NOTE: This disposition is nonprecedential.

United States Court of Appeals for the Federal Circuit

INTEL CORPORATION,
Appellant

v.

VLSI TECHNOLOGY LLC,
Appellee

2020-1744, 2020-1745

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in Nos. IPR2018-01312, IPR2018-01661.

Decided: May 26, 2021

MARK CHRISTOPHER FLEMING, Wilmer Cutler Pickering Hale and Dorr LLP, Boston, MA, argued for appellant. Also represented by DOMINIC E. MASSA, DONALD R. STEINBERG; ANH-KHOA T. TRAN, Palo Alto, CA; DAVID P. YIN, TODD ZUBLER, Washington, DC.

KAMRAN VAKILI, Irell & Manella LLP, Newport Beach, CA, argued for appellee. Also represented by MICHAEL RICHARD FLEMING, Los Angeles, CA.

Before NEWMAN, LOURIE, and DYK, *Circuit Judges*.

DYK, *Circuit Judge*.

Intel Corp. appeals the final written decisions by the Patent Trial and Appeal Board ("Board") in two *inter partes* review proceedings. The Board found that Intel did not show that the challenged claims of U.S. Patent No. 8,020,014 (the "'014 patent") were unpatentable as obvious. We *affirm in part, reverse in part, and remand*.

BACKGROUND

The '014 patent is directed to reducing the power consumption of a computing device by selectively powering down a component of the device, such as a memory cache. Some discussion of the underlying computer architecture is helpful.

I

Computer systems can be designed to incorporate different types of memory, which offer advantages and disadvantages in terms of storage capacity, access speed, and cost. The different memories in a computer are typically arranged in a hierarchy, in which a smaller, faster, and more expensive memory contains a subset of the information stored in a larger, slower, and less expensive memory.

A central processing unit ("CPU") may include an integrated cache memory, separate from the computer's main memory. Because the cache is smaller, faster, and physically closer to the CPU, the CPU can access data in the cache more quickly than data in main memory. Performance can thus be increased by copying items that the CPU accesses frequently from main memory to the cache.

Because accessing data in the cache is more efficient, a CPU generally searches the cache first when attempting to retrieve information. A "cache hit" occurs when the CPU

finds the requested data in the cache. In the event of a "cache miss"—when the requested data is *not* available in the cache—the CPU must instead retrieve the data from main memory.

After a cache miss, the data retrieved from main memory is copied into the cache, given that it will likely be needed again soon. If the cache is already full, the CPU must delete existing data from the cache to make room for the newly retrieved data. Before deleting data from the cache, the CPU must check whether the data to be deleted was modified while in the cache. Data that has not been modified—"clean" data—may simply be deleted, as an identical copy exists in main memory. However, "dirty" data—i.e., data modified while in the cache—must be written back to main memory before being deleted from the cache, so that main memory reflects the changes to the data.

A cache can be divided into portions—variously called "ways," "blocks," or "lines"—which can be powered on or powered down independently of one another. A cache way can store data when it is powered on, enabling the CPU to access data more efficiently. However, the energy required to keep cache ways powered on, called "leakage" or "static" power, represents a substantial portion of a device's power consumption. Leakage power can be reduced by turning off a cache way, but when a cache way is powered down, the information stored therein is lost. Thus, powering down a cache way increases the risk that the CPU will not be able to retrieve data from the cache, causing a cache miss—and the extra operations following a cache miss require additional "dynamic" power to perform.[1] That is, when a cache

---

[1] Note that the dynamic power cost of a cache miss varies with the number of operations required in response to the miss. E.g., all else held constant, more power will be consumed if data from the cache must be deleted to make

way is powered down, its data is rendered unavailable, and more dynamic power is required to retrieve the data from main memory. Powering down a cache way may also require operations to write dirty information back to main memory, imposing an additional dynamic power cost.[2] Strategies for reducing the power consumption of cache ways must therefore balance the reduction in leakage power against the risk of increased dynamic power costs associated with powering down a cache way.

II

The '014 patent, which issued on September 13, 2011, "relates to a method for power management and a device having power management capabilities, and especially for power reduction of a cache memory." '014 patent, col. 1, ll. 7–9. Claim 1, which the Board found illustrative, reads as follows:

> A method for power reduction, the method comprises:
>
> selectively providing power to at least a portion of a component of an integrated circuit during a low power mode; and
>
> determining whether to power down the at least portion of the component in response to a relationship between an estimated power gain and an estimated power loss resulting from powering down

---

room for the data retrieved from main memory; less power will be used if the cache is not full and the retrieved data can be copied directly into the cache.

