

**IN THE UNITED STATES DISTRICT COURT  
FOR THE WESTERN DISTRICT OF TEXAS  
AUSTIN DIVISION**

VALTRUS INNOVATIONS LTD.,  
KEY PATENT INNOVATIONS LIMITED,

Plaintiffs,

v.

ADVANCED MICRO DEVICES, INC.,

Defendant.

Case No. 1:25-cv-510

JURY TRIAL DEMANDED

**COMPLAINT FOR PATENT INFRINGEMENT**

Plaintiff Valtrus Innovations Limited (“Valtrus”) and Plaintiff Key Patent Innovations Limited (“KPI”) (collectively, “Plaintiffs”), by and through their undersigned counsel, plead the following against Advanced Micro Devices, Inc. (“AMD” or “Defendant”) and allege as follows:

**THE PARTIES**

1. Plaintiff Valtrus is the successor in interest to a substantial patent portfolio created by Hewlett Packard Enterprise and its predecessor companies, including Compaq, Verity, and Hewlett-Packard Development Company (collectively, “HPE”). Valtrus is an Irish entity duly organized and existing under the laws of the Republic of Ireland. The address of the registered office of Valtrus is: The Glasshouses GH2, 92 Georges Street Lower, Dun Laoghaire, Dublin A96 VR66, Ireland. HPE’s worldwide corporate headquarters is located in Houston, Texas. One of HPE’s primary US facilities is located in Plano, Texas.

2. Valtrus is the assignee and owns all right and title to U.S. Patent No. 7,930,539 (“the ’539 Patent”).

3. Plaintiff KPI is the beneficiary of a trust pursuant to which Valtrus owns, holds, and asserts the Asserted Patents. KPI is an Irish entity duly organized and existing under the laws of the Republic of Ireland. The address of the registered office of KPI is: The Glasshouses GH2, 92 Georges Street Lower, Dun Laoghaire, Dublin A96 VR66, Ireland.

4. The ’539 Patent was developed by inventors working for HPE. HPE and its predecessors have been developing innovative computer processing and server technology for decades.

5. On information and belief, Defendant AMD is a corporation duly organized and existing under the laws of the State of Delaware, having regular and established places of business in the Western District of Texas, including at 1340 Airport Commerce Drive, Austin, Texas 78741;

7000 West William Cannon Drive, Austin, Texas 78735; and 7171 Southwest Parkway, Austin, Texas 78735.

### **JURISDICTION AND VENUE**

6. This is an action arising under the patent laws of the United States, 35 U.S.C. § 1 *et seq.* Accordingly, this Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

7. This Court has personal jurisdiction over AMD because AMD creates products and services that are and have been used, offered for sale, sold, and purchased in the Western District of Texas, and AMD has committed, and continues to commit, acts of infringement in the Western District of Texas, has conducted business in the Western District of Texas, and/or has engaged in continuous and systematic activities in the Western District of Texas. For example, AMD maintains at least three offices in Austin, Texas.

8. Under 28 U.S.C. §§ 1391(b)-(d) and 1400(b), venue is proper in this judicial district because AMD maintains its principal place of business in this district and has committed and regularly commits acts of infringement within this judicial district giving rise to this action. For example, AMD maintains multiple regular and established places of business in Austin, Texas, including at 1340 Airport Commerce Drive, 7000 West William Cannon Drive, and 7171 Southwest Parkway. On information and belief, AMD's campus on Southwest Parkway occupies approximately 443,000 square feet of space. AMD has had an established presence in the Western District of Texas for decades. For example, on information and belief, AMD employs approximately 3,500 people in the Austin area, and has maintained a presence there since at least 1979.

**FIRST CLAIM**

**(Infringement of U.S. Patent No. 7,930,539)**

9. Plaintiffs re-allege and incorporate herein by reference Paragraphs 1-8 of their Complaint.

10. The '539 Patent, entitled "Computer system resource access control," was duly and lawfully issued on April 19, 2011. A true and correct copy of the '539 Patent is attached hereto as Exhibit 1.

