 75:7 98:13	
aware 87:18 165:5	bare 153:5	basically 106:21 129:19	
b	barrier 72:18	basis 7:2 21:19 22:8 27:2	
b 97:24 109:2	based 10:16,16 11:4 13:1 15:6 16:14,19 18:9 18:16 19:7,12 21:14 28:7,14 28:25 29:4,5 31:5 32:1 33:15 35:19 37:19 38:24		
back 8:10 11:3 26:7,19 30:3 43:16 50:9,16 61:11,22 62:7 64:4 65:8,15 66:23 70:9 71:17,19,21,24			

believed 152:14	board 26:21	130:24 163:17	53:20 55:12
berth 75:20	39:18 52:11	164:20 165:2	64:9 69:6,10
best 88:24	80:22 81:7	165:21 168:10	69:25 76:18,21
better 143:16	82:8 84:4,5,5	boot 26:17,20	77:15 94:11
beyond 153:12	85:5,7,14,14,19	26:23 27:2	100:2,12,15
big 85:23,24	85:20 86:3	52:24 98:17,22	102:3,20 104:7
159:1	90:10 94:19	99:3,8,10,12,14	122:21 127:11
binaries 42:20	95:3,5,12,14,22	99:15,20	130:20 131:9
96:1	95:24 96:14,25	130:25 131:3	131:10 132:1
binary 16:23	97:2,20,25	165:6,7,7,10	146:6 150:9
25:11,17,18	98:21 99:2,3,6	168:7	151:9,14
52:11 82:2,4	99:7,8,22,22	booting 26:20	158:10 162:20
95:24 97:20,22	101:2,3,6,7,9	53:2	166:16,23
167:8,21	101:10,11,25	border 6:22	168:17,22,24
binding 25:8	101:25 102:19	87:2 90:9	169:20 170:4,6
bister 99:14	105:9 107:21	borrowed	170:16,20,21
bit 30:5 33:21	116:4 117:18	86:15	171:1,3
39:12 45:6	121:16,18,19	boston 2:6	branch's 28:7
46:4 66:1,19	122:11 123:18	bottom 13:17	54:6,12 101:1
81:17 160:6	123:23 125:24	14:22 31:12	101:8 170:13
bits 25:19	127:3 128:16	37:4 78:19	brand 19:10
112:6	131:2 133:7,12	115:2	break 8:1,2
black 95:13	134:13 163:15	boundary 83:5	64:17,18 71:16
block 47:1	163:16 164:6	box 11:22	71:18 72:18
86:14,16 92:15	164:21 165:5,8	12:18 13:4,15	146:9,16
157:3	167:20 168:6	13:16 39:3	breakdown
blocked 35:11	boards 18:18	55:7	13:4
36:6,14	18:21,23 52:4	boxes 33:13	breathes 77:25
blocking 60:15	81:8,11 82:11	35:15	brick 11:25
blocks 35:7,16	84:9,25 88:22	branch 6:20,22	26:24 89:8
blue 11:22	94:24 96:11	7:1,9 9:7 10:5	brief 15:1
12:18 13:15	98:1,21,25	10:12 16:9,19	17:13 32:24
33:13 35:14	99:2,18 105:13	25:25 28:12	45:2 46:9 72:7
55:7 95:16	121:25 122:7	30:7 38:8	76:1 150:20
145:7,7	125:21 130:23	51:15 53:10,17	170:6

briefing 9:9 10:19 17:17 19:8 20:5 23:15 38:8 80:12 161:9 168:25 169:1 briefly 10:18 30:13 31:16 83:12 bright 140:5 158:15 bring 18:5 153:20 155:22 brings 18:21 38:7 broke 42:8,11 brought 80:9 108:10 bucket 63:15 buckets 54:10 157:7 buffer 58:9,10 58:13 bugg 1:21 172:6,21 building 101:16 built 85:8 86:15 101:5 102:7 105:6 117:25 bullet 111:22 bunch 61:3 112:17	burden 113:6 114:14,18 123:5,19 136:24 burns 83:22 bus 68:3 75:1 75:20 business 1:13 100:22 buy 151:5 bypass 123:8	81:20 82:2,16 82:21 83:8,18 84:4 85:14 90:15 92:7 128:5 140:16 141:1 142:1,4 142:5 144:16 161:20 calling 41:19 41:20 43:1,12 43:13 127:24 calls 42:18 capabilities 47:19 capability 17:8 18:15 55:9,16 55:18 66:12 118:6 149:24 153:4 capable 55:13 55:19 78:25 151:1 153:5 164:10 care 96:13,18 carry 52:7 case 5:9 10:9 11:6,8 17:2 19:10,11 22:11 24:2 25:22 31:20 54:13 65:23 66:4 68:13,15,18 80:24 81:3 85:5,9 89:2 93:25 94:5	104:19 111:14 121:16 126:10 136:12 142:17 148:24 149:4 149:22,23,25 150:15 151:14 154:25 165:3 165:18,22 cases 87:1 149:19 150:22 150:23 cassette 86:23 categories 18:20 80:23 86:1 139:19 category 13:10 94:23 cause 92:24 118:3,7,8 122:5,16 172:13 cbp 53:19 center 2:5 83:15 central 72:16 cerny 74:8 certain 1:7 17:5 44:20 50:11,11 58:21 66:12 150:8 certainly 88:20 103:10 143:1 170:23 certificate 172:1
--	--	--	---

certification 25:1,2,8 52:6 52:21 certify 103:19 103:21 104:3 172:6 cfr 6:16 chain 37:1 83:14 144:16 144:16 162:14 challenges 104:6 chance 26:18 change 16:16 16:25 17:9 25:7,12 38:24 42:23,24 43:6 43:8,11 45:16 52:7 65:23 66:1 68:23 96:4 106:12 108:20 109:2 109:20 113:12 120:12 134:23 148:9 154:22 160:6 161:10 161:15 changeable 25:20 changed 16:22 40:25 41:25 42:17,25 43:6 52:13 70:8 71:9,12 76:12 87:9 91:1,10	91:25 92:11 109:9 131:4 158:6 161:10 changes 45:11 84:17 85:11 88:17 90:6 96:20 106:23 108:1 109:5 123:2 166:2,2 changing 43:7 48:12 channel 109:13 characterized 64:2 characterizes 45:13 chat 8:14 check 6:7 chen 98:9 chief 6:22 74:9 chiggins 2:18 chime 58:16 137:2 chip 53:2 77:17 83:8,23 85:2 85:18,21 86:15 86:19 87:3,5,9 89:12 93:8 95:6,15,16 96:1 100:25 164:8,8 chips 84:2 85:15 86:25 87:1 130:22	chooses 85:16 choosing 157:19 chop 151:8 chose 105:24 114:12 162:2 chosen 157:5 157:25 chris 5:11,11 7:16 9:5 christopher 2:13 circled 95:16 circuit 10:24 80:22 81:7 83:2,8,9 84:5 86:2 88:16,25 90:1,15 92:9 95:12 117:23 128:4 133:6,10 149:23 150:22 153:3 circuitry 83:23 circuits 80:21 81:6 84:2,11 84:12 87:15,21 100:24 164:24 circular 46:4,6 circumstance 138:25 144:14 150:2 circumstances 149:14 156:6 citation 45:1	citations 113:19 cite 92:5 136:17 152:1 cited 17:17 44:2,19 48:24 93:16 97:16,17 111:13 139:16 149:22 151:15 154:9 157:12 cites 155:1 citing 93:11 120:2 claim 12:9,12 12:21,25,25 13:6,10 14:4 14:12,16,17,21 15:15 19:15 20:15,22 21:15 23:11 24:1,3 27:8,9 28:13 28:21 30:1,1,4 30:6,22,23 31:6,12,14 32:1,2,5,11,18 32:21 33:20,25 34:4,14 36:23 38:3 46:1 47:14,15 49:12 54:17,22 55:5 56:21,25 57:3 58:1 64:22,23 64:23 66:14,15 66:16,25 67:2 67:11,11,17,19
--	--	---	---

68:1,9,18	57:20 74:21	20:1,3 21:12	140:14,15,19
69:17 74:15,21	88:9,9 104:23	22:14,22,23,24	148:7,25 149:9
74:24 75:3	111:25 115:7	22:24 24:7,8,9	150:25 151:22
76:3,17,19,22	118:16 121:11	24:11,18,19,24	151:23 152:12
77:6,25 78:1	148:25 153:2	24:24 25:3,10	152:15,17,17
78:20,20 79:9	159:8 164:6	25:16,18 26:1	152:21 153:12
79:10,14 80:15	clara 72:13	26:9,19 27:4	154:4,4,10,11
104:22 110:24	clarification	28:5 35:7,16	154:11 161:15
112:11 115:19	170:2	36:25 37:10,11	164:9,13
115:21,22	clarifies 143:2	37:12 39:14	165:14,16,16
116:2 117:13	143:2	42:24,25 43:2	167:6,11,15,20
122:6,16,17	clarify 169:13	43:16,25 49:5	167:23,23,24
126:2 131:17	classification	51:22 52:7,10	168:14,15
132:9,10	110:9	52:12,22 53:4	coded 13:3
135:21 137:3,4	classified 110:8	60:6,25 61:22	cohn 2:4
138:3,5 139:17	clear 32:15	63:5 65:21	coincidental
143:9,9,18	33:3 34:3	81:19,20,21,22	73:24
147:10,18,20	37:10 39:4	81:22 82:2,4,6	colleague 5:11
147:23,25	45:25 47:11	82:14,15,17,20	color 13:3
148:1,22	78:10 89:24	83:17 84:18,19	colors 49:2
149:11,23	113:17 153:3	90:23 91:7,7	column 116:13
150:7,11 151:7	156:20 163:18	95:7,8,23 96:3	116:22 128:19
151:10 152:25	cleared 36:9	97:9 98:11,22	129:4,14 130:3
152:25 153:16	clearly 43:21	99:15 101:25	combination
154:18 159:11	63:16 87:18	102:1,4,7,9,21	87:12
claimed 21:23	91:3 103:3	105:1,14	come 11:3
37:20,21 56:19	114:16	106:20 112:5	12:15 13:14,24
77:20 137:4	client 9:10	113:22 118:7	16:24 29:18
164:10	close 100:22	121:24,25	38:23 40:10
claims 11:9,11	closed 94:25	122:12,19	45:17 46:12
11:14,14 12:6	95:1 97:15	123:16,21	47:7 55:1
13:25 14:2	133:12	124:1,1 125:19	57:22 58:7,10
15:9 16:20	code 16:6,22	125:20 131:1,3	58:14 60:11
20:20,20 32:3	17:14 18:4,13	131:15,19	61:10 71:17
33:2 53:24	18:20 19:2,4	136:3,21	118:8 123:9

128:7 134:7 137:23 150:9 162:12 comes 10:5 30:6 42:5 67:5 67:14 118:20 119:1 129:17 130:9 152:7 157:15 coming 25:18 29:15 57:17 60:9 62:5 161:8 commencing 1:18 comment 126:16 comments 87:6 commission 1:1 6:17 9:17 11:1 12:23 13:2,13 17:25 18:17 19:19 21:3 23:9 30:8,12 30:15 33:4 37:8,18 38:6 39:1,8,25 47:9 47:12 49:16,22 53:16 54:5,7 54:18 55:8,20 55:25 56:6,16 57:24 65:20 66:10 76:19 80:12,13,18 87:18,19,20	89:19 90:14 91:20 92:4 93:11 94:8,16 104:6 106:17 112:21 115:25 117:16 124:3 134:20 147:24 151:11 152:24 153:10 162:8 172:24 commission's 11:5 14:20 38:19 87:11,13 87:14 118:14 commissioned 23:4 common 42:11 110:13,16 communicate 79:5 communicating 85:21 companies 24:19 72:23 80:6 85:10 company 72:18 83:18 compare 121:13,14 compared 14:3 14:14 43:2 compares 124:1 125:21 comparison 82:10	compel 98:5 100:15 103:6 110:12 compile 82:1 168:2 compiled 42:6 81:25 compiler 42:21 82:22 129:15 129:16 complainant 7:5 150:6 complained 165:13 complaining 19:24 152:16 complete 86:10 102:1 112:5 129:9 131:19 144:18 155:4 completed 36:15 145:20 completely 9:15 43:14 62:21 85:18 87:8 89:5 99:4 102:17 104:21 127:17 complex 90:2 90:12,16 101:21 component 11:21 79:21 83:9 124:15 143:17	componentry 78:23 79:5 components 1:8 10:25 11:20 12:13 66:17,18 70:6 75:7 86:20 comprise 79:10 79:11 128:24 comprises 75:9 132:11,12 comprising 88:6 computation 46:25,25 47:2 47:4,19 139:15 140:2,23,24 141:15 142:1 142:21 computations 12:10 46:5,12 46:17,18,20,22 46:23 47:5,23 48:13 110:1 138:10 140:16 142:13,19 compute 13:9 13:11,23 14:14 14:18,19 15:14 15:24 21:10,11 21:13,17 32:9 32:13,18,19 33:1,2,12 34:1 34:1,6,9,9,16 34:21 35:7,10
---	--	--	---

35:13,15 36:1 36:4,7,8,13,17 37:1,5,13,21,22 38:15 39:4,6 40:7,8,13,22,23 41:10,16 42:12 42:18 43:1,3 43:11,12,12 45:19,19,22,22 46:1,1,4,11,14 47:17,25 48:8 49:14,17 50:2 50:6,7 51:2,2 58:25 59:1,2,4 59:6,6,12,14,16 59:25 60:2,8 73:1,4,17 74:2 74:12,14 79:12 92:12 106:10 106:17,22 107:23 108:1,7 108:8,12,14,19 108:22,25 109:23 110:8 113:13,13,15 118:10 122:3,4 122:24,24 124:10,16 125:6,6 126:3 126:8,18,18 127:8,10,16,17 127:19,24 128:4,11,12 129:1,11,25 130:6,12,13	132:12 134:5 134:18,20,21 135:2 136:11 136:19 137:4,7 137:15,16,16 137:24,25 138:2,13,20 139:12,22 142:23 144:2,4 144:6,7,8,9,15 144:17 145:7 145:11,23,24 156:7,15,22,24 158:13,16,20 158:21,24 159:6,9,16 160:14,15,16 160:16,18,22 161:1,4,4 162:1,9,9,10,12 162:14,15,21 163:7,23 computer 20:1 22:24 39:14 80:4 88:6 89:5 97:9 168:15 172:10 computes 48:15 computing 72:10 75:16 76:10 77:1,20 77:22,25 88:8 126:3 148:9,11 148:16 160:25	conceal 103:1 105:24 concealed 19:25 20:4 concealing 39:20 conceded 108:23 109:2 concentration 72:12 concepts 12:14 conceptually 134:15 concerning 6:14 75:24 conclude 102:13 concluded 171:16 concludes 171:12 concurrent 15:13,23 25:3 60:15 71:7 73:18 163:23 concurrently 32:3,8 79:13 134:7 162:17 conducting 5:11 confidential 1:13 configuration 18:15 36:17 74:2 76:4	78:10,17 79:8 79:17,24,25 80:1 88:21 90:17,21 91:4 91:5 92:5,17 92:22 93:12,17 96:18 100:13 101:6,22,24 102:5,11,18 105:12 109:18 117:17,19,22 117:24 118:6 119:18,19 125:25 127:3 133:17,18 134:3,12 135:6 135:9,9,16,25 148:21 150:5 configurations 94:20 configure 66:14 118:2 150:18 configured 36:18,20 52:2 55:6,9,12,19 66:9 67:3 68:7 76:6 78:22,25 79:15,18,20 91:9,15,17 92:3,19 93:20 94:9,12 118:4 143:10,21 144:23,25 148:1 149:21
--	--	---	--

