United States Court of Appeals for the Federal Circuit

INTEL CORPORATION,
Appellant

v.

QUALCOMM INCORPORATED,
Cross-Appellant

2020-1828, 2020-1867

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in Nos. IPR2018-01334, IPR2018-01335, IPR2018-01336.

Decided: December 28, 2021

THOMAS SAUNDERS, Wilmer Cutler Pickering Hale and Dorr LLP, Washington, DC, argued for appellant. Also represented by DAVID LANGDON CAVANAUGH, CLAIRE HYUNGYO CHUNG; JOSEPH F. HAAG, Palo Alto, CA.

JENNIFER L. SWIZE, Jones Day, Washington, DC, argued for cross-appellant. Also represented by ROBERT BREETZ, DAVID B. COCHRAN, DAVID MICHAEL MAIORANA, JOSEPH M. SAUER, Cleveland, OH; KELLY HOLT, New York, NY; MATTHEW JOHNSON, JOSHUA R. NIGHTINGALE, Pittsburgh, PA; ISRAEL SASHA MAYERGOYZ, Chicago, IL.

Before PROST, TARANTO, and HUGHES, *Circuit Judges*.

TARANTO, *Circuit Judge*.

Qualcomm Inc. owns U.S. Patent No. 8,838,949, which addresses multi-processor systems in which software stored in non-volatile memory coupled to a first processor is to be used by a second processor. The patent describes and claims systems, methods, and apparatuses for efficiently retrieving an executable software image from the first processor's non-volatile memory and loading it for use by the second processor. Intel Corp. challenged all claims of the '949 patent as unpatentable for obviousness in three inter partes reviews (IPRs) before the Patent and Trademark Office. The Office's Patent Trial and Appeal Board consolidated the proceedings and issued a final written decision holding that Intel had proved unpatentable claims 10, 11, 13–15, and 18–23, but not claims 1–9, 12, 16, and 17. Intel Corp. v. Qualcomm Inc., IPR2018-01334, 2020 WL 1286306, at *27 (P.T.A.B. Mar. 16, 2020) (Final Written Decision). Intel appeals.

We hold first that Intel has adequately demonstrated Article III standing to press this appeal. On the merits, we hold that in the decision before us, the Board failed to tie its construction of the phrase "hardware buffer" to the actual invention described in the specification. For that reason, we vacate the Board's decision as to claims 1–9 and 12 and remand for a new construction. As to claims 16 and 17, which are in means-plus-function format, we also vacate and remand. We conclude that the Board failed to determine for itself whether there is sufficient corresponding structure in the specification to support those claims and whether it can resolve the patentability challenges despite the (potential) indefiniteness of those claims.

I

A

The patent addresses a system with multiple processors, each of which must execute its own "boot code" to play its operational role in the system. Such code must be stored in non-volatile memory (*e.g.*, flash memory or read-only memory), since volatile memory is cleared when the device powers down; and the boot code generally must be transferred to its corresponding processor's volatile memory in order to be executed by that processor. '949 patent, col. 1, lines 39–41. In a multi-processor system, one possible design choice is to store the boot code for each processor in its own separate non-volatile memory. Another choice, to avoid the costs of multiple memories each adequate for such storage, is to store the boot code for one processor in the non-volatile memory of another processor, permitting elimination or shrinkage of the non-volatile memory of the first processor. *Id.*, col. 1, line 60, through col. 2, line 14.

The '949 patent, titled "Direct Scatter Loading of Executable Software Image from a Primary Processor to One or More Secondary Processor in a Multi-Processor System," assumes the latter, shared-storage choice. It addresses the problem, inherent in that choice, of loading the boot code for a "secondary" processor (into its volatile memory) from the non-volatile memory of a "primary" processor. *Id.*, col. 2, line 58, through col. 3, line 2. It uses a "direct scatter load" procedure to do so. "Scatter loading" refers to moving a "binary multi-segmented" software image into scattered parts (as opposed to one contiguous block) of the secondary processor's "system memory" before executing it. *Id.*, col. 2, lines 14–22. The patent discloses a "direct" scatter loading process, through which the segments of the software image are transmitted "directly" from a "hardware buffer" to their final locations in the secondary processor's "system memory." *Id.*, col. 2, lines 58–63.

Claims 1 and 2 are representative for the claim-construction issue on appeal. They recite:

> 1. A multi-processor system comprising:
>
> a secondary processor comprising:
>
> system memory and a ***hardware buffer*** for receiving an image header and at least one data segment of an executable software image, the image header and each data segment being received separately, and
>
> a scatter loader controller configured:
>
> to load the image header, and
>
> to scatter load each received data segment based at least in part on the loaded image header, directly from the hardware buffer to the system memory;
>
> a primary processor coupled with a memory, the memory storing the executable software image for the secondary processor; and
>
> an interface communicatively coupling the primary processor and the secondary processor, the executable software image being received by the secondary processor via the interface.[1]

---

[1] The indentation of the last two components of the "multi-processor system" (the "primary processor" and the "interface") has been altered from the original to reflect the

> 2. The multi-processor system of claim 1 in which the scatter loader controller is configured to load the executable software image ***directly from the hardware buffer to the system memory of the secondary processor without copying data between system memory locations on the secondary processor***.

Id., col. 12, line 60, through col. 13, line 16 (emphases added).

Claim 16 is relevant to the means-plus-function issue on appeal. It recites:

> 16. An apparatus comprising:
>
> means for receiving at a secondary processor, from a primary processor via an inter-chip communication bus, an image header for an executable software image for the secondary processor that is stored in memory coupled to the primary processor, the executable software image comprising the image header and at least one data segment, the image header and each data segment being received separately;
>
> ***means for processing***, by the secondary processor, the image header to determine at least one location within system memory to which the secondary processor is coupled to store each data segment;
>
> means for receiving at the secondary processor, from the primary processor via the inter-chip communication bus, each data segment; and

---

fact that they are parts of the multi-processor system, not parts of the secondary processor.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.