11. The '539 Patent names Donald C. Soltis, Jr., Rohit Bhatia, and Eric R. DeLano as inventors.

12. The '539 Patent has been in full force and effect since its issuance. Valtrus owns by assignment the entire right and title in and to the '539 Patent, including the right to seek damages for any infringement thereof.

13. The '539 Patent generally relates to technology implemented in a computer system having a plurality of resources, where the computer system receives a request from a software program to access one of the resources and determines whether the resource is a protected resource. If it is, the claimed method or device determines whether access to the protected resource should be granted based on whether or not the computer system is operating in a protected mode of operation.

14. Plaintiffs are informed and believe, and thereon allege, that AMD has infringed one or more claims of the '539 Patent, in violation of 35 U.S.C. § 271, by, among other things, making, using, offering to sell, selling, and and/or importing into the United States, without authority or license, AMD products that use the claimed computer system resource access control method and device in an infringing manner. AMD practices every step of at least claim 1 of the '539 Patent in the United States, including one or more steps that it practices in the Western District of Texas.

15. For example, AMD multicore processors, AMD EPYC processors, AMD Ryzen processors, and AMD Athlon processors (collectively, the “Accused Products”), embody every limitation of at least claim 1 of the ’539 Patent, literally or under the doctrine of equivalents, as set forth below. The further descriptions below, which are based on publicly available information, are preliminary examples and are non-limiting.

16. The Accused Products use a computer-implemented method for use in a computer system including a plurality of resources, the method comprising the steps below.

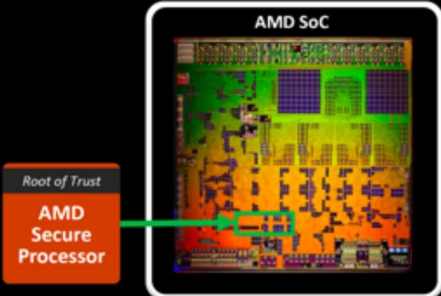
17. As one example, AMD EPYC 7003 Series processors, which include an ARM-based AMD Secure Processor, perform a computer-implemented method for use in a computer system including a plurality of resources. For example, the method is executed by a computer system with a processor where the computer system includes resources such as normal (non-secure) code and secure code.<sup>1</sup>

AMD SECURE PROCESSOR
AMD

### A Dedicated Security Subsystem

- ▲ AMD Secure Processor integrated within SoC  
– 32-bit microcontroller (ARM Cortex-A5)
- ▲ Runs a secure OS/kernel
- ▲ Secure off-chip NV storage for firmware and data (i.e., SPI ROM)
- ▲ Provides cryptographic functionality for secure key generation and key management
- ▲ Enables Secure Platform boot (Hardware validated boot)

*Hardware root of trust provides foundation for platform security*



6 | AMD MEMORY ENCRYPTION TUTORIAL | JUNE 19, 2016 |

<sup>1</sup> Presentation available at: <https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/kaplan>