150:24 153:2 164:10 configures 11:23 14:10 16:6,23 31:8 56:20 144:24 153:1 confirm 62:16 76:20 114:22 115:17 126:4 132:23 169:20 confirmed 106:4 115:2 125:17,22 126:1 conform 125:15 confused 125:1 confusing 154:2 confusion 15:3 conjunction 133:20 connect 89:3 97:3 connection 66:13,19 78:4 80:2 92:23 connotes 82:17 consider 10:6 23:14 28:4,12 30:7 53:17 102:11,17 104:7 107:20 112:15 127:10	127:11 131:11 131:14 133:15 144:23 158:10 consideration 9:8 39:9 considered 28:10 39:17 50:6 56:3 82:5 93:24 101:23 102:10 104:10 104:11 112:3,4 112:12 115:8 131:6 133:20 141:2,14 142:10 considers 107:12 consistent 93:22 101:1 136:18 161:2 169:16 console 72:22 73:23 74:5 constant 141:5 141:11 158:22 constantly 94:5 constants 49:19 140:22 141:1,2 141:10,13 142:19 158:22 159:6,12 construct 129:11,12 constructed 125:5 126:7	construction 20:16 45:18 46:3 66:9,14 68:18 75:22 76:18 78:1 104:22 112:11 115:21 116:2 117:14 131:17 137:3,8 138:5 138:10 139:11 151:10 162:21 constructions 19:15 112:19 construe 40:12 construed 76:19 93:20 116:3 consume 59:20 113:14 116:25 117:4,10 143:6 156:2,10 consumer 27:3 95:2 consumers 24:5 24:7,13 25:10 contain 75:14 75:15,17,19,19 81:9,11,13,13 87:22 135:2 contained 75:11 108:13 134:21 containing 1:9 80:21,22 81:6 81:7,8 83:10	106:10 135:1 contains 86:10 101:3 106:2,3 contend 44:3 contending 88:2 contents 4:1 97:20 continue 93:3 131:21 continued 134:5 continues 132:21 contract 96:21 contractual 96:23 contradict 19:17 30:12 167:7 contradicted 54:4 56:17 contradicting 47:8,9 53:16 contradicts 39:7,8 47:11 47:12 55:19,20 57:23 67:20 contrary 53:13 control 98:14 controls 11:23 118:20 119:1 conversion 142:5
---	--	--	--

converter 128:15 129:25 130:1 cook 3:3 5:5,14 5:16,17,21 6:6 6:11,24 7:15 7:18,20,22 8:23 27:8 64:16,20 69:5 71:15,20,23 72:1,5 137:2 137:13 139:9 141:3 146:5,8 146:18,20,23 146:24 168:5 168:16 169:2 169:18 171:8 171:10,11 copied 13:17 46:8 coprocessing 36:17 copy 170:25 171:5 core 12:11 15:13 27:24 28:14,19,24 29:9,11,24,25 31:16,20,21,24 32:7 34:11 35:15,23 36:2 36:9 37:5,17 37:22 38:18 59:18 60:3,13 67:17 68:3	75:10 79:12,15 79:17,18,19,25 92:2 93:13 114:10 118:2 122:5 130:7 143:21 144:11 144:12,14,22 144:25,25 145:11,17,22 155:20 156:16 163:24,25 core's 61:23 143:9 corollary 35:14 59:13 corp 2:19 corporate 98:6 109:7 119:24 corporation 6:15 correct 76:23 117:21 125:13 125:22 143:20 149:2 150:14 167:9 169:21 169:23 172:11 corrected 91:6 correctly 45:13 50:1 corresponded 32:21 67:10 corresponding 90:4 152:25 corresponds 32:11 81:8	89:25 92:8 couch 69:1 counsel 5:5,18 6:3 7:24 8:1,3 8:24 71:21,23 75:23 93:19 112:6 114:24 146:12,20 167:5 172:12 count 48:14 146:13 counter 50:3 countries 72:11 country 84:13 county 172:4 172:23 couple 20:23 24:12 116:15 coupled 75:1 course 69:19,24 96:25 124:23 court 6:8,10 cover 9:19,21 12:15 23:22 163:11 covered 10:18 52:5,20 54:23 58:4 74:1 93:6 147:6 coyle 3:14 5:8,9 5:16 cpu 72:16,17 72:19 75:6 79:2 90:19 124:21 149:13	cpus 73:17 create 59:7 created 12:18 111:16 112:4 123:1 creates 104:5 credibility 154:24 credible 120:17 credited 74:11 93:8 crediting 93:12 critical 80:5 criticisms 126:20 cross 104:18 145:7 crossing 87:2 90:8 crr 1:21 crux 34:17 41:24 crystal 33:3 csr 1:21 172:21 cue 29:19 70:23 70:23,24 113:15 144:8,9 cued 58:23 cues 58:9 70:25 current 123:24 custom 85:18 95:4 customer 18:25 25:7 27:3 52:15,19 82:1
--	--	--	--

85:3,11,16,18 85:20 97:22 165:22 166:1 customers 24:9 24:14,15,21 51:21 84:4,11 84:15,21,24 86:25 87:2 96:3,4,11,25 97:6,14 98:1,7 98:14 133:14 167:9 customs 6:22 82:12 111:9 cut 117:1	deal 9:11 38:21 dealing 30:21 decide 40:2 41:3 76:22 170:8 decided 10:16 103:4 decides 166:16 deciding 28:13 decision 16:10 17:3 57:12 65:16 87:11,13 87:14 128:14 130:10,14 137:22 142:17 142:18 decisions 16:9 17:16 18:1 65:15 150:14 declaration 15:18,21 46:8 48:25 63:4 97:17 105:14 154:16 157:17 163:1 default 96:7 defer 169:2 170:8 define 46:10,14 141:23 defined 82:24 defines 157:21 definitely 57:4 60:17 171:5	definition 40:11 46:7,13 46:23 55:18 138:23 definitive 155:8 155:10 degree 104:16 delayed 145:25 delete 53:3 deliberate 93:15 deliberately 92:24 delineation 39:4 delivered 18:24 52:10,18 demand 59:24 156:13,17 demo 18:18,21 18:23 26:21 39:18 52:4,11 84:4 121:16 163:15,16,17 164:6,20,21 165:2,5,8,21 167:20 168:6 168:10 demonstrable 7:7 demonstrate 34:24 114:14 122:10 demonstrating 111:14	demonstration 105:13 demonstrative 21:4,16 35:1 106:20 108:5 144:4 145:2,5 demonstratives 30:17 denotes 139:18 deny 64:10 100:2 113:8 132:1 depend 64:23 dependent 14:17 30:21 depending 23:23 123:6 128:13 deposition 46:10,21 98:5 105:18 112:25 119:25 124:10 154:16 163:4 depositions 104:23 describe 35:2 114:8 described 75:4 83:20 128:17 129:14 131:12 describes 44:3 60:16 description 45:7
d			
d.c. 1:2 2:16 data 45:9 51:8 114:19 130:8 158:23 159:24 160:1 date 7:11 169:20,22 172:7 dated 100:20 davenport 3:8 6:1 david 3:13 6:4 dax 3:4 6:22 132:4 169:2 day 20:1,7,17 80:11 deadline 169:12			

descriptor 50:13,17 161:20	74:20 75:21 78:16 79:19 81:16 86:5 168:8	130:23 133:7 133:11 134:13 160:20	60:23 62:10 79:20 82:14 86:15,23 90:19 92:24 94:3 95:10 96:6 102:18 109:5 111:2,3 112:20 120:13 121:21 124:22 127:6 128:23 129:6 136:23 156:13 163:6
descriptors 43:4 49:17 50:3 161:18	determination 26:7 93:10 151:24	device 8:17 16:24 31:7 33:16 68:25 69:18 99:20 100:16,25 103:23 124:14 124:25 128:5 136:19 150:8 163:22 164:25 165:23	differentiate 49:14,19 50:1 159:16
design 15:8 55:10 72:24 73:23 78:16 81:23 82:24,24 83:2,5,7,16 85:17 163:19 164:12 166:13	determine 51:16 121:4 128:10 148:12	devices 2:10 5:22 7:4 24:11 53:21 96:9	differentiates 155:13
designated 40:22,23	determines 53:10 129:17	di 148:23 149:17	differentiation 47:4
designates 40:19	determining 157:22	diagram 61:22 86:14 92:15 135:4	difficulties 8:9
designed 16:20 38:3 42:13 66:11 78:14,15 97:13 150:1	develop 85:1,12 99:1	diagrams 86:16	digital 1:8 80:20 90:3
designs 26:10 72:21 83:17	developed 39:21 103:4	dictate 29:25	direct 75:13 89:16 162:25
desired 153:8	developer 127:1	dictating 157:14,15	directed 64:23 65:13 67:19 73:7 100:21 148:22 149:24 152:20 163:19 163:21 164:11 166:12,22
desires 10:12	developers 125:23	diebold 17:4 54:13 65:15 66:4 150:14 151:3,14	directly 19:17 20:24 53:13,15 54:4 81:14 87:23 88:16
detail 12:8 17:22 33:22 41:8 42:1 43:23 85:1 89:9 143:14,15	development 73:25 81:8,11 82:8,11 84:4,9 84:25 85:5,7 85:13,14,19 86:3 88:22 90:10 94:18,24 95:5 96:11,14 97:2 101:24 102:19 105:8 121:18,19,25 123:7,17,22 125:20,24	difference 28:3 47:6 109:25	
detailed 89:10 89:18 90:23 99:25 125:19		different 14:13 40:12 41:22 43:14 57:21 58:9 59:5,6,23	
details 11:16 39:11 73:11			

94:21 111:12 113:18 167:7 directorate 6:21 directx 159:22 disable 15:11 16:7 51:17 53:11 57:13 86:5 133:9 134:15 166:20 disabled 9:16 9:22 14:23 15:22 18:7 19:5 23:1,6 27:12 37:9 54:19 65:19 70:12 119:16 131:6 133:11 134:12,13 151:4,12,19 158:7 164:4 disablement 82:13 86:7 98:19,24 99:23 130:25 disables 15:8 54:14 88:19 123:3 133:18 148:21 disagreed 133:17 disagreements 137:5 disclose 47:17 47:25 105:20	159:8,9 discloses 73:12 113:20 117:8 disclosure 48:8 disclosures 39:13 disconnect 89:5 discovery 105:22 107:12 110:4 140:9,11 discuss 12:8 discussed 14:13 31:1 33:25 38:14 51:20 113:18 128:1 discusses 79:23 79:24 discussing 9:12 10:4,21 14:25 31:17 32:1 143:3 discussion 1:16 4:5,6,7 6:14,19 7:12,14,19,23 8:20 9:1 11:15 32:6 72:4 85:25 93:18 122:13 147:2 164:14 168:19 168:21 170:13 171:1,12,16 172:7 dismiss 130:20 131:9	dismissed 77:21 78:9 112:22 dispatch 19:2 73:2,8,13,16 80:17 89:11 dispute 9:18 11:4 15:19 16:4 30:10 37:11 44:15 45:24 52:6 110:3 166:20 disputed 20:25 88:3 150:4 162:18 disputes 110:11 distinct 81:19 139:18 distinction 46:22 47:6 distinctions 82:19 distinguished 128:3 dive 6:7 divide 24:14 divorced 127:17 140:2 document 43:24 48:23 97:17,18 107:17,17 112:13 149:9 documentation 115:14,15	documents 81:18 96:20 97:8,11,11 110:6,10,12,15 110:17 149:4,5 doing 41:9,20 41:22 42:15 43:18 44:21 47:2,8 109:14 117:12 129:20 140:18 142:8 160:18 161:4 162:3 164:11 domestic 78:5,6 78:8 dot 145:6,7 double 104:16 doubt 44:18 56:10 download 84:16 167:7,17 downstream 25:6 dr 21:5,15 32:14,19 34:3 44:2 45:21 46:7,9,15,21 62:13 63:3 64:2 91:2 97:8 97:17 99:10,12 99:14 104:11 105:10,14 106:4,20 107:21 112:25 113:24 114:22
---	--	---	---

114:25 115:2 115:10 117:6 119:6,7,8 120:17,20 125:17,21 126:7,13,15,20 126:22,24 131:18 135:22 136:18 138:19 138:20 143:15 153:19,24 154:6,8,8,14,17 154:19,22 157:12,16 162:22,23 draw 139:20 158:15,16 161:20 drink 158:12 driven 120:13 driver 11:22,22 12:18,24 13:15 14:7 16:6,6 17:13 18:4,13 18:19 19:2,4 22:14 24:7,8 25:10 26:19 29:15 31:7,7 31:10,13 33:16 36:5 37:1 40:21 42:5 52:10,22 53:4 54:14 56:19 58:7 61:22 62:4 65:14,16	65:21 67:9,10 67:12,15 68:25 69:3 70:11 78:17 81:14,24 84:12,14,16,21 86:4 87:3,4,11 87:17,19,22 88:18 89:3,22 92:14 95:21,25 96:3,5,6,6,17 100:1,25 101:5 101:12 104:13 106:2 111:17 111:25 117:22 118:1,1,7 119:17 120:9 121:6 122:23 123:2,13 125:10 127:2 130:22 131:12 131:15,19 134:5,15,24 135:6,7,10,15 135:25 136:5 142:22 144:20 144:24 145:10 146:2 148:7,24 149:10 150:4,5 150:25 152:15 152:24 153:12 154:4 157:21 163:22 164:8 164:13 165:23 166:2 167:6,11 167:12,12	driver's 14:10 52:11 133:9 152:9 drivers 81:9,12 150:8 164:25 drm 149:9 drop 11:12 dropped 11:10 due 8:11 duration 114:20 dvalin 90:16 e e 104:16 e.g. 129:7 earlier 30:5 51:20 54:12,24 56:18 65:1 70:15,16 82:8 86:10 90:12 91:11 92:6 116:16 121:12 144:10 159:19 160:19 early 76:14 168:11 easier 147:8 easily 25:20 112:22 167:22 167:22 easy 118:14 effect 127:2 effectively 12:5 54:19 65:19 99:21 108:24	effects 55:11 150:13 efficiency 41:17 42:19 efficient 44:5 162:2 eight 129:4 131:24 either 15:3 53:16 64:8 68:14,22 70:8 75:18 122:2,2 122:20 128:14 156:21 157:3 172:12 electro 150:23 element 30:1 31:4,11,14 54:17 55:5 56:2 57:12,14 67:8,12 89:10 92:2 152:25 153:1 elements 14:2 30:24 75:11 116:19 129:6 149:17 150:11 164:5 eliminate 154:20 eliminated 111:24 113:7 email 100:19 100:21 169:19
---	---	--	---

embodiments 113:20 emphasis 117:1 emphasize 9:13 employee 166:7 empty 89:8 enable 74:2 117:9 enabled 74:22 90:25 enablers 74:16 enables 73:7,13 89:11 enabling 59:20 73:11 79:24 116:24 117:3 143:5 156:1 encolor 160:8 encompass 138:10,13 139:1 encompassed 104:2 108:8 137:22 ended 33:4 73:21 80:3 endpoint 92:12 92:12 enforce 37:3 enforcement 6:20 7:1,9 engine 33:9,10 33:14 35:6,8 35:13,18 47:2 78:21 92:23	125:12 engine's 92:18 engineer 41:3 engineering 110:6,12 115:15 engineers 95:4 engines 12:8 14:4 30:25 31:21 55:5 67:3,16 68:7 68:10,20 69:2 74:14,14,15 75:10 76:5,5 76:24 77:6,9 77:12 89:20 91:4,16,22,24 92:1 93:16 143:23 148:1 enhanced 101:12 enhancement 19:22 20:8 38:9 40:3,19 45:11 48:2 100:7 101:4,14 102:8 103:18 104:1,4,9,10,13 106:3,13 123:12 131:16 133:19,25 135:24 136:5 161:8,22 164:18 166:15	ensure 35:25 52:21 ensures 37:4 ensuring 63:24 enter 103:23 entire 8:6 33:19 53:2 80:3 86:14 92:15 105:1 144:15 144:16 entirely 10:9 140:2 entirety 131:14 154:11 entities 52:18 entitled 94:22 entry 18:11 53:20 87:22 environment 78:12 89:15 95:18 111:7,13 eo 69:6,10,25 76:18 77:15 94:11 100:2,11 100:15 101:1,8 102:3,20 104:7 122:21 127:11 130:20 131:9 131:10 132:1 146:6 168:17 168:22,24 170:4,5,20,21 equal 63:11 140:20,21	equated 12:25 equipped 150:19 equivalent 82:22 erroneously 40:23 error 40:24 107:13 essence 9:15 essentially 10:8 11:6,23 12:20 13:18,22 19:10 26:17 29:21 44:12 46:24 48:2 67:1 103:6 107:18 112:12 115:11 event 94:10 172:13 eventually 21:3 72:22 73:23 145:22 everybody 5:6 5:14,18 evidence 52:13 52:15 69:21 77:24 78:7 79:22 90:22 98:15 120:18 121:24 123:6 123:25 126:19 126:22 136:8 149:4 165:22
---	--	--	---

evidentiary 80:11	144:2,4 158:19 160:18	31:19,20 33:9 33:10,14 34:13	126:22,23 138:21 152:18
exact 32:25 40:21 46:24 57:8,12 63:10 69:13 70:18 78:16 93:25 94:17 100:12 102:5 106:21 109:3,24 110:1 124:5 134:3 136:16 152:10 158:8	examples 138:15 139:2,4 140:6 excellent 143:13 except 158:6 excluded 87:15 87:22 excluding 136:5 exclusion 6:16 6:20 7:1,9 16:21 38:4 80:19 exclusive 37:16 59:17 145:19 155:20 excuse 158:11 executable 82:3 82:4 execute 79:13 79:20 92:17,20 109:16 128:23 143:23 executed 32:3,7 126:9 134:7 135:11 144:3,5 144:6 145:20 executing 33:11 35:11 79:2 143:24 execution 12:7 25:3 28:23	35:5,8,13,18 36:15 37:14 47:2 73:18 92:18,23 93:16 113:14 125:7 125:12 126:2 143:12 144:12 144:18 exercise 98:14 exercised 94:7 exhibit 4:12 exhibits 4:10 4:13 exist 22:4,8 36:2 37:17,22 37:25 38:17 117:20 163:17 existed 19:13 20:11 23:18 27:18 70:17 exists 22:2 expect 9:24 expert 15:18,20 21:5 26:22 32:10,14,22 35:1 44:2 46:7 48:24 49:6 97:7 99:11,13 104:11,15,17 108:3 111:23 112:25 117:5 118:24 120:16 124:7,7 126:11	153:19 154:7 154:10 162:22 163:1,2 168:12 expert's 154:10 experts 37:11 100:18 101:6 101:22 102:10 102:17 103:8 104:9,20 105:3 112:2 113:22 115:8 118:23 119:5 122:20 131:14 134:1 134:16 165:15 165:17 expires 172:24 explain 66:19 107:9 115:16 explained 119:9 123:18 explaining 38:12 explanation 27:7 136:22 explicitly 124:20 express 43:20 expressed 38:5 expressly 88:9 91:21 119:7 extensive 80:12 extent 58:3 104:21 108:16