[2] The claims of the '014 patent encompass both types of dynamic power costs, viz., to retrieve data from main memory when a cache way is powered down, and to write dirty data back to main memory before powering down a cache way.

> the at least portion of the component during the low power mode.

Id., col. 7, ll. 27–35.[3]

According to the embodiments in the specification, the cost-benefit analysis of the estimated power gain and loss focuses on one aspect of powering down a cache: "estimat[ing] the amount of dirty data stored within the cache" and comparing this estimate "to a power gating threshold TH." *Id.*, col. 4, ll. 49–50, 64–65; *see also id.*, col. 6, ll. 42–45. (The specification does not explain what the threshold measures or how it is calculated or chosen, though apparently it can be "predefined" by, e.g., the manufacturer. *See id.*, col. 5, ll. 64–67.) The estimated amount of dirty information is used as a proxy for the power loss resulting from powering down the cache (due to the extra operations needed to write back the dirty data), while the threshold is "representative of the estimated power gain." *See id.*, col. 6, ll. 32–45. Thus, if the estimated amount of dirty information is below the threshold, "then the cache memory is flushed and is powered down." *Id.*, col. 4, ll. 65–67. The specification further explains that this analysis can be conducted separately across different cache ways, so that the decision to power down can be made separately for each cache way. *Id.*, col. 5, ll. 1–2; col. 6, l. 56–col. 7, l. 2.

In other words, the specification of the '014 patent addresses a situation where part of a cache is to be powered down, and some dynamic power must first be expended to write back the dirty information in that part of the cache to main memory. The specification does not address the dynamic power costs of retrieving data from main memory after a cache miss.

---

[3] Claim 12, which the Board also found illustrative, recites substantially the same limitations for a device rather than a method. *See* '014 patent, col. 8, ll. 8–18.

III

In 2018, Intel filed two petitions for *inter partes* review, arguing that various claims of the '014 patent were invalid as obvious under 35 U.S.C. § 103. As relevant on appeal, Intel's argument centered on two prior art references: U.S. Patent No. 5,761,715 (hereinafter "Takahashi"), and an academic paper titled "Let Caches Decay: Reducing Leakage Energy via Exploitation of Cache Generational Behavior," by Zhigang Hu et al. (hereinafter "Hu"). Intel argued that claims 1–3, 12–14, 18, and 20 were obvious over Takahashi alone, and that claims 1–5, 12–16, 18, and 20 were obvious over the combined teachings of Takahashi and Hu.

The Board entered decisions instituting *inter partes* review on February 20 and March 1, 2019. In final written decisions dated February 19 and 26, 2020, the Board found that Intel had failed to demonstrate obviousness by a preponderance of the evidence and upheld all the challenged claims of the '014 patent. Intel appeals. We have jurisdiction under 28 U.S.C. § 1295(a)(4)(A).

DISCUSSION

A patent claim is invalid "if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art." 35 U.S.C. § 103(a) (pre-AIA).[4] Obviousness is a mixed question of law and fact. *Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016). In reviewing the Board's determination on

---

[4] Congress amended § 103 when it enacted the Leahy-Smith America Invents Act ("AIA"). Pub. L. No. 112-29, § 3(c), 125 Stat. 284, 287 (2011). However, because the challenged claims of the '014 patent have an effective filing date before March 16, 2013, the pre-AIA version of § 103 applies. *See id.* § 3(n)(1), 125 Stat. at 293.

obviousness, we review the ultimate legal conclusion de novo and the underlying factual findings for substantial evidence. *Id.* (citing *In re Cuozzo Speed Techs., LLC*, 793 F.3d 1268, 1280 (Fed. Cir. 2015)).

I

Before the Board, Intel argued that many of the challenged claims were obvious over Takahashi alone. In relevant part, Intel asserted that Takahashi disclosed determining whether to power down a cache way "in response to a relationship between an estimated power gain and an estimated power loss," as recited by the '014 patent.

In Takahashi, the decision to power down a cache way is based on the change in the cache-miss rate. Takahashi teaches storing a previous measurement of the cache-miss rate as a "predetermined value." *See* Takahashi, col. 3, ll. 23–31; col. 9, ll. 54–61. Next, if the present cache-miss rate does not exceed the predetermined value (i.e., the previous cache-miss rate), then one of the activated cache ways will be powered down. *See id.*, col. 9, ll. 55–59; *see also id.*, col. 3, ll. 57–65.

Intel argued that Takahashi's predetermined value was "representative of the estimated power gain" from powering down a cache way. *See* J.A. 4038, 5039. The Board rejected this ground for Intel's petition. The Board acknowledged that Takahashi teaches an increase in the cache-miss rate as "a proxy for estimated power loss." J.A. 16, 45. However, the Board concluded that the predetermined value—a previous measurement of the cache-miss rate—did not constitute an estimated power gain.