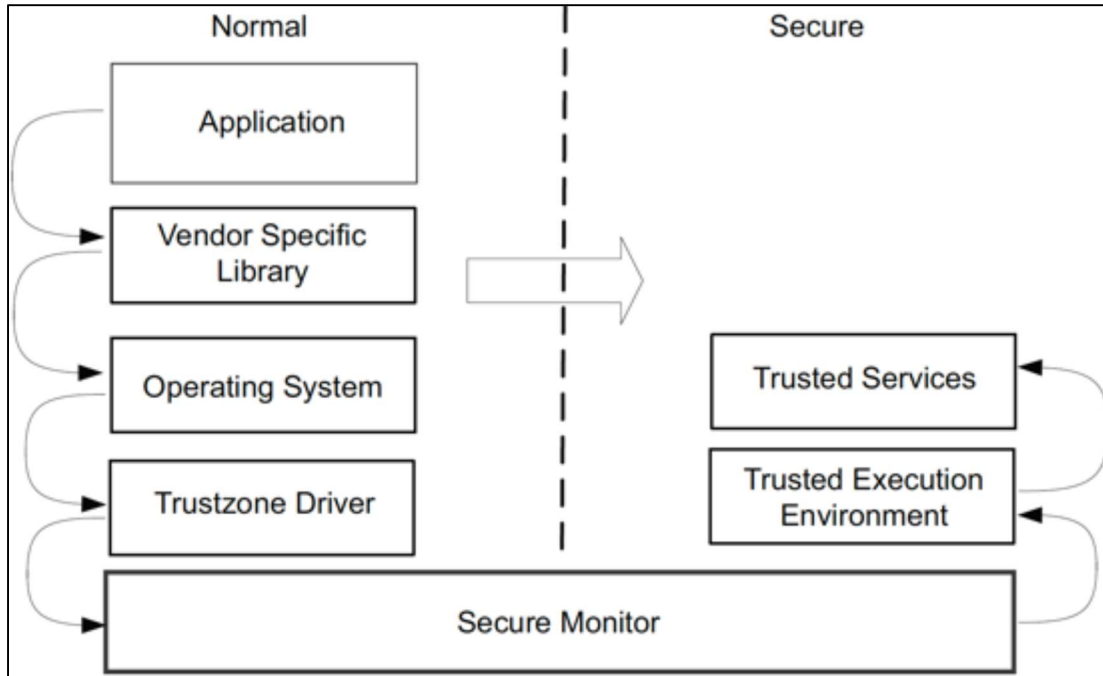
18. The Accused Products perform the step of receiving a request from a software program to access a specified one of the plurality of resources.

19. As one example, AMD EPYC 7003 Series processors perform the step of receiving a request from a software program to access a specified one of the plurality of resources. For example, as shown below, a software program such as the TrustZone API driver in the “Normal [(non-secure)] world” running on the ARM-based AMD Secure Processor may request access to a resource, such as normal (non-secure) or secure application code, on behalf of an application in the “Normal” (non-secure) world.<sup>2</sup>

Generally applications developers won't directly interact with TrustZone (or TEEs or Trusted Services). Instead, one makes use of a high level API (for example, it might be called `reqPayment()`) provided by a Normal world library. The library would be provided by the same vendor as the Trusted Service (for example, a credit card company), and would handle the low level interactions. Figure 21-3 shows this interaction and illustrates the flow from user application calling the API that makes an appropriate OS call, which then passes to the TrustZone driver code, and then passes execution into the TEE through the Secure monitor.