115:20 external 18:13 79:1,2,6 88:18 89:14 111:6 extra 146:14 extremely 9:9	failure 114:15 131:13 135:18 135:21 fair 63:9,10 64:1,4 103:2 fairness 63:24 64:3 fall 18:20 51:19 falls 63:15,17 far 139:25 150:12 fashion 69:18 fast 111:19 faster 114:4,9 115:5 favor 166:24 favorable 107:16 favoring 118:15 feature 16:18 20:24 26:2 38:5 66:2 68:11 69:13,16 69:20,25 79:3 99:8 features 9:16 9:23 14:22 15:8 16:7 18:10 19:13 53:21 65:18 66:16 74:3,21 80:1 90:4 98:22 123:8 151:12,20	february 100:20 168:11 federal 88:24 149:23 150:22 153:3 feeding 93:15 feel 170:9 feels 104:25 ferris 2:4 ffs 118:19,25 119:3,7 120:1 157:11 fielded 78:11 fifo 29:21 58:9 58:10,12,15 figure 97:3 153:7 file 82:3,4 103:12 filed 150:17 163:17 files 97:20,22 filled 10:15 filling 157:18 finagle 163:6 final 80:11 93:10 finally 8:3 12:7 14:17 19:1 52:17,23 80:13 135:18 financial 2:5 find 34:12 57:6 finding 14:20 14:21 37:19	89:16 93:12 findings 54:7 54:13 finish 114:9 115:4 finishing 114:4 firm 5:23,25 6:3,5 first 7:23 9:7 11:21 12:17 15:1 18:12,20 20:22 21:18 22:11 28:16,19 30:24 35:4 54:11 58:10,10 62:2 65:7 68:3 72:18 74:25 75:1,2,4,8,8,11 75:14 76:8,18 79:10 85:24 86:9 100:6,8 111:22 112:24 113:10 116:17 121:3 128:20 132:11 137:12 139:8 141:3,4 143:23 147:6 148:17 169:11 fit 13:23 five 8:12 11:8 39:10,16 60:20 80:11 116:13 160:20 fixed 142:15
f			
face 107:17 fact 14:13 20:6 26:3 28:1 29:2 33:24 37:19 44:11 45:15 47:16,25 48:14 48:21 51:15 58:12 60:2,13 63:3,13 68:23 75:16 76:25 89:2 91:5,21 93:9 99:13 102:25 124:4 124:19 125:22 128:2 129:22 134:11 136:19 143:8 148:10 152:11 159:14 165:1,25 166:3 factor 120:13 factors 60:5 facts 10:16 failed 110:9 123:5 127:13 131:14,16 fails 56:2,4 88:22			

flagged 123:16	134:6 135:2,12	46:1,6 47:10	59:11,15 60:9
flash 53:3	167:21	47:12 51:17	60:13,24,25
167:18	formally	52:2 53:16,23	61:9,12,13
flashed 16:23	134:18	54:6,15,18,25	62:2 92:12
flesh 170:10	format 7:23	55:20,24 56:6	109:16 114:9
flexibility	52:12	56:22 57:5,14	126:9 134:8
45:12	former 41:4	57:24 64:11	135:12 142:2,4
flip 11:7 37:7	formerly	65:20 69:16,17	143:6 144:19
156:8	106:16	81:14 93:13	144:19 145:6,8
floor 142:22	forms 86:23	94:8,21 97:8	145:14,16,19
flowing 145:9	111:2,3,24	99:10 106:17	145:19 156:8
focus 11:15,20	113:9	111:11,12	156:22,23
64:10 65:3	forth 61:11	118:15 126:25	160:7
68:6 111:15	64:4 70:9	131:20 132:19	fragments
116:16 170:6	143:15 172:8	134:4,8,16,20	61:15
170:18	forum 10:10	139:2 147:25	frankly 24:23
focused 116:10	forward 19:8	148:5 151:4,13	150:6 151:10
116:14,15,24	168:20	151:20 164:5	154:23 156:21
focuses 20:9	foster 6:4	165:25 166:21	158:2
121:8	found 9:17,23	four 9:21 39:10	free 125:13
follow 169:14	11:1,17 12:23	39:16 52:18	frequently
followed 7:25	13:11 14:1,4,7	103:22,23	119:22
8:2 164:16	14:16,23 15:8	104:1,3 112:20	front 12:16
following 11:13	15:15 16:8,13	140:20 160:20	38:11 41:7
fond 16:2	16:19 17:1,7	fragment 13:6	69:11
footprint 72:11	17:25 18:7,16	13:21 14:14	fronts 151:9
force 62:16	19:6,19 21:22	15:14,24 28:15	full 93:11 95:6
forces 135:10	22:19 23:3,8	32:13,17 33:1	95:8 102:1
135:13	24:25 25:25	33:11 34:7,16	120:5 155:18
foregoing	26:8 29:2 30:8	34:20 35:4,5	156:18 172:10
172:8	30:12,25 31:5	35:17 36:6,7	fully 59:20 83:1
form 82:21	31:15,20 35:2	36:10,13,14	113:14 116:25
83:17 84:19	35:19 36:20	37:2,15,20	117:4,9 143:6
95:23,24 101:2	37:8 38:5 39:8	38:16 48:1	156:2
113:12 126:8	39:25 40:1	49:4 51:5,10	

function 17:6 29:6,20 70:16 70:21 81:23 118:19,25 119:4,8 120:1 129:15 142:15 145:10 150:1,2 150:19 153:8 157:12,18,22 functional 28:22,23 35:24 157:24 functionality 23:18 37:12 54:15 74:19 88:19 89:10 90:24 92:8 93:1 94:6 103:15 111:10 120:7 122:8 131:4,21 133:8 146:2 158:8 164:9,12 functionally 90:2 functioning 70:25 functions 71:8 83:1 93:2 fundamental 131:13 fundamentally 122:4 furnished 133:21 134:2	further 16:8 17:1,10 25:9 26:16 28:11 29:17 36:3 42:22 61:25 64:13 65:22 66:3 79:9,14 80:6 116:22 123:3 148:12 151:15 157:21 168:25 g gained 42:19 gaming 72:22 73:22 74:5 general 13:9 32:9,18 33:2 33:12 34:1,5,8 34:21 35:7,10 36:17 37:21 40:8,13,22,23 45:19,21 46:1 46:4,5,11,11,14 46:17,22 47:4 47:17,19,23 48:8,13,15 49:14 50:2,7 51:1 73:17 79:12 95:19 107:18,23,25 108:25 124:16 129:25 130:6 132:12 136:11 136:19 137:4,7 137:9,16,25	138:13,20 139:12,14,22 140:22,24 141:15 142:8 142:13,20 158:12,16,21 158:23,24 159:6,9,16 160:22,24 161:4 162:10 162:21 163:23 169:4 170:24 generally 73:19 120:21 138:20 141:2,14,25 170:8 generate 117:8 generic 75:3 geometry 13:7 13:22 32:12,16 45:8 127:21 128:24 129:7 getting 113:16 119:22,22 gigahertz 72:18 gist 98:13 give 8:10,12 9:2 41:5 43:23 62:2 63:9,11 65:5 98:11 102:5 143:19 144:5 146:14 147:1 155:10 given 12:20 25:17 61:11	84:15 99:19 101:7 102:3,18 102:20,21,23 102:24 103:15 105:7,12 112:5 113:5 114:13 155:9 161:5 168:25 170:21 gives 59:16 giving 59:25 61:9,15 63:9 63:12 global 72:11 glovky 2:4 go 6:7,11 7:22 17:24 25:1,4,9 25:12,14,23 26:3,16 30:22 36:8 41:7 55:24 58:11,14 58:23 60:5 64:20 66:7 70:5 71:16 75:21 78:15 80:16 83:19 89:19 91:2 102:14 106:25 112:22 113:19 120:10 121:7 123:12 128:8 130:16 135:4 139:9 140:6 143:14,20 144:3 146:25 147:8 150:6
--	--	---	---

155:2,16	gold 74:23	116:25 117:4	49:3,9,15,20
156:24,25	goma 3:11 6:3	117:10 121:7	51:3 72:10,17
159:2 167:15	6:3	123:3 124:16	72:23 73:18
167:17,21,24	good 5:8,20	125:25 128:22	79:11 89:2,13
goes 16:8 29:24	7:15 9:5,11	131:4 142:22	108:23 128:12
42:22 50:12,24	99:9 101:19	143:7 144:21	128:15,22
54:3,11 70:3	131:1 156:4	147:11 150:4,5	129:9,21,24
118:20 119:2	170:7	151:1 152:21	130:5 132:11
121:3,4 156:16	gotten 6:8	153:1 154:4	135:1,11
157:6,14,19	gpu 10:25	156:2,10,19,24	137:10,18
159:5	11:18,23,25	156:25 160:19	139:21,24
going 6:7 7:22	12:11,19,20	161:1,5 163:25	140:1,2,18
10:23 11:18	14:5,11 16:7	164:1	141:17,20
20:21 24:11	16:24 17:16	gpu's 125:12	142:1,10
25:20 28:18	22:2,5,14	gpus 32:20	149:20,20
29:7 30:16,22	28:22,22 31:8	63:21 74:4	158:17,21,23
34:25 43:15	31:9,18,22	80:21 81:6,13	159:17,21,21
48:19 51:1	35:24 36:2,8	149:13 152:6	159:22,23
52:7,16 53:20	41:10 52:9	grant 166:23	160:1,9,12
64:4 70:22	56:20 59:21	graphic 44:8	161:4 162:6,11
71:16 81:18	61:10 62:25	87:16 159:23	great 140:4
82:6 85:23	68:17 72:16,21	163:24	green 13:16
86:8,16 97:1	73:1,7,18,23,25	graphically	grounds 130:21
103:5 105:18	74:10,24 75:3	33:6	grouping 63:17
105:19 109:11	75:5 77:8	graphics 1:7	guarantee
109:14,15	78:15 79:4	13:5 32:8,15	24:10
118:22 121:9	81:14 83:10,16	33:1,11 34:1,5	guess 24:9
127:5 137:17	83:17 84:12,14	34:7 35:10,17	40:19 59:3,4
141:15 142:20	84:16 86:11,11	36:3,18 37:6	60:1 66:18
142:21 144:23	86:18 88:13,13	37:15,20 42:15	68:8 132:4
145:13,16	88:15,19 89:21	42:16 43:18,21	137:2,4
151:2 152:9	90:1,14,20	43:22 44:7,9	guessing
155:3 156:24	92:9,13,14,15	44:11,14,15	159:13
160:25 161:10	93:14 94:2,15	45:3,5,9,10,24	guidance 101:1
161:13 169:7	98:25 113:2,16	46:20 48:16,22	101:8 138:14

170:16 guy 107:5	77:12 78:10,23 80:1 82:17,20 89:25 90:23 91:7,22,24 92:9 93:3,14 94:21 109:11 109:12,18 111:5 133:8 134:11 142:15 147:11,12,15 147:15,21 149:1,24 150:1 150:12 153:5	145:11,24 149:25 153:11 help 102:20 helpful 145:3 170:5,13,15 helps 141:22 169:20 hereinbefore 172:8 herring 54:21 herrington 2:14 hey 97:3 103:1 hi 5:20 7:20 49:13 hiding 39:20 higgins 2:13 4:5,8 5:11 7:15 7:16 8:21 9:1,2 9:5 24:12 27:15 30:3 40:17 41:4 49:21,25 50:9 50:20 51:12 58:17 59:9 60:19 65:6 66:22 68:12 70:2 71:4,22 77:7 82:9 87:6 89:8,21 98:17 99:16 101:15 104:12,15,16 105:3 109:4,8 109:22 112:18 116:10,23	117:1 118:18 118:24 119:3,4 119:7,19 126:10,11,13 126:16,20 127:7,15 146:19 147:2,3 167:10 168:9 168:23 171:5,9 171:13 high 12:13 63:6 70:24,25 73:15 86:21 120:11 120:24,24 121:3 higher 63:4 highest 21:17 highlighted 11:21 14:6 29:6 128:17 highlighting 129:23 history 72:15 hit 38:10 147:5 hold 13:20 holding 13:18 holds 13:22 21:10,11 61:2 honest 61:7 154:6,7,14 hope 103:12 hoping 147:4 hour 7:25 8:1 53:9 72:2
h			
hack 25:14,16 26:18 27:4 hacking 52:13 half 32:18 53:8 hamburger 101:16,17 hand 83:24,25 108:5 handle 45:9 handles 161:18 handling 7:1 happen 35:21 59:23 81:9 93:21,21 117:16,18 118:3 124:9 happening 29:23 37:14,25 73:21 happens 59:22 93:23 130:9 happy 171:3 hard 50:17 89:6 hardware 16:14,18 17:15 63:21 64:24,25 65:13,17,24 66:2,17 67:13 67:19 68:11,14 69:1 73:10,12 74:17,18 77:11	hardware's 78:14,15 head 38:11 100:6 105:5 107:6 113:10 headquarters 72:13 hear 7:24 49:21 49:25 82:16 158:17 164:15 heard 76:16 100:6 161:14 hearing 11:10 11:13,13 21:7 30:18 32:23,24 80:11 100:9 heart 80:16 132:21,24 heat 95:14 held 61:3,4 114:16 116:3 144:10,20		

human 82:5,18 hypothetical 51:20,23 155:10 hypothetically 125:5 135:15 hypotheticals 155:6	ignore 134:11 152:6,9 ignored 62:10 117:2 ignores 62:4 immediately 116:11 impacts 66:20 implement 15:7 20:11 53:1 129:6 136:11 implementati... 83:4,5,7,21,21 83:22 92:18 implemented 89:12 90:3 91:8 95:22 116:21 164:12 166:13 implements 90:1,16 import 26:5 100:17 106:14 importance 124:13 important 9:9 12:6,9 42:3 76:15 82:6 83:14 155:16 importation 18:22 81:12 importations 81:1 imported 83:10 86:21	importing 25:1 impose 169:5,7 imposes 84:20 112:20 improper 105:23 improperly 97:9 inaccessible 98:23 131:5 incapable 59:3 59:3 include 10:24 17:8 49:17 81:5 84:12 95:20 103:7,14 104:4 116:5 137:18 138:12 159:3 included 108:7 133:24 134:21 136:4 includes 103:21 103:22,25 104:1,3 including 6:1 11:11 17:22 86:17 142:4 150:22 152:19 inconsistent 94:16 incorporate 19:2 24:17 84:6 87:15 95:17 100:24	164:24 incorporated 87:3 101:12 incorrect 132:17 incorrectly 87:10 increased 60:24 61:15 159:3 incumbent 122:9 124:23 independent 30:23 122:17 125:7 indirect 7:7 individual 74:8 107:5 industry 72:16 72:17 74:23,24 78:5,6,8 80:3 141:9 information 1:13 6:9 50:12 50:22,23 51:6 105:16,17,20 110:21,22 125:18 infringe 9:17 9:23 11:2,17 14:2,23 15:9 15:15 16:2,8 18:7,16 19:6 23:4 30:9 31:15 34:18
i			
ic 19:2 84:6,8 85:4 86:9,9,12 86:13 88:19,21 90:11 92:10 95:13 134:12 152:14 ic's 18:12 ics 18:19 81:5,9 81:13 84:7 90:8 94:19 96:12 icus 87:12 id's 14:20 93:12 137:22 identical 69:18 identically 69:23 identified 7:6 128:8 identifies 45:2 130:5 138:17 identify 50:15 51:6 99:14 identifying 51:7			