We agree. As the Board observed, Takahashi does not suggest that the predetermined value represents an estimated power gain. Rather, Takahashi explains that "the change of the cache-miss rate" is used to "determine[] the optimum number of [cache] ways to be efficiently accessed." Takahashi, col. 2, ll. 54–60. While Takahashi uses the

previous measurement of the cache-miss rate to track whether the cache-miss rate is improving or worsening, the previous cache-miss rate itself does not estimate the power gain from powering down a cache way.

Intel also argued that the '014 patent uses substantially the same method as Takahashi, insofar as the '014 patent compares an estimated amount of dirty information (a proxy for estimated power loss) against "a power gating threshold TH" (which may be a predefined value). Thus, Intel contends, if the unspecified "threshold TH" can represent an estimated power gain in the '014 patent, then the materially identical disclosure in Takahashi must also teach an estimated power gain, rendering this limitation of the '014 patent obvious.

The purported similarity between Takahashi's predetermined value and the '014 patent's threshold TH, however, does not suffice to show that Takahashi teaches an estimated power gain. Instead, this argument seems better suited to a challenge to written description or enablement under 35 U.S.C. § 112—i.e., that the '014 patent itself fails to disclose an estimated power gain. But such a challenge is not before us on appeal, nor could Intel have raised it before the Board. *See* 35 U.S.C. § 311(b).

We thus find no error in the Board's determination that Takahashi does not teach an estimated power gain.

II

We now turn to the Board's determination that the combined teachings of Takahashi and Hu did not render obvious the claim limitation of "determining whether to power down the at least portion of the component in response to a relationship between an estimated power gain and an estimated power loss." In relevant part, Intel argued in its petition (1) that the "relationship" taught in Hu could be combined with the "determining" step in

Takahashi, and, alternatively, (2) that Hu taught both the "determining" and "relationship" limitations.

As previously discussed, Takahashi discloses a method for determining whether to power down a cache way based on a change in the cache-miss rate.

Hu seeks to "reduc[e] leakage power" by selectively powering down cache ways. J.A. 1118. As Hu explains: "We wish to turn off cache lines as often as possible in order to save leakage power. We balance this, however, against a desire to avoid increasing the miss rate of the L1 cache." *Id.* at 1123.

Hu introduces the "*L2Access:leak* ratio" to measure "the energy dissipated due to an extra [cache] miss." *Id.* This ratio is used to decide whether to power down the L1 cache. *Id.* Simply put, *leak* represents "the leakage energy dissipated by the L1 data cache" when powered on, while *L2Access* represents the power cost of accessing the "L2 cache" when the L1 cache is powered down and a cache miss ensues. *Id.* at 1123–24.

Using the *L2Access:leak* ratio, Hu proposes a time-based "cache decay policy" for powering down cache lines:

> The longer we wait [to turn off a cache line], the higher the leakage energy dissipated. On the other hand, if we prematurely turn off a line that may still have hits, then we inject extra misses that incur dynamic power for L2 cache accesses. Competitive algorithms point us towards a solution: we could leave each cache line turned on until the static energy it has dissipated since its last access is precisely equal to the dynamic energy that would be dissipated if turning the line off induced an extra miss.

Id. at 1125.

For L1 and L2 caches of the sizes under consideration in Hu, Hu cites empirical estimates of "3 to 5 nJ" per access of the L2 cache and ".45 nJ static leakage per cycle" of the L1 cache. *Id.* at 1123–24. This yields an *L2Access:leak* "ratio of 8.9 relating extra miss power to static leakage per cycle" (i.e., 4 nJ ÷ 0.45 nJ). *Id.* at 1124. Because "these estimates will vary widely with design style and fabrication technology," Hu studies the *L2Access:leak* ratio at assumed values of 5, 10, 20, and 100. *Id.*; *see also id.* at 1131–32.

Hu tested its time-based decay policy by running "a collection of integer and floating point programs." *Id.* at 1129. The results showed that powering down a cache line after an appropriate "decay interval" led to reduced power consumption, and that lower values of the *L2Access:leak* ratio corresponded to shorter optimal decay intervals. *See id.* at 1131–32, 1132 fig.9. Thus, Hu concludes, "a decay interval can be chosen considering the relative cost of a miss to leakage power." *Id.* at 1132.