---

<sup>2</sup> See <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>

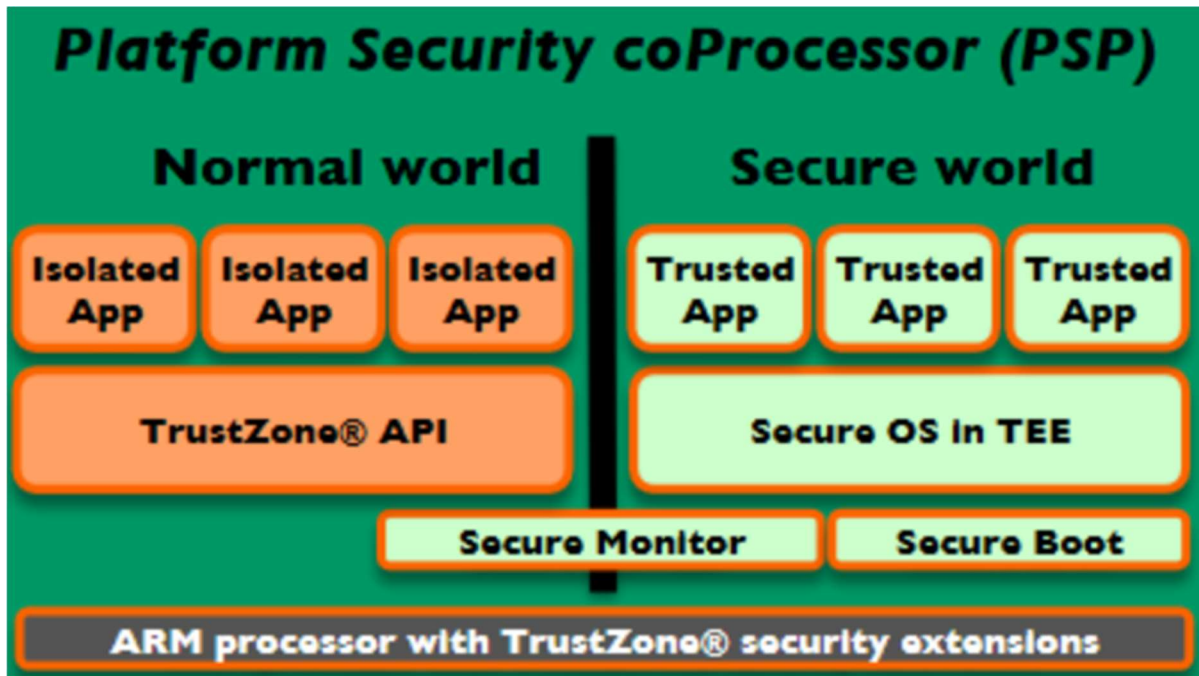


20. The Accused Products perform the step of determining whether the specified one of the plurality of resources is a protected resource.

21. As one example, AMD EPYC 7003 Series processors perform the step of determining whether the specified one of the plurality of resources is a protected resource. For example, the AMD EPYC 7003 Series processors include an ARM-based AMD Secure Processor (also known as a Platform Security coprocessor (PSP)), as shown above, that determines whether a resource is a protected resource, such as secure application code performing a security service in the “Secure world.”<sup>3</sup>

If you do have to access a secure application, you will require a driver-like function to talk to the Secure world OS and Secure applications, but the details of creating that Secure world OS

<sup>3</sup> See <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>; see also <https://freundschafter.com/about-amd-trustzone-amd-platform-security-processor-psp-amd-secure-technology/>



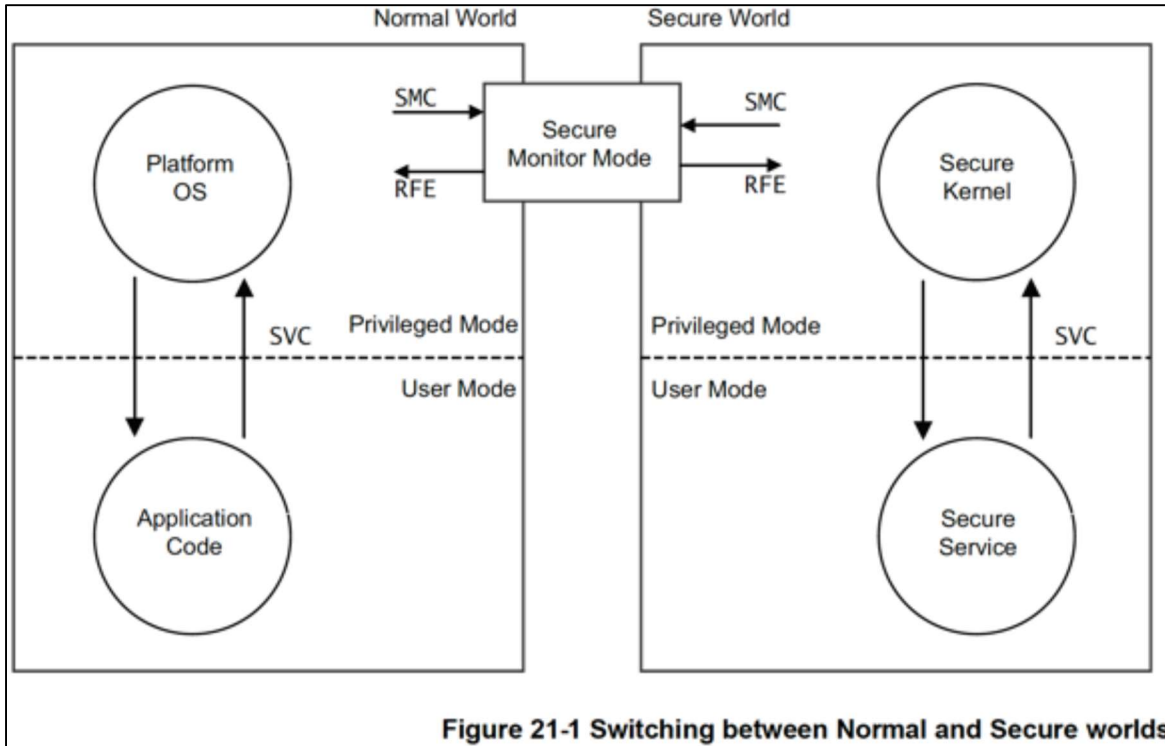
22. If the specified one of the plurality of resources is a protected resource, the Accused Products perform the steps below.