35:2,19 36:20 36:21 37:8 38:6,19 39:25 40:1 51:17 52:2,3 54:15 55:1 57:14 64:12 65:20 69:16 77:17 81:15 87:23 88:14 94:21 106:18 111:12 115:19 131:21 138:11 139:3 151:4,13,20 152:16,22 164:5 166:21 infringed 22:19 23:8 54:2 90:5 infringement 10:9 16:13 19:12 20:15 22:8 23:11 26:2 27:10 33:20 34:13 37:19 38:25 48:17 53:23 57:5 64:9 66:21 75:12,13 76:10 80:15 89:16 108:2,15 132:7,15 138:1 148:6 149:17 150:9 165:3,18 infringes 21:8 35:3 88:16	104:14 135:17 infringing 17:8 18:10,15 27:13 94:15 125:25 132:9 initial 26:7 39:13 40:4 41:9 42:4 105:25 initially 42:8 97:4 initiate 138:7 injury 80:6 innovation 72:15 innovator 72:10 input 70:8 82:21 83:2 130:8 134:4 141:23 167:15 inquiry 54:8 inside 31:18,21 47:1 88:7 89:7 90:11,14 92:9 93:14 128:4 136:4 installed 123:17,22 instruct 105:19 instructed 110:20 instruction 139:1	instructions 96:24 97:5,19 105:23 integrate 85:17 integrated 10:24 80:21 81:6 83:8,9 84:2,10,11 86:2 87:15,21 88:15 100:24 117:23 133:6 133:10 164:24 integration 96:12 intel 85:8 intend 9:19 intended 9:13 24:21 26:2 78:12 95:2,3 96:10 97:10 98:25 111:12 133:13 165:8 165:12 intention 55:10 inter 7:2 interact 79:1 interchangea... 50:7 interchangea... 141:9 interest 7:7 110:14 121:9 interested 168:24 172:13	interestingly 149:22 interests 110:16 interlay 92:25 internal 79:21 92:21 93:7,7 110:6,12 129:16 international 1:1 6:17 interpret 76:21 interpretation 76:20 104:22 interrupt 76:13 introduce 5:7 5:19 12:16 invalid 147:20 159:11 invalidity 47:13,22 48:16 invented 72:25 invention 46:19 73:20 74:22 147:12,13,16 invested 80:10 investigation 1:7 6:18 7:6 10:7,22 12:21 13:5 19:14 21:6 22:9 26:8 29:4 32:10 33:7 47:22 53:22 56:9 62:15,18 63:22
--	---	--	---

68:13 80:9 91:15 98:4 124:12 150:7 154:7 163:10 165:2 involve 111:10 139:13 involved 15:2 invt 149:21,23 ipr 19:18 48:5 55:15 56:7 57:20 150:17 150:17,21 irrelevant 10:1 19:21 62:21 124:18 151:25 irrespective 78:17 issue 11:5 19:3 26:7 38:7 51:18 68:12 69:22,25 72:8 77:4,4,16 78:4 83:14 85:24 87:19 88:1,2,3 99:22 100:3 102:2 103:19 104:20 108:10 121:10 124:8 131:9 132:8 135:19,21 137:13 158:12 158:13 162:18 issued 6:17 80:18 122:1	issues 9:12,25 10:5 30:11 75:22 85:23 91:11 105:5 111:10 131:8 it'll 89:6 128:16 140:15 itc 2:9 7:6 77:3 77:4,13 78:2,5 79:23 80:9 91:14 93:24,25 94:17 106:6,7 108:11,18 110:10 121:22 121:23 127:13 136:24 137:14 138:4,7,11 139:2 150:7 itc's 75:21 items 62:9 131:22	jm 90:16 joanne 1:21 172:6,21 job 1:25 12:2,2 12:5 13:16,20 13:21,24 14:5 14:5,18,19 15:12,21,24,24 21:9,10,10,13 21:13,13,17,18 21:19,23,24 22:2,7,7,7,18 22:25 23:1,3,4 23:5,7 27:12 27:14,14,18,20 27:22,23,25 28:1,2,13,15,16 28:16,17,20 29:1,4,8,24 30:25 31:1,3 31:18,22 32:13 32:19 34:8,10 34:10,15,16,17 34:20,20,21,22 35:16 36:1,3,4 36:4,5,6,7,7,8 36:10,11,13,13 36:13,14 37:1 37:2,5,6,13,15 37:20,21 38:1 38:15 40:5,6,8 40:8,20 41:10 41:20 42:12 43:4,5,9,9,12 44:22,24 45:5	45:14 50:10,12 50:13,17,22,23 50:24,24 51:7 58:21,22,24,25 59:2,3,4,11,11 59:12,12,13,14 59:14,15,17,25 59:25 60:2,7,8 60:9,11,13 61:1,1,3,4,8,8 61:23 62:2,2 62:24,25,25 63:2 68:16,16 70:12 77:7,8 83:15 87:7 89:20,21,22,24 90:2,6,10,16,24 106:17,23 108:19 109:13 109:23,24 111:19 113:13 113:15 114:3,9 114:11 115:25 118:13,15,17 118:20 119:1,2 119:10,10,11 119:12,14,15 119:17,20,21 119:21,21 120:1 137:24 142:24,24 143:5,6,16 144:16 145:23 154:7,10 156:15,23,25
	j		
	j 2:13 jailbreak 25:23 26:3 jammed 106:22 jasid 49:18 50:13 jess 6:23 139:9 jessica 3:5 jiao 47:15,18 47:25 48:6,9 159:4,4,7,11 161:3		

157:2,11,14,18 157:20,22,25 158:3,6,7 161:17,18,18 161:21,23 162:1,3,13,14 162:16 jobs 12:17,24 13:4,7,7,8,11 13:14,19,21,23 14:13,14,15,18 14:19 15:14,14 21:10,11,22 27:22 28:18 29:4,7,15,18 32:12,12,12,13 32:16,16,17,25 33:1,24,25 34:7,9 35:7,15 35:17 37:3 38:16,17 39:2 39:4,6,6 45:19 45:22 58:6,10 59:5 60:24,25 61:2,4,4,9,10 61:12 62:24 63:4,6 91:4 106:5,8,9,12,14 108:8,12 118:9 118:10,10 121:7 122:3 124:10,11 125:6 126:3,9 134:6,17,19,21 134:22,23,25	134:25 135:3 136:11,20 137:16 144:2,5 144:6,7,8,9,13 144:15,16,17 144:19,19 145:6,7,8,11,14 145:16,19,24 156:13 157:20 157:22 162:9 162:10,12,12 joined 5:10,24 6:2 joining 6:13 jordan 3:14 5:9 jump 26:6 27:15 30:3 51:13 60:21 62:7 85:22 jumping 72:6 junior 116:4 jurisdiction 10:10 19:11 juxtaposing 109:21	81:10 99:15 147:22 165:7 165:10 kicking 90:8 kind 7:23 21:1 24:10 27:3 30:13 41:17,24 42:22 49:19 61:15 141:8 151:8,24 152:13 154:21 155:5 kinds 85:6 kitchen 53:13 knew 21:24 161:8 know 17:16 24:5 26:16 31:2 43:15 50:25 53:8 60:6 65:23 72:9 75:22 80:13 85:15 89:5 93:22 96:5 97:16,21 97:25 98:3,12 98:13 102:21 103:2,13,20 106:1 107:9,9 109:7 111:1,9 114:23 116:11 120:5,23 122:14,16 123:15 124:4 127:1,7 129:12	132:17 133:16 137:23 139:11 139:12,17,23 139:23 140:1,5 141:16 142:19 143:4 146:1 153:25 158:15 166:17 167:23 known 106:5 kronos 125:15 141:22 142:7
			l
			label 29:18 label's 127:16 labeled 15:10 90:15 labeling 127:23 lack 104:25 105:1 lacks 154:24 language 42:5 76:23 94:9,12 94:17 117:17 139:17 142:3 143:9 149:21 languages 82:15 large 17:12 larri 107:5 109:1,6,6,21,22 113:25 114:23 115:3 119:24 124:6,9 125:1 125:2,3,4,9,15 125:22 135:22

136:15,17 155:2,6,9 larri's 110:1 114:6 late 168:11 launched 130:7 laundry 62:20 64:8 law 5:23,25 6:3 6:4 lay 15:5 132:22 layers 24:13 26:25 83:14 layout 11:18 leads 58:6 leave 98:16,17 127:4 130:19 leaves 18:8 60:19 leaving 155:18 155:25 led 20:14 73:5 74:3 left 11:13 12:22 27:18,23 28:22 53:9 55:8 58:9 68:5 74:7 83:24,25 86:13 108:5 legacy 23:18 28:2,3 31:1 40:21,24 43:16 61:3 63:13 66:17 69:16,23 70:17 81:5,8	81:11,12 82:10 106:5,9,11,16 106:17 120:7,9 121:13,14 122:3 124:2,2 125:21 131:22 134:18 136:9 142:23 158:5,9 161:24 legal 98:23 110:13,16 112:11 131:6 legitimate 15:19 lens 112:10 117:6 118:2 leo 18:6 19:5 23:12 24:25 26:5 52:1,20 81:2,4 152:23 163:17 letter 113:4 level 12:14 47:3 47:5 61:21 73:15 152:13 152:17,21 159:20,21 levels 70:19 levin 2:4 5:23 5:25 lie 76:10 life 77:25 light 25:21 lighting 160:5	likely 93:21 limit 60:1,15 156:17 169:8 limitation 56:8 56:8 67:25 69:2,17 76:20 91:15 132:10 132:10 148:12 limitations 53:24 57:8 66:15,16 68:10 77:2 limited 6:16 38:4 56:5,12 76:2,3 77:24 80:19 87:11 88:6 125:18 limiting 67:2,7 67:25 75:25 76:4 87:25 88:2 148:5,9 limits 59:24 79:10,14 117:2 156:15 168:25 169:5 line 78:20 114:7,7 120:3 120:3 125:4,16 128:19 129:4 129:14 130:4,4 139:21 140:5 158:15,16,18 159:5 lines 98:10 110:2,2 116:13	116:22 120:3 125:4 154:12 165:16 list 62:20 64:8 108:21 listed 111:17 131:23 139:4 lists 156:12 literally 22:1 90:1 113:3 115:13 165:24 litigation 107:4 107:6,8 little 10:12 20:21 39:12 43:23 66:19 81:17 112:9 145:2 147:8 160:6 llp 2:14 load 26:19 28:19 29:7 36:7 59:12,15 105:7 loaded 22:3,4 28:15,17 38:1 50:23 59:12,14 60:11,12 loading 27:21 29:1,4 158:3 lobue 3:10 6:2 located 21:23 location 50:14 51:2,5
--	--	---	--

lock 98:17,22 99:3,8,10,14,21 130:25 131:3 168:7 logic 90:3 130:1 logical 82:24 83:2 long 44:1 62:20 72:15 76:25 125:14 160:17 160:21 longer 16:21 22:2 23:10 30:5 34:20 35:3 38:4 42:18 43:12 80:25 113:2 119:19,20 120:11 154:3 look 29:7 33:21 44:19,21 47:24 48:23 55:14 59:18 65:15 68:3 86:16 92:3 102:8 116:9 141:21 154:1 166:11 168:20 looked 156:3 looking 69:14 70:22 100:19 126:6 130:3 132:5 135:5 137:14 147:11	looks 71:20 los 168:14 lose 151:6,9 lost 8:10 lot 9:25 11:11 11:11 14:8 19:20,24 20:9 38:8 48:18 54:24 58:2 70:4 111:1 164:14 low 70:23 84:10 120:11 121:2 lower 70:24	148:23 156:12 157:7,13 158:24 165:3 165:17 169:6 main 14:2 22:10 147:16 maintain 52:5 maintained 63:20 76:2 165:13 maintaining 64:3 maintains 18:22 major 73:22 74:5 make 8:16,17 17:23 20:14 24:14 27:17 35:25 54:2,4 60:2 61:12,16 71:10 84:17 85:6,11 96:15 96:16,21 103:1 109:24 115:24 127:13 138:1,2 145:14 147:8 151:23 163:8 166:1,2 170:9 makes 94:11 115:20 118:4 127:18 130:10 140:3 156:6 making 19:16 24:16 26:13	48:5 64:22 66:9 106:23 108:20 134:22 156:5 mali 10:25 14:5 14:7 31:6,13 56:18 63:21 67:10 69:3 77:8 78:15,17 79:19 81:9,14 84:12,13,16 86:4,10,11,18 87:3,4,11,17,18 87:22 89:22 90:1,14 92:9 92:14,15 93:14 95:21,25 96:3 96:5,6,6 99:25 101:5,11 104:13 106:1 111:17,20,25 112:5 113:2 117:21 118:1,1 121:5 122:18 122:23 123:2,3 127:2 130:22 131:4,12,15,19 133:9 134:5,15 134:24 135:6,7 135:25 136:5 146:2 148:7 152:6,24 154:3 157:21 167:6 manager 12:2 13:16 14:5
	m		
	machine 90:3 90:13,13,17,24 machines 17:4 made 16:6,12 17:24 18:4 29:3,5 30:17 32:23 39:21 42:24,24 43:20 48:20 53:25 54:14 65:23,24 67:20 68:20 70:4,10,11 71:11 75:23 88:17 90:6 97:9 100:6 106:13 108:1 108:17 109:2,5 111:1 120:12 130:14 142:17		

28:20 29:1,8 29:24 31:19,22 50:23 58:21,22 58:24,25 59:2 59:11 60:7 61:23 62:24 63:1 64:2 68:16 managers 59:4 manner 94:15 manufacturer 7:3 manufacturers 24:15 march 168:12 marie 1:21 172:6,21 mark 74:8 marked 4:13 marker 123:3 market 62:3 marks 62:5 masimo 25:23 massachusetts 2:6 mathematical 46:18,19,23 47:5 139:15 matter 1:5 9:8 10:2 20:6 54:19 63:13 65:14 102:25 124:6 150:10 152:4 159:15 159:24 160:1	162:17 165:1 169:4 170:22 170:24 matters 147:14 148:14 154:24 matthew 3:9 6:1 mean 25:15 35:9 46:24 60:4 65:22 89:7 122:6 132:14 136:12 148:17 156:11 156:19 164:19 meaning 76:5 138:23,24 139:14 means 23:10 34:24 36:4 40:13 55:13 59:18 62:25 78:22 148:20 148:25 150:18 153:12 155:14 measure 52:25 53:4 mechanisms 115:9 medium 29:18 60:9 62:6 70:21,23 71:1 120:11 meet 77:6 86:6 98:19 99:23 114:17 118:16	123:5 132:9,10 meets 21:14 153:12 memorialized 170:14 memory 18:13 68:3 74:25 75:1,20 95:16 97:21 mention 15:21 29:12 127:18 129:1,2 161:6 mentioned 15:25 28:12 30:4 31:7 35:22 51:22 52:23 54:12 73:6 86:10 89:21 109:4 127:7 128:21 143:5 149:20 153:17 157:11 158:22 162:4 168:6 mentioning 111:21 160:23 merchandising 7:3 mere 55:9,16 60:2,13 107:13 merely 105:3 134:25 merit 57:24 merits 133:15 134:14	met 30:25 55:16 91:17 136:24 148:6,6 148:7,25 149:11 michael 3:7 6:1 michigan 172:2 172:23 micro 2:10 5:22 7:4 92:21 93:7 microsoft 72:20 million 26:18 millions 154:12 154:12 160:4 160:10 mind 59:7 125:2 minimum 112:1 122:10 138:9 166:18 mintz 2:4 5:23 5:25 mintz.com 2:8 2:9 minute 7:25 8:2 8:12,13 10:20 64:17 71:16 146:9 minutes 7:25 8:2,3 72:2 146:13,14,25 147:4 misapprehen... 15:3
--	---	---	--