A

Before the Board, Intel argued that the *L2Access:leak* ratio in Hu teaches the "relationship" limitation of the '014 patent and could be combined with the "determining" step disclosed in Takahashi. The Board, however, refused to consider Intel's "newly raised argument" for combining the "determining" step of Takahashi with the "relationship" in Hu, stating that Intel raised this combination "for the first time" at the oral hearing. J.A. 24–25, 53. The Board erred by disregarding this argument.

Intel's petitions for *inter partes* review plainly argued that Takahashi discloses the "determining whether to power down" limitation, which could in turn be combined with Hu's specific "relationship" (i.e., the *L2Access:leak* ratio). Intel argued at length that Takahashi alone teaches "determining whether to power down," and incorporated this material by reference when it turned to the

combination of Takahashi and Hu. The Board's institution decisions recognized that Intel raised this argument: "The Petition proposes that a person of ordinary skill in the art would apply Hu's comparison of estimated power gain (leakage energy that would be saved) and estimated power loss (based on additional cache misses) to determine when to power down a cache way in Takahashi." J.A. 4159, 5154. Furthermore, Intel expressly cited both Takahashi and Hu when it addressed the "determining . . . in response to a relationship" limitation.

The Board therefore erred when it declined to consider this argument. We vacate the Board's final written decisions on this point and remand for due consideration of Intel's argument. *See, e.g.*, *Vicor Corp. v. SynQor, Inc.*, 869 F.3d 1309, 1324 (Fed. Cir. 2017) (noting the Board's "requirement to address all grounds for proposed rejection under the APA" and remanding for the Board to address the improperly disregarded arguments).

B

Intel also argued that Hu itself teaches "determining whether to power down" in response to the *L2Access:leak* ratio. The Board rejected this argument. According to the Board:

> Hu . . . does not teach using the *L2Access:leak* ratio to determine whether to power down the L1 cache . . . . Rather, Hu discloses determining values of the normalized cache leakage energy for different decay intervals at four values of the *L2Access:leak* ratio—5, 10, 20, and 100. Hu uses these assumed values of the *L2Access:leak* ratio as inputs to generate simulation data, not to determine whether to power down a cache or portion of a cache.

J.A. 24, 52 (citation omitted). This finding mischaracterizes Hu and is not supported by substantial evidence.

As the Board itself observed, a "'basic premise' of Hu's evaluations 'is to measure the static power saved by turning off portions of the cache, and then compare it to the extra dynamic power dissipated' by turning off cache lines." J.A. 20, 49 (quoting *id.* at 1122). Hu formalizes this comparison with the *L2Access:leak* ratio and teaches using the *L2Access:leak* ratio to estimate the decay interval for turning off a cache. *E.g.*, *id.* at 1125 (using *L2Access:leak* ratio of "roughly nine" to estimate an optimal decay interval of "roughly 10,000 cycles"). Hu's experimental results also show that the *L2Access:leak* ratio affects the optimal decay interval for powering down a cache. *E.g.*, *id.* at 1131, 1132 fig.9 (explaining that "short decay intervals may induce extra cache misses by turning off cache lines prematurely; this effect is particularly bad when *L2Access:leak* is 100 because high ratios mean that the added energy cost of additional L2 misses is quite high"). That is, Hu teaches using the *L2Access:leak* ratio to estimate the optimal decay interval, which is in turn used to determine when to power down a cache. This satisfies the claim limitation of "determining whether to power down," because the decision to power down is governed by the decay interval, which is calculated using the *L2Access:leak* ratio.

It is also immaterial that Hu's experimental method relied in part on assumed values of the *L2Acess:leak* ratio. Hu demonstrates that empirically measured values can be used for *L2Access* and *leak*; therefore, Hu teaches that an optimal decay interval can be calculated based on the static and dynamic energy consumption of a given memory structure.

The Board thus erred in determining that Hu does not use the *L2Access:leak* ratio to determine whether to power down a cache or portion of a cache.

However, the fact that Hu discloses using the *L2Access:leak* ratio to determine whether to power down a cache (or portion thereof) does not resolve the obviousness

question, as Hu does not involve all of the limitations of the challenged claims. On remand, the Board must address whether the combination of Takahashi and Hu satisfies all of the claim limitations, and whether there was a motivation to combine Takahashi and Hu with a reasonable expectation of success.

CONCLUSION

We affirm the Board's determination that Takahashi does not disclose an "estimated power gain"; reverse the Board's decision that Intel failed to properly raise the argument that Takahashi teaches the "determining" step; and reverse the Board's determination that Hu does not teach using the *L2Access:leak* ratio to determine whether to power down a cache or portion of a cache.

AFFIRMED IN PART, REVERSED IN PART, AND REMANDED

COSTS

No costs.