23. For example, if the resource is a protected resource, such as a resource application code performing a security service in the secure world as described above, then a secure monitor call (SMC) is made to the secure monitor in order for the secure code to be executed in the secure world, as shown below.<sup>4</sup>

It is common to share data between the Secure and Normal worlds. For example, in the Secure world you might have a signature checker. The Normal world can request that the Secure world verifies the signature of a downloaded update, using the SMC call. The Secure world requires

<sup>4</sup> See <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>;  
<https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture>;  
<https://developer.arm.com/documentation/ddi0406/cd>





The Secure Monitor Call instruction, SMC, requests a Secure Monitor function, causing the processor to enter Monitor mode. For more information, see [SMC \(previously SMI\) on page B9-1988](#).

24. The Accused Products perform the step of, if the computer system is operating in a protected mode of operation, then denying the request regardless of access rights associated with the software program including software programs having a most-privileged level.

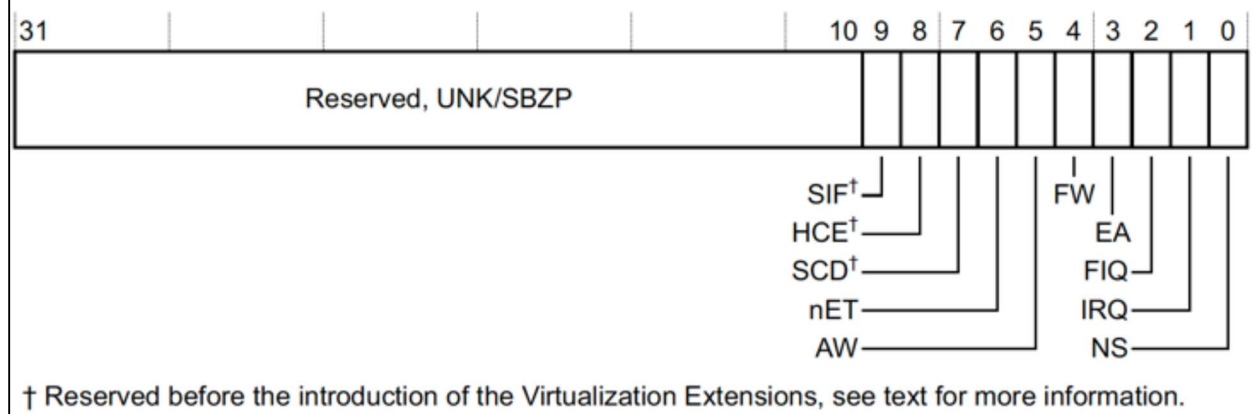
25. For example, if the computer system (*i.e.*, a processor core) is operating in a protected mode of operation, defined, for example, by setting the SCR.SCD bit to 1 and thus disabling the secure monitor call at PL1 and above, then the request is denied regardless of access rights and privilege level, because the SMC instruction cannot execute and the system cannot enter the Secure state to access the secure application code.<sup>5</sup>

<sup>5</sup> <https://developer.arm.com/documentation/ddi0406/cd>

When the **SCR.SCD** bit is set to 1, entry to Secure state by taking a Secure Monitor Call exception is disabled. This means that, when **SCR.SCD** is set to 1:

- An SMC instruction executed in Non-secure state, and not trapped by the **HCR.TSC** mechanism described in *Trapping use of the SMC instruction on page B1-1253*, is UNDEFINED.
- An SMC instruction executed in a Secure PL1 mode is UNPREDICTABLE.