mischaracteri... 30:19	166:4,7	move 24:2 95:5 118:12 121:10	necessary 10:25 53:1,5
misquote 166:10	modifying 17:13 65:21	moved 17:19 103:5,6 110:11	78:23 79:5
missed 141:3	135:16	134:24	85:6 90:17
missing 47:15	module 14:6	moving 24:1	91:5 94:15
132:24	18:3 31:5,6,13	27:8 100:10	124:17 129:12
mistake 32:23	54:12,17 55:3	multinational 85:10	152:25
107:13 108:12	55:6 56:1,5,11	multiple 26:25	need 6:9 10:5
mistaken 169:14	56:19,20 57:1	41:11,13 86:23	16:24 27:1
mitek 150:22	57:8,20 65:10	92:19,20	31:25 32:2
mode 16:25	66:24,25 67:4	132:15 151:9	47:23 53:9,17
model 147:22	67:6,11,15,24	multiply 47:3	62:1 63:1 68:4
modification 16:12 18:4	68:24 75:15,18	140:20	75:11,14,15,17
38:25 54:14	75:25 76:3,6,8	multiplying 141:1	75:18 77:19
70:10,11 95:19	76:22,23 77:18	multiport 72:19	78:22,25,25
133:13 138:8	77:19 78:7,13	murphy 3:12	85:12 86:4
148:21	79:1,3,7 87:24	6:4,4	88:13 99:24
modifications 64:25 85:7	88:12 89:1	muted 8:18	110:14 116:4
modified 14:9	90:18 91:9,16	133:2	123:16 140:22
17:5,9 31:14	91:17,19 92:7	mutually 37:16	147:4 148:12
52:22 57:13	147:10,18	mux 128:9	148:16 151:5
65:17 87:4	148:3,13 149:6	n	151:10,16
95:3 96:10	151:6,11	name 5:21	153:18 160:11
97:13 106:14	module's 88:7	42:23 43:6,8	161:19
147:24 163:22	89:14	43:12 74:8	needed 52:25
modifies 106:5	modules 86:17	161:11,15	53:6 151:23
133:17 135:6,8	157:12	nearly 55:13	152:16 160:7
modify 17:15	moment 6:9	necessarily 108:8 122:8	165:18
26:9 53:3	103:11 143:19	138:12 144:8	needless 10:14
91:25 97:19	144:5 145:14	156:11	needs 17:23,24
98:10 99:1	145:25		30:7 75:19
	monday 169:22		127:16 133:19
	months 102:17		166:16
	morning 5:8		negate 16:13

neglected 103:7 103:8 neither 149:21 156:22 never 22:18 70:24 108:11 108:17 112:3,4 114:19 115:8 153:8 162:6 163:14 new 10:4,9,12 15:7 19:10,12 19:14 20:14,15 20:15 23:14,16 23:24 24:6 27:25 30:11 38:23 40:10 45:17,18 46:3 46:6,23 47:7 54:8,24 55:2 55:17 56:4 57:16,22 60:20 64:9 74:10 78:17 82:10 86:3 87:4 89:22 99:25 101:5,11 103:13 104:13 106:1,14,15 111:17,25 112:3,5 113:9 115:9 117:21 117:25 118:1,1 119:13,18 121:5,18	122:17,23 123:1,2 127:2 130:22 131:12 131:15,19 133:9 134:5,15 134:23 135:6,7 135:24 136:5 146:2 152:18 157:7 158:9 nickel 3:13 6:4 nine 98:10 non 59:1,6 66:20 113:15 122:24 125:6 126:3 158:23 nonexistent 27:13 noninfringem... 114:14 normally 107:20 notable 116:19 notary 172:1 172:22 note 8:15 20:2 22:12,17,17,21 22:25 26:14 36:24 62:8 107:2,3,9 noted 26:15 124:20 137:15 166:1 notes 172:11 noting 26:6	notion 116:5 124:17 novel 72:21 74:12 novelty 147:17 nuance 99:25 number 6:18 21:17 69:12 77:11 84:6 87:6 94:14 98:21 99:2,3,6 99:7,22 101:2 101:3,9,10,25 107:14 111:20 112:19 115:21 116:3,8 117:12 117:13 118:13 120:5,19 121:1 121:2,2,3,16,19 131:2,23 132:20 144:7 numbers 120:14 133:1 160:11 numerical 120:23,24 numerous 10:4 nw 2:15	obviously 169:5 occupy 38:8 occur 12:10 44:20 94:10 occurring 35:20 occurs 35:12 oems 24:16 offered 104:21 138:21 162:23 office 6:21 officially 160:19 offload 73:17 oh 65:6 109:4 129:23 137:23 139:10 153:5 163:7 okay 50:9 59:22 63:1 70:22 79:9 121:9 141:3,7 143:20 146:3 164:20 168:18 170:1 old 26:19 33:16 53:3 61:18 once 13:14 20:9 36:9 39:25 51:15 52:7 53:10 60:12 132:23 ones 25:19
		o	
		object 83:24 obscured 95:13 obtain 83:16 102:20 103:6 170:24	

online 167:13 167:23 open 10:8 130:17 133:12 133:13 169:7 opengl 160:18 160:24 opengl 42:4 43:14 159:22 opening 10:3 23:22 161:12 operate 18:14 93:3 107:19 122:12 126:12 131:12 149:13 159:23 operated 51:9 71:9 93:4 operates 69:18 70:17 operating 41:11 43:15 49:1 78:12 81:24 89:1,15 95:17,20,25 96:16,17 111:7 111:13 126:14 126:15,17 151:22,23 152:8,12,17 159:25 operation 11:24 42:15 43:17,21 44:8 44:9,9,10,12,15	97:25 122:22 142:2 162:4 operations 142:4 160:9,10 opinion 39:1 49:16,22 102:22 104:15 104:15,19,21 115:17 118:23 118:24 120:20 120:21 126:22 127:15 138:21 154:15 162:25 opinions 112:8 112:9 154:8,25 opportunity 65:5 103:15 opposed 138:25 169:6 opposite 35:12 36:12,19 63:11 124:5 opposition 17:13 optics 49:2 option 63:9 84:15 98:17 171:2 oral 1:16 4:5,6 4:7 6:13 9:1 72:4 118:18 147:2 168:19 168:20 170:12 170:25 171:16 172:7	orange 12:7 39:3 145:6 order 6:17,20 7:1,3,9 16:21 17:23 29:21 38:4 58:7,11 58:11 75:12 76:10 80:19 85:12 88:14 116:2,8 149:12 158:1 ordered 133:22 133:23 164:22 orders 80:25 ordinary 88:11 138:22,24 139:14 orient 10:20 original 27:18 42:6 43:16,19 44:4,16 105:21 originally 12:3 96:19 orrick 2:14 7:16 9:6 orrick.com 2:18 os 129:9 output 41:13 41:22 42:14,16 82:23 83:3 128:13 130:8 141:24 outputting 49:2	outside 18:6 19:4 23:12 24:25 26:5 31:23 52:14 77:1 88:8,13 89:4 111:11 114:24 124:7 163:16 overall 124:21 overcome 57:7 57:15 67:7 68:1 overlapping 34:12 overview 12:14 15:1 39:12 72:7 own 30:20 62:15 77:7 83:16 96:12 97:4 112:24 113:22,25 119:19,24 142:3 162:22 163:2 165:23
			p
			p.m. 1:18 5:3 71:18,19 100:21 146:16 146:17 171:17 package 86:24 packages 86:22 packaging 86:24

page 4:3,12 49:16,22,24 88:4 91:13 93:11 110:2 113:3 114:6 124:20 125:3,4 125:16 137:14 149:3,10 150:20 168:25 169:5 pages 19:9 169:6 paired 65:25 pairing 16:15 16:25 papers 15:17 paragraph 157:17 parallel 37:2 38:16 parent 161:23 162:3 parenthetical 137:17 part 6:16 9:24 10:3 13:17 15:4 19:23 32:17 33:9 39:9,13,14 40:4 43:19,21 44:4,17 49:9 50:22 66:5,8 67:17 68:6 75:8 76:24,25 77:20 79:4	86:11,12 88:13 88:18 89:14 92:13,14 100:7 101:5,13 105:11,12,13 106:16 111:6 115:22 118:21 124:2 127:9,9 127:16 134:18 134:24 135:24 137:9 140:11 141:3,4 147:22 147:25 150:5 151:7,17 156:4 157:19 167:19 partes 7:2 partial 155:5,5 participating 6:19 participation 7:11 particle 49:6,7 160:2 particles 126:16 particular 30:16 68:5 69:17,20,25 143:17 170:5 170:17 particularly 113:5 parties 6:25 7:6 7:7,10 8:5,5,19 52:10 77:5,22	77:23 137:6 168:19,20 169:5,8 170:6 170:8,17 171:15 partitioning 116:17 parts 42:9 166:18 party 18:24 22:15 84:7 172:13 pass 161:19 passage 116:12 117:5,7 passive 48:20 past 31:18 143:22 151:3 patch 14:9 15:23 18:19 19:4,22 20:8 20:16 23:5 25:3,24 31:2 35:6,16 37:12 37:24 38:3,9 38:13,13,21 39:12,13,23 40:3,4,19 45:11 48:2 51:22 60:25 62:3,5 63:5 100:7 101:4,13 101:14 102:9 103:18 104:2,5 104:9,10,13	105:24 106:3 106:13 122:2 123:11,12 127:7 131:16 133:16,19,25 135:24 136:6 142:24 152:5,9 161:8,23 164:13,18 165:14,15 166:15 patched 23:11 24:24 25:10 60:6 101:4 118:8,9,9 127:2 145:7,10 145:23 patches 9:21 11:17 12:3 15:7,10,11,19 16:5 18:9 20:13,18 22:13 22:15,24 36:25 38:14 39:10,16 39:24 51:16 53:11 64:11 102:4,6 103:21 103:22,25 104:3 105:15 106:2 122:23 125:7 151:18 151:25 154:20 164:4,21 166:13,20
--	---	---	---

patent 11:12,14 19:18 44:1,3 44:13,19,23 45:6 46:19 48:6 58:12 59:19 60:3,16 73:3,5,6,9,12 73:20,25 74:1 75:4 80:16 104:24,25 105:2 107:24 112:15 113:18 113:20 116:22 117:12 127:6 127:17,18,25 128:2,2,18 138:16,17 139:4,16 140:25 142:17 147:12,16 150:18 155:12 155:19 156:6,9 156:12,20 159:18 160:17 160:21 163:9 patented 72:8 74:22 80:7 patents 11:8,11 129:3,4 path 43:2 pause 24:1 49:10 pc 2:4 85:8 pdf 153:22	peek 29:5 70:16 peel 140:15 pending 19:18 48:6 93:25 94:17 pennsylvania 2:15 people 25:16 percentage 116:18 perform 47:19 48:21 66:11,11 69:23 78:23 109:25 122:20 128:22 140:22 150:1,2,19 153:8,9 performance 19:22 20:8 38:9 40:2,19 45:10 48:2 80:5 100:7 101:4,12,14 102:8 103:18 104:1,4,8,10,13 106:3,13 123:11 131:15 133:19,24 135:24 136:5 161:8,22 164:18 166:15 performed 42:10 110:19 110:21 114:22 131:20 140:16	performing 17:6 41:12 43:16 55:14 142:18 performs 40:6 46:5,11 73:11 93:2 129:14,16 139:14,22 141:25 perils 104:18 permissible 17:2 permission 103:23 permit 38:15 38:17 permits 23:22 permitted 18:11 52:16 persist 90:10 person 88:11 130:14 petition 85:22 88:22 103:3,7 105:21,25 130:21 pfister 112:25 120:20 153:19 153:24 154:6 154:14,19,22 phone 16:15 25:24 26:4 65:25,25 phones 16:13	phrase 73:3 76:2 physic 48:21 138:18,18 139:5,23 physical 73:10 82:24 83:3,4,6 83:7,20,21,22 92:18 100:16 102:19 103:14 106:2 133:21 135:7 136:1 physically 17:15 physics 48:25 139:15 pick 21:12 157:1 pickles 101:17 101:18,18 102:15 164:15 164:17 picture 86:13 95:11 piece 89:25 95:19 160:1 piecemeal 102:4 pieces 112:6 pilot 40:5,20,21 41:9 44:3,4,12 44:21 45:16,22 106:15,22,24 107:10 108:9 108:13,24
---	---	---	---

109:3,10 110:4 110:7 134:6,17 134:22,23 135:2,10 137:20 138:12 140:7 142:12 158:25 161:7 163:7 ping 61:11 64:3 70:8 119:4,5,8 119:9,10 pipeline 43:22 44:25 49:4,9 92:21 93:7 137:10 139:24 144:10 160:12 pipelines 92:19 pixel 45:8 127:21 128:25 129:8 142:2 place 129:19 130:11 placebo 138:22 places 65:7 plain 138:22,24 139:13 plainly 36:25 plan 146:9 plans 89:18 play 67:5,14 123:24 playstation 74:9 please 5:7,19 8:17,24 64:20	72:3 102:5 137:10 139:9 146:9 plurality 14:3 30:24 55:5 67:3,16 68:7 68:10 69:2 74:15 75:9 76:4,5,7,24 77:9,12 78:13 78:21,24 79:6 79:15 89:20 90:25 91:16 128:23 129:6 143:10,22,22 143:24,25 plus 17:17 146:1 pluses 145:10 point 12:9 51:1 51:4 60:1 63:25 65:8,14 68:8 69:22 76:15 81:10 88:9 99:19 102:12 111:22 132:25 141:19 144:19 146:24 147:6,13,17 148:23 150:7 151:2 153:4 155:17 156:5,5 156:12 157:13 165:20 170:11	pointed 56:18 137:15 149:19 152:24 153:23 154:17 158:19 162:8 pointer 50:21 pointers 49:17 50:4 161:18 pointing 32:15 47:24 71:9 points 98:18 147:6 162:24 pong 64:3 119:4,5,8,9,10 pongs 61:11 70:8 popeo 2:4 portion 17:12 51:13 117:2 148:24 167:11 pose 170:4 posing 168:25 posita 117:6,8 posited 155:6 position 21:9 41:1,13 43:25 47:21 49:1,5,7 55:15 56:14 57:25 65:9 75:24 76:11,16 88:14,25 89:17 112:2 120:6 132:23 135:14 137:12 159:10 160:4,6 165:13	positioned 160:3 positions 120:6 170:10 possession 18:22 52:5 possibility 36:1 36:16 possible 25:15 25:16 73:19 170:24 post 32:24 45:2 168:20 170:25 potential 95:19 potentially 25:13 34:11 95:17,20 130:24 powers 72:21 practiced 122:16,17,18 preamble 75:17 77:23 88:5 precedent 28:7 53:19 55:12 69:10 70:1 88:25 112:11 131:7 153:3 precise 100:13 precisely 79:7 precluded 33:17 preclusion 111:10
--	--	---	--

predesigned 83:17	presumption 105:6,7	prio 29:5 70:17	prioritized 119:23 156:23
predominantly 120:18	pretend 101:15	prior 11:10	prioritizes
prefaced 70:15	pretty 20:23	56:9,23,23	27:14
preferred	162:24	57:8,15,20	prioritizing
119:11	prevent 14:24	67:8,21,22	28:1 125:11
preferring	15:13 16:7,17	68:1 73:19	157:4
120:1	17:5 26:20	82:12 84:23	priority 14:18
prefers 119:10	38:17 53:2	98:3 111:9	14:21 21:14,18
prelude 65:17	57:13 80:3,6	147:20 148:19	21:24 22:6
preprocessing	125:10	159:7 170:25	23:4 25:4
44:20,22	prevented	priorities 29:14	28:14,25 29:7
present 5:6,18	17:10 35:20	143:4	29:13,16 32:1
7:25 8:2 53:21	37:25 38:2	prioritization	38:17 58:5,5,8
72:2 76:8 78:8	71:7 113:16	94:12 110:25	58:12,13,15
101:24 121:21	prevents 25:3	111:2,3,16,18	59:7,18,24,25
131:22 133:10	58:19,21	111:18,19,24	60:4,8,8,10,10
146:21 148:15	163:23,25	112:3,9,12,14	60:14,17 61:8
presentation	preview 76:1	112:16 113:6	61:9,11 62:2,6
5:12 65:4	previous 39:23	113:10,11,12	62:9,10,17,17
66:23 76:14	40:16	113:17,21	63:4,6,11,12,20
132:6 151:17	previously	114:3 115:4,5	63:23 64:6
153:20,23	167:5 170:12	115:7,9,11,16	70:6,13,19,20
155:17 157:10	primarily	115:19,25	71:2,8 79:16
159:19 161:13	112:18	116:20 117:12	80:2 113:2
presented 7:8	primary 132:6	117:18,25	115:13,22
89:19 90:14,22	132:17,18	118:12,13	116:5,5 117:9
131:11	principal 133:5	119:14 120:4	120:11,24,25
presenting	133:14 135:19	120:18,22	122:6 143:7,11
137:3	principally	122:25 123:1	143:11 145:1
preset 59:24	120:10,13	125:10 126:2	145:12,18,21
117:1 156:14	printed 80:21	126:24,25	145:24 152:7
presume	81:6 84:5	127:3 157:1,3	154:1,3,13,18
103:11	95:12	prioritize 59:1	154:20 155:14
		59:2	156:11,14,17