The SCR bit assignments are:



#### SCD, bit[7], when implementation includes the Virtualization Extensions

Secure Monitor Call disable. Makes the SMC instruction UNDEFINED in Non-secure state. The possible values of this bit are:

- 0** SMC executes normally in Non-secure state, performing a Secure Monitor Call.
- 1** SMC instruction is UNDEFINED in Non-secure state.

26. The Accused Products perform the step of processing the request based on the access rights associated with the software program if the computer system is not operating in the protected mode of operation.

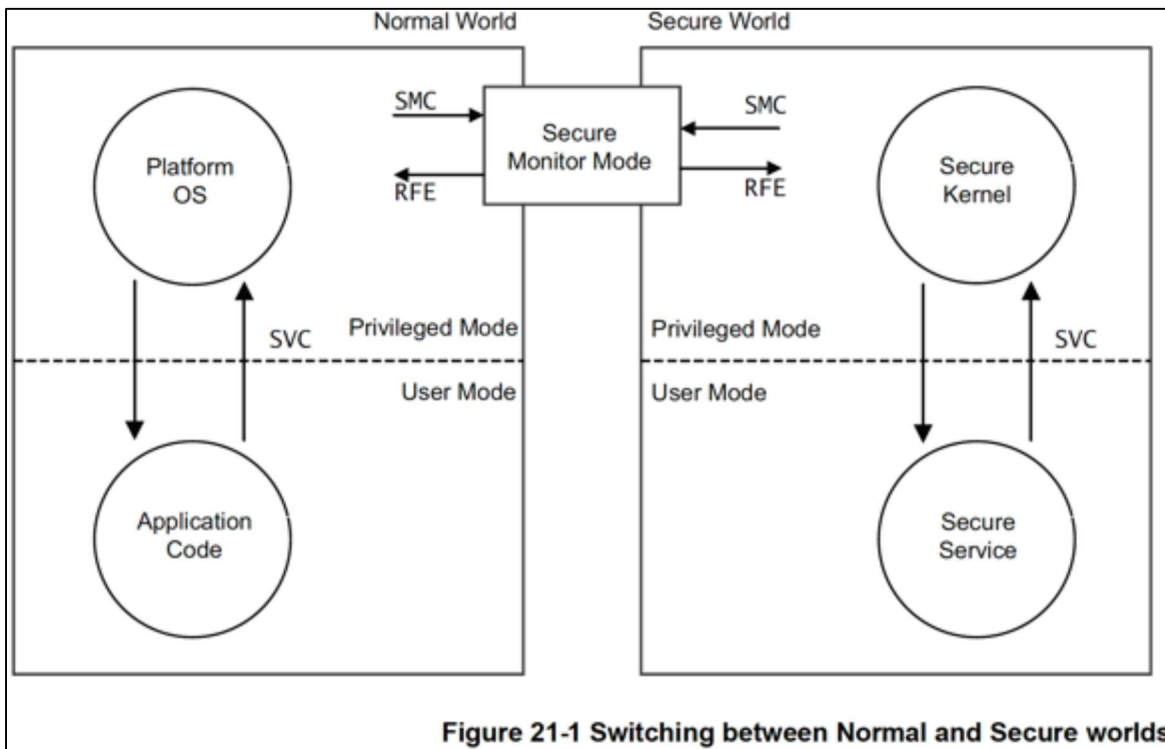
27. As one example, AMD EPYC 7003 Series processors perform the step of processing the request based on the access rights associated with the software program if the computer system is not operating in the protected mode of operation. For example, if the computer system (*i.e.*, a processor core) is not operating in a protected mode of operation, defined, for example, by setting the **SCR.SCD** bit described above to 0 and thus enabling the secure monitor

call, then the request is processed based on the access rights—such as to SMC—associated with the software program.<sup>6</sup>

It is common to share data between the Secure and Normal worlds. For example, in the Secure world you might have a signature checker. The Normal world can request that the Secure world verifies the signature of a downloaded update, using the SMC call. The Secure world requires

Secure Monitor Call causes a Secure Monitor Call exception. For more information, see [Secure Monitor Call \(SMC\) exception on page B1-1210](#).