156:20 158:2 164:1 privileged 105:23 probably 70:3 70:3 96:8 110:14 141:13 159:14 163:2 problem 118:21 159:1,5 161:14 171:2 problems 40:14 103:17 104:6 proceed 8:24 72:3 proceeding 10:1 19:9 55:23 57:21 78:2,5 84:23 94:17 98:3,4 102:12,12 103:10,13 105:8 106:8,9 110:2 120:1 121:20 122:10 123:15,20,23 125:3,16 127:14 133:22 133:24 136:16 136:17 138:8 169:8 process 16:15 45:3 47:16 64:1 75:8,16 83:19 108:23	140:3,12 141:16 142:4 160:7 processed 32:20 34:6 processes 98:5 159:12 processing 13:5 15:13,23 32:8 32:15 33:11 34:1,5,7 35:6 35:10,17 36:3 36:18 37:6,16 37:20 40:7 43:19 45:5,10 47:18 48:23 49:15,20 51:3 56:12 57:2,9 67:1,4 68:4,24 71:8 72:11,16 72:17,23 73:10 73:13,18 74:25 75:1,2,4,6,9,12 75:14,19,20 76:8,9 79:11 87:16 88:12 90:19 116:4,19 128:13,15,22 129:5,9,22,24 130:6 132:11 135:1,12 137:19 139:21 139:24 140:1,3 140:18 141:17 141:20 142:1	142:10 147:19 148:4,13,17 149:7,14 158:17,21 160:16,17 162:6 163:23 163:24 processor 85:8 89:4 149:24 produce 42:21 100:16 110:10 110:12,16 114:15 115:14 123:19,21 124:18 125:18 produced 22:11 39:13 42:24 99:15,20 114:19 122:15 122:19 133:22 133:23 136:22 163:13 164:20 165:12,14 producing 41:21 165:10 product 16:22 18:14 24:20 26:5,22 40:24 52:12 80:23 82:10 85:2 95:2,10 97:2 100:13 103:5 106:6 111:11 120:8,9 132:8	products 9:16 10:23 11:19 12:4,5,24 14:3 14:24 15:9,12 16:3 18:6,10 18:11,19 19:1 19:13 20:12,25 21:12 22:2,13 22:20 23:6,10 23:11,18 24:17 26:10 27:25 28:2,3 30:9 31:1 33:17 34:19,23 35:16 35:21 36:19 37:9 38:3 39:17 51:24 52:9,14,18 54:2,16,20 57:13 61:3,18 63:14 65:17 66:17 69:24 70:17 78:5,6,8 79:8 81:4 84:25 86:1 94:23 97:13,15 100:3 121:14 121:15,15 122:1 124:3 125:21 131:22 136:9 158:5,9 163:22 166:22 167:16 168:2 professional 85:8 107:7
---	---	--	--

professionals 99:1 profits 161:25 program 40:20 40:21 41:2,20 42:4,7,12 43:7 43:14 44:5,16 45:4 49:18 50:3,13 51:1,4 51:8 52:24 55:13 61:24 62:15 106:16 106:22,24 107:10 108:24 109:3,4,5,9,12 109:19,24,25 127:8,19,24 128:11,24 129:4,17,18 130:7,12 140:7 140:18 142:6 programmable 26:17 27:2 99:15 165:7 programmed 66:11 118:3 150:18,25 programmer 62:1 152:7 programmers 125:13 programming 82:15 153:7 programs 42:21 45:2,7	105:17 108:9 108:20 109:11 109:15,19 110:5,7 125:11 125:14 127:20 127:22 128:7,8 129:7 130:5 135:10,13 137:21 138:13 139:3 140:8,15 142:13 159:20 159:21,23 167:25 promise 77:14 promised 87:25 proof 131:13 135:18,21 136:25 proper 54:6 110:9 properly 103:12 proposed 103:17 proprietary 85:4,19,20 prosecution 57:7 147:18,23 148:19 protection 6:22 prove 94:1 proved 24:9 provide 26:16 42:1 72:7 76:1 78:16 96:2,23	96:24 97:2,23 97:24 98:2 100:1,22 102:22 103:14 105:25 110:24 118:23 131:16 164:23 provided 19:25 20:6,10,16 22:12,14,16,22 22:22,23 25:10 26:1,9,22 39:18 45:1 74:10 76:21 84:18 95:18 97:1,5 99:11 101:1 107:2 115:12 120:17 120:20 131:18 134:16 138:15 163:1 166:8,9 168:7,13,14 provides 24:18 24:19,24 45:6 90:17 119:17 providing 42:15 91:5 92:5 102:24 104:18 130:16 143:11 167:8 provision 52:21 public 167:10 172:22 pull 143:18	pulls 29:20 pure 51:22 166:3 purpose 15:3 85:13 107:23 108:1 111:13 136:11 141:15 142:20 pursuant 101:7 push 48:19 116:7 put 15:6 21:7 26:9,15,21 39:17 44:7 48:24 77:14 84:7 87:25 90:9 95:24 96:7,14 102:6 103:3,5 119:6 123:2,13 125:23 135:7 141:16 153:18 155:23 158:1 164:25 166:5 168:2 puts 26:10,11 89:8 96:8 putting 27:23 39:17 120:23 157:20,24 167:17
q			
qualify 137:24 138:6			

question 7:4 24:4,6 27:9 31:18 40:15 41:6 42:23 49:14 50:1 54:23 64:19 66:18 69:9,9 82:8 132:4,5 135:20 139:20 140:4 141:18 142:25 143:3 143:13 152:14 154:1,21 158:14 162:25 167:4 168:5 169:3,9,15,22 170:3,7,23 questioning 91:6 questions 7:8 8:19 24:2 42:2 49:11 51:12 64:13 69:5 70:16 71:14 130:18 146:4,5 146:11,12 157:9 161:17 167:1,2 168:17 168:22 169:11 171:8 quick 49:13 130:17 167:4 169:11 quickly 11:7 15:4 23:22	30:22 51:13 53:8 58:1 60:21 quite 12:15 quote 35:12 63:3 77:14 91:13 137:17 149:8 153:18 153:24 154:21 155:18,19,24 155:25 159:3 166:6 quoted 124:20 150:20 156:4 quotes 87:25 88:1 155:4,5 quoting 150:21 r raise 19:14 23:19 61:6,18 62:22 78:1 170:10 raised 10:6,7 19:15 23:14,15 23:17 25:13 28:6,8 52:6 53:14 54:21 60:20,23 61:5 61:18,20 117:14 138:4 170:12 raises 51:14 raising 30:11 53:12 54:8,25 70:6	ran 42:13 107:14 138:19 rather 38:10 64:24 135:22 140:17 143:11 reaching 150:12 read 14:4,7 65:13 66:16 67:8,14 68:10 68:25 69:3 104:24,24 137:17 140:8 147:10 149:1 readable 82:5 82:19 reading 68:15 105:2 116:12 real 103:16 104:5 really 9:18 11:4 15:17,18 45:16 89:6 99:4,24 100:9 101:20 103:1 106:25 113:22 115:6 118:19 120:16 124:8 126:12 126:17 127:9 132:20 140:5 142:20 147:3 147:13 170:22 realtek 2:19 5:6 5:7,10,12,13,15 6:15 7:2,13,17	7:24 8:3,21,24 9:6,10 18:12 18:19,21,23 19:2 22:15,16 24:5,10,18,19 24:23 26:9,10 26:16 41:1 51:24 52:4,9 52:17,20 65:3 68:8 69:22 71:21 75:23 77:5,16,17 78:1,6 80:7,14 80:19 81:2,3 82:1,11,11 83:4,15 84:6 84:10,11,15,20 84:22 85:4,17 85:19 86:1,2,2 86:19 87:10,20 88:5,15,16,19 88:21,22 90:6 90:8,9,11 91:23 92:10,10 93:4 94:1,3,18 94:22 95:7,13 95:18 96:2,8 96:10,11,12,14 96:18,19,20 97:10,23 98:2 98:6,6 99:17 99:18 100:16 100:21 102:15 102:19,25 103:12 105:16
---	--	--	--

105:24 108:10 108:11,17 110:5,9,13,19 110:23 111:23 112:17,20 113:3,21 114:1 114:12,23 115:20 117:11 117:21,23 118:11 119:15 119:16 121:6 121:13,18,19 122:7,9,11,15 123:4,7,10,19 123:21 124:5,5 124:24 125:18 127:13 130:23 131:10,11 133:6,7,10,11 133:20 134:12 134:13 136:13 136:21 137:6 137:22 146:18 146:25 164:8,8 165:24 166:7,9 167:8,14 168:23 171:9 realtek's 15:1 24:7,15 41:1 51:21 52:15 81:3 84:24 85:2,11 91:12 93:19 94:10 95:6 101:10 103:17 104:9	104:20 111:23 112:2,6,24 113:1,6,8 114:13,21,24 115:8 118:2,22 120:5 122:22 124:17 127:11 129:15 130:21 131:14 132:1,7 146:15 reason 14:20 20:22 25:5 55:21,23 56:24 67:24 99:9 156:4 160:23 162:2,11 reasonable 102:13,13 reasonably 131:5 reasoning 143:15 reasons 56:3 77:10 131:1,25 rebut 154:8 rebuttal 4:7 8:4 23:23 38:10 58:3 64:14 71:6 99:11 146:15,25 147:2 rebutted 112:1 recall 61:21 recap 45:18	receive 28:24 55:6 67:3 76:6 78:22 79:5,15 90:18 91:9,16 91:17 92:3,17 92:19 101:16 122:5 143:10 143:21 144:23 145:1,18,20,23 145:25 148:1 171:3 received 28:14 28:18 29:11,25 31:25 32:7 38:18 58:6 91:1 144:11,14 148:2 169:1 receives 78:12 109:12 145:23 167:14 receiving 78:24 145:12 164:1 recent 16:9 25:22 recitation 75:5 recite 153:2 recited 12:22 75:17 77:2 150:19 recites 55:5 66:25 recognized 62:14 record 69:21 71:19 98:15	110:5,23 113:23 123:25 136:8 146:17 165:24 records 169:12 recourse 103:11 rectangle 95:13 red 29:6 33:13 54:20 95:13 redesign 27:12 27:13 58:19 69:18 114:14 redesigned 20:12 redo 25:17 102:14 reduced 172:9 reel 86:22 refer 20:2 reference 48:7 48:20 49:23 56:24 63:19 67:23 158:25 159:7 referenced 34:16 149:11 references 69:12 127:20 referencing 50:3 referred 73:2 75:2 141:8 referring 59:10
---	--	---	--

refers 157:18	125:17 126:13	released 144:20	removed 9:15
reflash 53:3	126:15,20,24	160:19	16:2 17:9 18:9
reflect 170:11	131:18 136:18	relevant 20:18	20:25 22:1,8
refuse 53:20	138:19,20	30:23 35:22	23:6,9 26:2
refused 98:2	154:9 157:12	54:7,9 69:2	29:17 30:9
110:5,23	157:16 162:22	76:19 100:24	34:19 38:5
125:18	162:23	100:25 123:18	40:1 42:19
regard 127:12	reinman's	151:20 162:15	54:25 61:25
regarding 8:19	21:15 46:9	164:23 165:14	65:19 70:12
16:11 17:3	63:3 97:17	relied 33:24	87:8 90:7
18:2 51:18	99:12 105:10	34:7,18 91:3	139:25
74:10 82:9,12	105:14 106:20	92:6 101:23	removes 31:15
114:20 139:12	114:25 115:10	149:2	35:9 89:22
143:3	119:6 126:7,22	rely 92:4	renaud 3:7 6:1
regardless 18:2	143:15 154:8	relying 91:6	render 115:17
18:4 19:23	rejected 29:10	remainder	repeat 141:4
52:22 54:16	29:16 68:21	41:14	repeated 63:5
65:9,10 142:23	70:18 77:13	remaining 58:2	repeatedly
158:4	87:21 91:21	remains 81:2	126:13,15
regards 80:6	93:24 116:1,8	remedial 80:25	replace 96:5
regulations	117:17	remedy 49:11	97:20 165:23
6:20	relate 96:20	51:13,15,18	reply 10:15
reiman 125:21	111:9 164:17	138:4,6	17:20 65:2
154:17	related 6:16	remember 12:9	99:18 132:7
reinman 21:5	60:22 74:13	12:18 14:10	150:20
32:14,19 34:3	112:11 115:15	34:8	report 15:20
44:2 45:21	121:12 139:24	remote 1:17	123:23 138:21
46:7,15,21	162:3 172:12	removal 69:16	143:16
62:13 64:2	relates 20:24	69:19,24	reported 1:20
91:2 97:8	29:3 66:24	122:25	172:9
99:10 104:11	161:23	remove 15:11	reporter 6:8,10
106:4 107:21	release 20:2	20:13 39:24	reports 107:6
113:24 114:22	22:11,17,17,21	60:8 64:11	representative
115:2 117:6	22:25 36:24	70:20 71:8	15:7 109:7
119:7,8 120:17	62:8 107:2,3,8		119:25

represented 145:5,6	130:12 150:13 155:7	requiring 79:10	146:12 157:8 161:16
representing 5:22	required 8:6 18:3 31:13	requisite 30:1 resolve 110:3	responses 166:20
reprogrammed 96:10	32:18 34:13 37:8 41:18	resolved 88:10 resource	responsible 124:15
reps 98:6	56:7,8,21 57:2	117:10 145:13	responsive 132:15
request 6:14 7:2,8 9:13 10:15 15:2,5,6 15:16 16:1 19:23 20:10,14 22:23 26:15 39:10,14,22 62:9 69:23 99:7,21 100:2 100:8 101:8 109:16 113:3,8 124:22 131:9 132:1,2 134:2 153:25 163:12 163:18,18,21 164:3,11 166:12,22	57:4,16 65:12 67:11,17,19 68:14,19 78:16 79:16,25 88:8 89:15 90:21 91:13,18 92:5 98:19,23 107:24 115:6 117:19 122:6 142:7,9 147:25 148:18,20 149:15 150:8 150:10,11 151:7,12,25	155:14 resources 59:21 113:14 114:10 116:25 117:4 143:7 156:2,10,19 respect 25:9 56:23 80:20 85:15 94:22,24 114:17,21 115:24 117:25 122:23 126:1 133:11 155:18 161:7	rest 63:8 64:14 155:25 restrictions 84:20 result 80:18 114:3 115:18 118:3 136:7 144:22 reveal 110:20 review 77:4 105:1 112:14 165:15 167:24 reviewed 37:11 117:5 154:8 revolves 30:10 rhetoric 10:14 10:17
requested 9:22 110:22 121:24 131:10	requirements 112:20 141:24 141:25	respectfully 99:19 132:1 respective 143:25 respond 137:10 respondent 52:19 respondents 88:5 91:14 responding 155:9 response 31:17 99:12 132:6,22 134:2 135:20	rid 63:6 68:2 right 27:16,20 28:22 29:19 30:3 33:9 38:11 50:18 59:19 62:8 85:22 86:14 95:1 106:19 111:21 117:1 129:19 130:11
requestor 5:10 9:6	requires 28:13 28:21 34:5 55:9,18 66:10 66:12 76:17 78:21 94:13 115:22 117:18 150:11,24 153:3,7		
requests 35:5 61:21 134:6,8 152:6			
require 32:4 66:15 76:7 94:9 114:4,11			

139:7 140:21 141:14 145:9 155:23 170:1 rightly 101:23 rizk 2:3 4:6 5:20,21 7:20 7:20 8:22 71:25 72:4,5,5 77:3 78:19 133:2,5 137:12 140:4 141:6,8 143:13 146:22 147:6,11 148:23 149:19 151:21 152:11 153:16 155:3 155:23 157:11 157:13 158:13 159:5,10,13 160:14 162:20 162:22 164:16 166:5 169:10 169:10,23 170:1,19 171:10,14 rizk's 151:17 155:17 rmr 1:21 road 141:16 roadblocks 107:14 robin 63:25 70:9 roku 96:16	role 41:2 107:7 roughly 80:9 round 63:25 70:9 rpr 1:21 rtd 86:15 rtk 97:18 rtl 17:15 67:13 70:7,7 81:20 82:14,16,17,21 82:25 83:1,7 83:17 90:23 91:7 154:4,11 rubber 105:4 rule 166:16 ruled 67:24 68:13 ruling 6:15 7:2 7:8 9:13,22 11:5 15:2,5,6 15:16 16:1 19:23 20:10,13 22:23 25:22 39:10,15,22 54:6 55:8 56:16 69:23 94:22 99:5,18 100:8,14 122:8 131:9 148:8 163:12,15,18 163:18,19 164:3,6,7,11 166:12,14,17 166:22	rulings 6:21 38:20 69:13,15 82:12 111:9 run 37:2 38:16 40:5 41:10 43:7,10,10 44:16 48:15 51:8 63:4 107:20 108:21 109:24 159:23 161:22 running 40:9 41:16,19 45:23 49:8 62:25 109:20 124:21 161:25 162:9 162:10 s s 2:3 salient 170:9 sam 3:8 6:1 sample 100:16 100:23 103:14 106:2 133:21 135:8 136:1 164:23 samples 102:19 santa 72:13 satisfy 14:16 53:24 56:21 69:17 77:9 123:19 138:3 save 23:23 38:10 64:14	saved 8:8 saw 25:21 32:10 46:18 65:3 142:16 161:9 saying 65:12 97:12 118:5 124:5,6 126:10 126:11 134:10 139:13 159:6 164:16 166:7 167:5 says 22:17,25 22:25 25:2 36:25 50:17,25 55:18 62:1 74:24 75:8 99:17 104:12 107:17 116:10 116:17 117:3 128:21 129:5 139:17 140:19 149:8 155:13 156:9 158:14 159:10,18 167:8 scale 74:4 scan 128:14 129:25 130:1 142:5 scanning 17:6 scenario 51:20 60:23 114:8 schedule 23:1 55:6 157:22
---	---	---	--

169:19 scheduler 129:10 149:9 schedules 130:6 scheduling 14:6 18:3 31:4 31:6,13 54:11 54:16 55:3 56:1,5,11,19,20 57:1,8,20 65:10 66:24,25 67:4,6,11,15,24 68:23 75:15,18 75:25 76:3,6,8 76:22,23 77:18 77:19 78:7,13 79:1,3,6 87:24 88:7,12 89:1 89:14 90:18 91:9,16,17,18 147:10,18,22 148:3,13 149:6 149:11 151:6 151:11 scope 10:21 11:6 15:16 18:6 19:5 23:12 26:5 51:19 52:1,14 163:12,16 screen 9:3 127:6 133:3 147:7 153:17 155:21 156:7	160:3,3,5,8 163:14 164:19 second 9:3 11:18 18:18 28:17 40:11 45:21 56:12 57:1,9 58:16 66:7 67:1,4 68:24 75:6,15 75:19 76:9 77:16 79:11 88:12 92:2 104:20 116:18 132:12 143:24 147:19 148:4 148:13 149:7 149:13 158:11 167:19 170:3 secondly 13:9 secret 20:3 123:11 section 17:20 30:5 53:7 66:23 80:14 128:17 150:21 sections 9:20 see 9:9,24 12:3 12:21 13:17 15:17,20,22 19:8 20:3,22 29:19 30:10,24 31:12 33:23 35:4 39:18 44:2 46:3 48:9 50:17,18 51:11	61:2 62:8 63:8 66:8 73:9 74:5 74:7,15 79:7 79:18 81:18 83:23 86:13 87:14 89:9 92:3 95:12 102:6 106:19 109:1 112:7 116:9 118:22 121:24 129:23 133:3 142:11 145:5,8 153:22 166:19,19 seeing 8:23 82:19 146:8 168:20 170:14 seeking 39:1,15 39:22 99:4,18 100:14 122:7 123:23 124:25 134:4 seemed 164:17 seems 15:2 50:7 100:10 132:18 132:19 137:5 167:7 seen 19:21 38:8 select 115:25 121:6 selected 72:20 119:22 121:5 selection 64:1,1 64:5 70:9 119:20 120:9	selects 21:17 sell 84:1,9 semiconductor 2:19 6:15 send 16:15 63:1 128:14,16 142:21 161:1 171:6 sending 27:24 sends 31:8 67:15 70:10 129:19,21,24 129:25 130:11 sense 61:12,17 132:16 140:3 145:15 163:8 sensors 17:6,7 sent 12:19,24 13:16 35:8 58:6 sentence 116:10,12,14 116:14,16 117:2 sentences 116:15 separate 16:24 34:10 43:2 75:5 127:23,24 162:16 166:16 166:17 separated 32:20 separately 31:12 96:2
--	--	--	---

sequence	14:14 15:13,14	109:3,10,12,16	160:14,16,23
111:20 120:5	27:24 28:14,19	109:19 110:4,7	161:7 162:9,11
120:14,19	28:24 29:9,11	114:10 117:17	shading 13:21
121:1,2,2,3	29:24,25 31:16	118:2 122:5	14:15 15:14
sequential	31:19,21,23	125:11 126:19	61:13,14
58:14	32:7,12,12,13	127:3,8,9,17,19	share 9:3 63:9
serialization	32:16,16,17	127:20,21,21	63:10 116:18
37:3 146:2	34:11 35:4,15	127:21,22,25	127:5 134:8
serialized	35:23 36:2,9	128:7,8,11,11	147:7 155:21
118:11	37:5,17,22	128:16,21,24	159:12
serializing	38:18 39:2,5	128:25,25,25	shared 162:13
122:24	40:6,9,20,21	129:1,2,3,4,7,7	sharing 114:10
serially 145:17	41:9,15,17,19	129:7,8,8,17,18	153:17 163:14
set 21:1 30:13	41:20,23 42:4	130:5,7,7,12,13	shift 106:21
43:9 58:4	42:4,6,14,18,20	134:5,6,17,22	shifted 134:17
60:15 62:5	42:21 43:1,1,2	134:23 135:3	shifting 108:21
78:11 86:7	43:3,10,11,13	135:10,12	ship 18:24
120:11 143:15	43:13,14,19,21	137:21 138:12	86:24 87:1
152:10 172:8	44:4,4,12,16,21	139:3 140:8	shipped 84:13
setting 65:24	45:2,4,5,7,8,8,8	141:21,22,25	shipping 83:21
settings 60:10	45:14,22,23	142:7,13 143:9	shortly 79:7
151:4 167:14	47:25 48:1,1,3	143:21 144:11	shot 130:25
167:15	48:10,10,14,21	144:12,14,21	show 33:6,10
settled 80:24	49:2,4,8,18,18	144:22,25,25	35:3 91:8
setup 168:13	50:25 51:2,3,4	145:11,17,22	93:22 94:4,13
seven 129:14	51:5,8 59:17	155:20 156:16	98:20,24 113:6
130:3	60:3,12 61:23	157:19 159:20	124:24 145:4
several 17:17	67:16 68:3	160:13 161:3	148:6,25 155:7
18:20 89:22	72:21,24 75:10	162:15 163:7	showed 42:25
98:18,18	79:12,14,17,18	163:24,25	56:18 126:8
108:11 109:4	79:19,25 92:2	167:25	161:14
150:21,23	93:13 105:17	shaders 35:5	showing 33:8
151:15	106:15,22,24	44:3 45:9,16	108:5 149:6,25
shader 12:11	107:10,21	129:11 142:2,3	150:3
13:7,7,8,11	108:9,13,24	158:25 159:23	

shown 13:3 18:9 21:3 33:13 55:7 67:21 83:24 88:17 92:13 95:11 131:3 133:9 136:2,13 148:20 149:16 149:18 shows 123:6 149:9 155:25 side 29:23 37:7 50:21 83:24,25 108:6 side's 8:16 sift 154:10 signature 172:19 significant 72:12 94:14 138:14 silent 88:9 silicon 83:8,23 86:18 89:2,13 91:8 92:10 149:20,20 silver 83:24 similar 29:5,22 36:22 66:4 110:11 155:24 similarly 17:3 145:22 simple 20:23 130:21	simpler 80:1 simply 10:1 14:18 15:6 17:14,18 18:14 20:18 22:4 26:23 27:2 29:1 34:22 35:21 54:9 55:25 57:24 58:14 123:12 124:19 129:12 148:8 150:15 156:17 157:24 168:2 simulation 48:25 138:19 simulations 48:22 126:9 138:18 139:5 139:16,23 simultaneous 125:6 126:3 simultaneously 109:17,21 126:9 135:11 single 128:22 129:9 sink 53:13 sit 29:19 31:21 61:1 site 130:2 sits 29:20 sitting 28:24 61:25 113:15	situation 41:1 six 9:24 60:20 110:2 125:4 140:21 skill 88:11 slide 12:19 21:4 21:4 27:16 30:4 33:23 34:25 40:16 46:18 50:16 56:17 62:7 65:8 69:11,12 69:14 70:15 74:6 83:13 87:14 92:13 95:11 106:20 111:22 126:6 131:23,23 132:19,20,21 132:25 134:10 152:4,20 153:19,23 155:15,15,16 155:21,23 156:21 159:2,2 166:5,6 slides 9:3 17:21 20:23 23:21 30:20 41:5,7 132:19 134:8 144:4 155:3 slightly 41:22 slot 13:20,21 14:18,19 15:12 15:21 21:10,10	21:13,13,17,18 21:19,23,24 22:2,7,7,7,18 22:25 23:2,3,4 23:5,7 27:12 27:14,14,20,25 28:1,2,4,16,17 29:4 34:8,10 34:16,17,20,22 35:16 36:5,7 36:11,13 37:15 59:3,12,13,14 60:1,11 61:1,1 61:3,4,8 70:12 73:22 87:7 89:23,24 90:2 90:7,16,24 109:13 111:19 118:13,15,17 119:10,10,11 119:14,15,20 119:21,21,21 120:1 156:23 156:25 157:11 157:14,19,23 157:25 158:6,7 162:13 slots 12:2,5 13:18,24 14:5 21:10 27:18,22 27:24 29:8 30:25 31:2,3 34:10,15 38:2 68:16 77:7,8 89:21,21 90:10
--	--	--	---

118:20 119:1,2 119:12 157:2 157:20 158:3 162:16 slow 35:3 small 47:1 smoke 164:18 soc 24:18 26:20 socs 51:24 soft 62:24 software 16:11 17:4,9 25:24 51:25 64:24 66:18 68:15,21 74:18 81:20,22 81:23 82:1,15 82:23 84:7 85:4 91:12,19 95:4,17,21,25 111:6 117:22 123:13 150:8 150:19 152:5 152:13 166:8 sold 84:12 86:20 soldered 95:14 sole 14:20 147:13,16 155:13 solely 147:11 149:1 solution 103:17 103:20 120:2 somebody 96:7	somewhat 60:22 139:24 sony's 74:9 sophisticated 24:5 84:3 85:10 sorry 76:13 101:4 112:17 113:19 116:15 122:2 124:11 129:22 sort 120:6 132:4 138:7 source 20:1 22:24 25:18 39:14 46:6 81:19,20,21,22 81:22 82:2,4,6 82:14,15,17,20 83:17 84:18,19 90:23 91:7,7 95:7,8,23 96:2 97:9 98:11 101:25 102:1,4 102:7,21 105:1 105:13 112:5 121:24,25 122:11,19 123:16,21 124:1,1 125:19 125:20 136:3 136:21 151:22 151:23 152:12 154:9,11 167:6 168:14,15	sources 22:10 space 72:17 167:11,12,12 spare 147:5 speaking 7:13 7:16,19,21 8:17 spec 116:9 special 128:4 130:12 138:23 141:23,24 142:1 specially 118:3 specific 9:16 16:7 89:25 111:15,18 113:10,19 115:4,18 129:3 137:3,18 139:1 140:18 141:19 148:3 151:19 163:16 164:7 specifically 15:8,15 16:2 19:5 23:8 33:24 37:7,24 39:24 46:10 47:14,17 53:23 54:18 56:10,18 57:7 62:7,11 74:11 97:19 100:11,11 113:20 116:2 138:17 139:4 139:18 148:16	149:16 154:19 156:3 157:18 158:22 164:4 165:16,25 166:8,21 specification 138:15 141:22 specifies 83:1 specify 148:2 speculation 166:3 speculative 26:12 spend 9:11,25 10:3 11:18 12:1 14:25 19:24 38:22 48:18 53:9 spending 10:13 spends 19:20 spent 17:12 151:21 152:11 splashed 52:12 splits 129:18 spot 28:5 squarely 77:3 77:13 87:19 117:16 ss 172:3 stage 21:1 138:5 stages 144:10 stamped 105:4 stand 81:13 84:1,2 86:9,9
---	---	--	--

88:20 94:19 164:7 standalone 81:5 86:20,25 87:12 standard 72:22 74:23 86:6 98:19 99:23 125:15 159:22 160:9 standards 98:24 standing 152:15,21 start 20:21 21:2 27:25 30:22 38:12 49:18 50:14 51:1,2,4,5 55:3 55:4 61:16 82:6 86:8 112:24 123:14 147:7,9 157:11 started 6:12 9:7 11:8 28:2,4 starter 51:4 starting 27:22 68:8 105:6 130:19 starts 132:20 156:1 state 90:3,13,13 90:17,24 123:24 172:2	stated 55:11 56:10 152:3 160:14 statement 109:8 113:1 144:1 159:14 statements 101:10 172:8 states 1:1 72:12 81:1 83:11 86:21 87:23 97:18 100:17 103:24 130:4 stenographic 172:11 stenographic... 1:20 172:9 step 10:19 25:9 26:16 28:11 29:17 36:3 58:1 66:3 78:24 stepped 28:2 steps 17:1 41:18 steve 3:15 5:13 stock 96:13 stop 9:20 42:1 62:24 63:1 153:17 163:13 166:25 stored 96:1 story 71:6 straight 19:7	straightforward 9:14 16:1 30:7 94:20 strange 154:21 stream 49:8 strongly 88:25 stuttard 56:24 57:4 67:23 sub 160:10 subject 16:21 20:13 38:4 80:25 81:2,4 95:18 104:17 152:22 submission 102:24 125:11 submissions 7:10 168:21 170:15 171:1 submit 34:20 34:21 36:10 61:8 122:3 125:6 126:4 136:19 submitted 6:15 35:18 36:4,24 37:14 38:18 43:25 79:22 109:23 162:16 171:1 submitting 124:16 subsequent 103:10	subset 108:19 137:21 substance 154:22 substantially 116:25 143:6 success 74:22 sufficient 18:5 100:2 113:7 118:15 sufficiently 16:20 suggested 11:12 87:10 114:4 suggestion 93:19 suggests 118:18 sum 108:4 109:10 131:25 166:12 summarily 64:10 summarize 37:18 127:5 summary 126:6 130:17 131:23 summation 132:5 sums 113:5 supply 83:14 84:3 support 22:10 36:23 89:15 114:20 153:25
---	--	---	--

supported 112:10 126:11 126:19,21 supports 88:25 supposed 24:20 99:8 sur 10:15 17:20 65:2 sure 8:17 20:3 24:12,14 27:15 38:7 40:17 48:19 49:24 58:17 65:3,6 66:13,22 70:2 89:4 96:15,16 155:7 156:6 162:7 164:16 165:9 166:14 surprise 17:19 123:9 surprised 100:9 suspect 103:16 sutcliffe 2:14 swap 167:22 sync 95:14 synthesizer 82:22,23 83:3 synthesizing 83:20 system 75:16 76:10 77:1,20 77:22,25 81:25 88:6,8 89:1 94:25 95:1,1	95:20,25 96:16 96:17 124:15 133:12,13 148:9,11,16 151:22,23 152:8,12,17 systems 1:7 101:21 t ta 1:8 6:18 table 4:1 take 23:2 26:4 27:3 40:4 58:8 64:17 66:22 70:3 71:16 81:10 83:4 89:4 95:4 100:5 101:18 102:15,16 104:8 105:5 107:16 113:9 119:14,16 134:11,14 139:12 146:9 165:23 168:1 taken 1:17 34:22 65:16 103:9 172:7,11 takes 22:19 45:11 53:7 62:4 talk 11:19 13:24 14:8 44:13 54:11 74:10 88:21	110:7 144:5 talked 13:15 70:15 116:24 117:15 155:19 talking 9:25 10:23 12:1 44:14 46:20 48:12 61:24 74:17 79:4 82:7 85:24 106:11 147:21 151:21 152:12 164:25 talks 44:19 58:12 59:19 159:19 160:22 tape 86:22 target 100:10 task 32:8,9 33:12 34:2,5,6 35:10,10,13 37:21,22 40:13 40:22,23,25 41:2 44:21 46:4,5,11,14 48:8 50:15 51:2,3,6,7 55:14 58:25 66:12 73:2,8 73:13,16 79:10 79:11,11,12 80:17 89:11 106:14,15 107:23 108:1 108:22,23,25	109:3 111:18 113:10,13 115:3,4,23,23 116:3,6,24 117:4,9,10 120:15 127:8 127:10,16 128:3,4,12,13 128:14,22 129:9 130:11 132:11,12,12 132:13 137:4,7 137:9 138:20 139:14,22,22 140:1 141:20 143:11,12,23 143:24 156:7 158:17,17,21 158:21,24 162:6,11,21 163:7 tasks 12:22,25 13:6,10 14:15 21:23 28:25 31:25 32:2,6 32:11,16,19,25 33:1,2,11,25 34:7,9 36:18 36:18 44:20 45:20 46:2,11 47:17 49:15,15 49:17,20,20 50:2,2,6,8,10 50:14 58:21,22 59:1,2,20
---	--	---	--