SMC is available only from software executing at PL1 or higher. It is UNDEFINED in User mode.



28. On information and belief, AMD had knowledge of the '539 Patent prior to the filing of this complaint, at least because Plaintiffs have previously raised infringement claims regarding the '539 Patent against third parties, including customers of AMD, that implicate the Accused Products.

<sup>6</sup> See <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>; <https://developer.arm.com/documentation/ddi0406/cd>

29. Plaintiffs are informed and believe, and thereon allege, that AMD actively, knowingly, and intentionally has induced infringement of the '539 Patent by, for example, using, selling, and offering for sale the Accused Products, which in turn use the method claimed by the '539 Patent. AMD offers for sale and sells said Accused Products with the intent to encourage and facilitate infringing uses of those products in the Western District of Texas, in the United States, and throughout the world.

30. As a result of AMD's infringement of the '539 Patent, Plaintiffs have been damaged. Plaintiffs are entitled to recover damages sustained as a result of AMD's wrongful acts in an amount subject to proof at trial.

31. In addition, AMD's infringing acts and practices have caused and are causing immediate and irreparable harm to Plaintiffs.

32. Plaintiffs are informed and believe, and thereon allege, that AMD's infringement of the '539 Patent has been and continues to be willful. As noted above, on information and belief, AMD has had knowledge of the '539 Patent and its infringement of the '539 Patent. AMD has deliberately continued to infringe in a wanton, malicious, and egregious manner, with reckless disregard for Plaintiffs' patent rights. Thus, AMD's infringing actions have been and continue to be consciously wrongful.

**PRAYER FOR RELIEF**

WHEREFORE, Plaintiffs pray for judgment against AMD as follows:

- A. That AMD has infringed the '539 Patent, and unless enjoined will continue to infringe the '539 Patent;
- B. That AMD has willfully infringed the '539 Patent;
- C. That AMD pay Plaintiffs damages adequate to compensate Plaintiffs for AMD's past, present, and future infringement of the '539 Patent, together with interest and costs under 35 U.S.C. § 284;
- D. That AMD be ordered to pay prejudgment and post-judgment interest on the damages assessed;
- E. That AMD pay Plaintiffs enhanced damages pursuant to 35 U.S.C. § 284;
- F. That AMD be ordered to pay supplemental damages to Plaintiffs, including interest, with an accounting, as needed;
- G. That AMD be enjoined from infringing the '539 Patent, or if its infringement is not enjoined, that AMD be ordered to pay ongoing royalties to Plaintiffs for any post-judgment infringement of the '539 Patent;
- H. That this is an exceptional case under 35 U.S.C. § 285, and that AMD pay Plaintiffs' attorneys' fees and costs in this action; and
- I. That Plaintiffs be awarded such other and further relief, including equitable relief, as this Court deems just and proper.

**DEMAND FOR JURY TRIAL**

Pursuant to Federal Rule of Civil Procedure 38(b), Plaintiffs hereby demand a trial by jury on all issues triable to a jury.

Dated: April 4, 2025

Respectfully submitted,

By: /s/ Max Ciccarelli

**Max Ciccarelli**  
State Bar No. 00787242  
CICCARELLI LAW FIRM LLC  
100 N. 6th Street, Suite 503  
Waco, Texas 76701  
Tel: 214-444-8869  
Email: [Max@CiccarelliLawFirm.com](mailto:Max@CiccarelliLawFirm.com)