67:15 73:17,19 76:7 78:13,24 79:6,13,15,20 90:25,25 92:19 92:20,24,25 93:15 94:3 106:10 108:7,7 108:13,14,22 109:21 110:8 111:19 113:15 116:18 118:8,9 120:10 122:4,5 122:24 124:16 126:5,8 129:18 129:24,25 130:6,13 134:21 135:1,2 135:12 136:20 137:15,16,18 137:21,24,25 138:3,13 139:12,19 141:17 142:10 143:10,24,25 145:1,12 148:2 156:2,14 158:13 159:9 159:16,17 160:23 162:10 163:24,24 164:1,2 tawfik 3:11 6:3 tcl 19:3 24:16 24:23,24 25:6 25:11 26:7,8	26:11 52:19,19 80:7,19,24 95:1 98:10 121:25 165:22 165:25 166:7,8 166:9 167:20 tcl's 25:1 team 107:4,8 teams 1:17 5:1 8:13 technical 8:9 81:18 86:5 97:7 104:17 108:3 110:6 113:25 technically 81:3 techniques 123:10 technologies 2:11 5:23 6:5 7:5 technology 72:8 73:16,25 74:12,17 80:8 80:17 83:16 85:10 television 167:20 televisions 1:9 19:3 24:16 80:20 tell 28:17 29:2 29:10 97:12 155:3 158:20	158:23 159:6 telling 162:19 162:24 163:5 tells 11:24 31:9 60:3 161:21 165:16 temporal 116:17 temporarily 113:14 116:24 117:3,9 143:4 143:5 156:1 term 14:4 93:19 115:13 116:3 119:4 128:20 termination 163:21 terms 12:14 14:1 24:8,13 69:2 81:17 109:18 124:12 terrill 3:4 5:21 6:23 24:4 27:6 30:2 40:15,18 58:16,18 60:18 64:18,21 66:7 68:5 69:4,8 71:3,13 76:13 78:18 96:22 100:20 121:12 132:3,4 133:3 133:4 135:20 137:1 146:7,11 167:2 169:4,16	169:21,24 170:7,19,23 171:7 terrill's 31:17 54:23 82:8 157:9 169:19 tessellation 129:8 test 85:3 95:6 96:11 100:18 165:4,12 tested 26:22 101:23 134:1 165:5 168:12 testified 21:16 34:3 91:3,3 98:7 117:7 119:8 126:15 126:24 135:22 136:16,17 154:23 testifier 107:7 testify 46:16 154:15 testifying 98:9 98:10 testimony 21:15 84:22 97:24 98:5,13 105:11 107:25 113:23 115:10 115:12 119:6 134:16 149:5 155:1 162:23
---	--	---	---

testing 110:18 110:19,21 114:5,11,12,13 114:16,22,25 155:7 tests 114:20 126:7 tgc 64:3 79:21 79:24 92:7,7 92:22 93:3,13 94:2,6 thank 5:14,16 5:17 6:6,11,13 7:18,22 9:2,7 27:6 30:2 51:11 60:18 64:14,16,21 69:4,7 71:3,13 71:15,23 72:1 72:5 78:18 132:3 137:1 139:10 143:2 146:4,10,11,15 146:20,22,24 147:3 167:2 168:4,16,18,19 169:8 170:1,19 170:20 171:7 171:10,11,13 171:14 thanks 7:10 that'd 52:13 themselves 47:9 theoretically 9:20	theories 19:12 57:16 132:15 theory 25:5 132:7 thereof 1:8 thing 8:15 26:14 32:25 46:25 47:15 127:5 141:10 141:19 148:3 160:25 162:19 163:11 165:20 things 63:10,10 63:19 73:15 101:21 138:14 153:16 156:13 157:25 161:19 think 16:1 19:7 24:6,12,13 26:6,11 27:1 41:5,25 42:3 50:10 52:24 53:5,9 58:18 58:24 60:19 62:23 64:25 68:6 69:1 70:2 72:9 76:14,15 76:16 77:7 85:7,25 89:13 93:10 96:22 99:2 113:4 114:15 116:23 117:14 118:14 120:5 121:12 137:6 138:14	138:23,24 140:11 143:1 145:3 146:3,13 158:18 161:21 162:19,24 163:5,14 166:11,18 170:12,15 171:3 thinking 169:17 third 18:24 22:15 52:10 84:6 thought 45:12 164:15 thread 128:23 threads 128:23 three 9:20 11:20 12:3 21:9 63:10,19 70:6 80:23 83:13 90:10 99:3,19 101:9 111:15 113:9 127:20,22 128:7 135:19 156:12 165:5 166:18 168:10 168:11,12 throughput 60:24 61:16 throw 117:15 thursday 100:20	tie 143:8 tied 115:23 time 5:5 6:8 8:7 8:7,10,14 9:8 9:11,25 10:3 10:13 12:1 14:25 16:25 19:20,24 23:22 26:17 27:1 33:12,14 34:6 34:12,21 35:11 36:2 37:6,17 37:23 38:1,2 38:22 42:1,11 44:1,6 48:18 49:6 53:10 58:2 60:14 61:9 64:15,17 69:7 71:14,15 72:2 80:10 81:12 89:22 92:20,21 93:21 94:10 99:15 100:6,8,23 102:13,14,16 102:22,23 116:18 117:16 117:19 121:10 126:21 128:20 129:13 144:9 145:14 146:1 146:14 147:5 151:21 152:12 153:9,11,14 159:22 160:15
---	---	--	--

162:2 165:6 168:9,13,22 170:21 timeline 73:22 73:24 times 94:3 108:11 109:4 156:9 tiny 11:25 tis 150:22 title 73:8,9 today 5:10,24 7:21 9:12,19 10:14,21 11:15 11:19 12:15 14:8,25 20:4 23:25 32:6 48:5 77:15 85:24 94:11 100:6,8 164:15 168:19 today's 6:13,19 7:11,14,19,23 8:20 171:12 together 101:22 102:6,7 141:2 143:8 told 105:18 161:12 tomorrow 100:23 took 22:1 30:8 33:16 126:21 146:13	top 11:22 31:5 31:23 55:7 116:12 123:14 123:14 130:20 topic 32:5 153:15 158:11 tortured 112:19 touched 83:12 towards 73:7 trade 1:1 6:17 6:21 transcript 110:1 114:6 120:2 125:2,3 156:3 170:25 171:6 172:11 transcription 172:10 transcripts 125:1 transfer 83:5,6 transform 44:23 transformation 48:11 transformatio... 160:5 transforms 108:22 trial 45:2 154:23 tried 26:18,19 46:15,16 49:6 53:2 77:17	96:19 107:11 115:23 116:7 119:6 140:9 tries 40:10 101:15 102:25 true 36:12 59:14 106:7 111:3 117:24 121:23 161:24 172:10 truly 107:13 truth 105:3 try 26:4 27:4 39:5 47:13 55:17 57:15 62:16 143:16 143:17 147:4 154:12 trying 10:8 19:24 40:12 47:7 51:21 54:4 55:1,21 63:11 68:2 69:1 132:6,16 153:6 163:6 164:19 tsmc 83:6 turn 8:16 turning 73:6 79:9 tv 95:1 166:9 tv s 26:11 two 9:21 11:14 12:5 13:18 14:15,19 15:7	15:12,21 17:16 18:9 20:19 21:10,13,18,23 22:2,7,7,10,18 23:1,2,3,5,7 27:12,19 28:4 28:18 31:2,3 32:6 34:15,16 39:24 42:9,20 42:21 44:10 51:16 53:11 54:10,23 55:13 59:5,5 61:3 63:10 64:19 65:6,7 66:20 70:12 75:7,10 77:10 79:13 80:10 81:19 85:23 87:7 89:23,24 90:2 99:3,6,7,19 101:9,10 109:20 115:25 118:7,10,13,15 119:11,15,21 120:6 130:21 130:24 131:8 139:2,18 140:21 141:1 153:7 157:25 158:6,7 160:11 162:16 163:25 164:4,5 166:13 166:18 168:10 169:11
---	---	--	---

twofold 36:24 type 43:9,14 51:7 86:22 95:10 104:7 106:14,15 113:12,17,21 114:2,3,10 115:3,11,18 116:20 119:13 119:18 127:24 129:3,3 130:10 134:18 141:19 141:23,24 142:1 156:13 types 13:23 14:15 48:13 59:6 79:13,20 81:19 92:24 93:15 97:13 98:1 100:3 109:20 111:15 118:7,10 124:22 126:5 127:20,22 128:8 129:17 137:24 140:7 140:14 144:13 typical 85:2,5 typically 95:16	97:7 umc 83:6 unable 110:22 unapproved 24:9 unattached 99:4 unchanged 28:9 70:7 93:1 unclear 97:10 under 6:16 38:19 40:20 70:3 76:16,22 115:7 118:2 131:6 137:7 149:14 underlined 116:11,13 117:3 underlying 7:5 74:16 106:24 108:20 understand 8:9 27:10 64:22 66:8,13 88:11 117:7 139:10 140:13 146:3 understanding 40:18 50:5 132:23 169:24 170:14 understood 76:15 160:25 undisputed 123:5 137:20	undisputedly 101:3 undue 20:17 25:24 26:4 27:4 38:13 39:23 40:3 44:11 51:21 56:13 57:10 unequivocally 22:18 unfortunately 107:1,3 140:8 unified 72:21 72:24 uniform 141:11 uniforms 107:19 141:1,6 141:9,12 142:19 unique 73:14 unit 56:12 57:2 57:9 67:1,5 68:24 72:16,23 73:10,13 74:25 75:2,2,4,6,9,12 75:14,20 76:9 76:9 88:12 90:19 128:15 147:19 148:4 148:14,17 149:7,14 united 1:1 72:12 81:1 83:10 86:21 87:23 100:17	103:24 units 12:20 28:23 35:24 47:18 87:16 157:24 unknown 123:7 unmodified 63:20 unrealistic 114:25 unreasonably 98:22 unrebutted 105:11 108:2 112:1 115:11 126:23 134:16 162:23 unrestricted 84:17 95:9 unsubstantial 108:2 unsupported 118:24 untethered 104:22 updated 8:14 uphold 57:3 use 8:6,8 23:6 24:7,22 34:25 52:14 57:3 70:12 85:1,2,5 86:6,24 89:3,6 94:5,14 96:11 96:25 97:22 98:1 111:14
u			
ucl 7:5 ulc 2:11 5:23 ultimately 73:4 80:4 82:20 85:16 91:20			

122:4 123:10 142:14 149:21 164:19 167:25 used 8:8 15:22 17:5,10 21:5 22:3,18 24:13 24:20,21 29:13 33:7 35:1,2 41:14 42:14 43:5 44:24 48:1,21 50:15 50:22 51:6 52:22 56:8 73:3 76:3 81:23 94:5 95:3 100:25 123:11,11 141:16 useless 11:25 26:23 user 85:9 167:12,17,20 uses 40:6 45:14 using 30:16 41:23 43:20 47:25 80:7 82:16 85:2,4 112:10 121:6 122:2,2 124:10 135:23 163:22 usually 81:25 utilize 74:18	v validity 47:22 48:13 55:23 56:15 57:2,3 57:18 value 16:16 48:11 58:4 62:9 71:1 84:3 107:17 120:23 120:24 152:10 various 9:12 68:9 vector 49:1 velocity 49:1,6 ventures 150:23 verbatim 21:22 version 96:14 143:18 168:6 versions 167:10 versus 15:24 34:1 140:1 150:22 158:21 vertex 13:7,22 14:15 28:16 32:12,16 39:2 39:5 40:5,6,6,7 40:8,9,20,25 41:11,15,19,19 41:20,23 42:3 42:6,14,16,20 42:21 43:1,2,5 43:9,10,13,15 43:17,19,20 44:6,8,9,16,22	44:24,25 45:4 45:4,8,14,23 48:1,3,9,10,14 48:21 49:2,7,8 49:8 50:21 51:4,9 61:14 62:2 105:17 106:5,8,9,12,14 106:15,21,24 107:10,21 108:9,13,22,24 109:3,10,12,19 109:23 110:4,7 118:8,9 122:3 124:10 126:12 126:14,17 127:8,9,20 128:5,6,9,20,25 129:2,7,13,20 129:21,24 130:4,10,13 134:6,17,22,23 134:24,25 135:1,2,10 136:11,20 137:20 138:12 139:3 140:7,17 141:21,22,25 142:2,7,12,14 142:16,24 159:12,24,24 159:25 160:2,4 160:12 161:3 161:18,21,23 162:3,4,10,10	162:12 vertices 41:12 41:13,21 42:10 44:10 49:1 126:14,16 159:25 160:4 vhdl 82:18 videoconfere... 1:17 5:1 view 53:17 107:9 violation 80:14 virilog 82:18 virtual 68:20 virtue 152:5 volume 84:10 86:22,24
			w
			waived 162:5,7 waiver 108:11 108:16 137:13 walk 15:4 30:14 53:8 71:4,5 89:9 158:4 walked 36:22 149:5,16 159:19 161:16 163:3 walking 143:16 163:8,9 want 9:7,12 10:13,20 11:17 12:15 15:4 20:6,19 24:14

34:24 38:10,12 39:2,3,19 48:18 51:14 54:10 55:4 58:20 84:18 85:3 95:9 96:4 96:15,16 98:8 100:17 101:17 102:6 103:21 103:23 139:8 145:4 147:5,7 147:9 153:15 163:11 165:20 166:14 wanted 65:4 78:1 110:15 112:6 138:2 169:13 170:3 wants 97:22 98:11,12 102:15 152:8 166:8 warning 8:13 8:13 warp 64:1 washington 1:2 2:16 wasteful 107:20 watch 16:16,16 16:17 65:24,25 66:1 94:25 97:16 watches 16:11 16:12,14	163:20 water 158:12 waved 127:12 way 30:18 31:18 41:17,23 44:5,7 53:16 57:10 60:7,16 61:10,25 65:21 67:24 68:12 70:18 71:10 78:11 93:4,20 97:1 105:25 111:23 117:8 119:11 121:6 123:9 128:2,6 142:3,10 158:4 158:8 163:8 wayne 172:4,23 ways 48:4 59:23 121:4 140:6 163:6 we've 9:18 10:18 12:4 16:5 17:1 18:8 18:9 22:8 29:16,17 35:11 36:20 39:17 40:1 43:6 54:19,22,25 58:3 66:3 83:12 150:20 153:8 wednesday 1:19 5:2	weigh 113:22 weighed 107:22 114:2 115:10 120:16 124:8 weird 112:10 welcome 8:5,6 78:19 went 17:10 21:16 29:23 36:3 65:22 100:11,11 102:2 147:12 152:18 154:19 156:3 159:13 whatsoever 84:21 87:5,9 106:23 109:20 127:19 white 5:9 wide 74:4 win 85:17 wish 125:14 withdraw 159:14 withdrawing 165:6 withdrawn 99:21 165:11 withdrew 99:7 withheld 113:21 withhold 96:19 114:13 witness 97:24 105:19 110:20	113:25 114:1 witnesses 84:22 98:2,6 wondering 137:8 168:24 word 29:13 119:15,17 164:19 words 20:3 21:18 47:24 53:25 work 12:20 31:8 48:16,16 62:3,12 84:8,8 99:11 101:21 105:4,7 124:22 131:2 151:25 160:19 161:5 165:11 worked 39:19 165:8,11 working 99:20 141:12 workloads 96:12 114:19 works 96:13,15 96:17 101:20 118:25 126:25 128:7 158:4,8 167:16 world 72:23 worry 55:22 56:14 57:17 worst 62:23
---	---	---	--

worth 26:6 111:21	y
worthless 31:9	y 140:20,20
write 62:15	yeah 50:9 70:2
125:14,23	89:4 133:5
152:7 167:25	140:4 141:6,7
167:25	143:1
writes 127:1	year 16:10
written 42:4	years 80:10,10
46:8 78:20	160:20
82:14,16,18	yellow 14:7
104:18 105:2	z
107:4	zero 13:20
wrong 17:18	27:14,19,20,25
23:20 29:23	28:1,2,16 31:2
51:23 55:25	34:8,16,20
124:19 130:2	35:17 36:7,11
150:15 157:15	36:13 59:12
165:9	61:1 90:6
ws 170:4	98:14 109:13
wu 3:5 6:23	118:13,17
49:13,24 50:5	119:10,21,21
50:19 51:11	120:1 156:23
69:7 139:6,10	157:11 165:21
140:5 141:7	zeros 25:19
143:1,14	zoom 33:22
145:15 146:3	
158:14 167:4	
168:4	
wu's 161:17	
x	
x 140:20,20	
xbox 72:21	